electronics

MDPI

Article

Depth-Quality Purification Feature Processing for Red Green Blue-Depth Salient Object Detection

Shijie Feng [1], Li Zhao [1,*], Jie Hu [1], Xiaolong Zhou [2] and Sixian Chan [3,4,*]

[1] Key Laboratory of Intelligent Informatics for Safety & Emergency of Zhejiang Province, Wenzhou University, Wenzhou 325035, China; 21451943004@stu.wzu.edu.cn (S.F.); 20160204@wzu.edu.cn (J.H.)
[2] The College of Electrical and Information Engineering, Quzhou University, Quzhou 324000, China; xiaolong@qzc.edu.cn
[3] The College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China
[4] Hubei Key Laboratory of Intelligent Vision Based Monitoring for Hydroelectric Engineering, The College of Computer and Information, China Three Gorges University, Yichang 443002, China
* Correspondence: lizhao@wzu.edu.cn (L.Z.); sxchan@zjut.edu.cn (S.C.); Tel.: +86-173-5722-8908 (S.C.)

Abstract: With the advances in deep learning technology, Red Green Blue-Depth (RGB-D) Salient Object Detection (SOD) based on convolutional neural networks (CNNs) is gaining more and more attention. However, the accuracy of current models is challenging. It has been found that the quality of the depth features profoundly affects the accuracy. Several current RGB-D SOD techniques do not consider the quality of the depth features and directly fuse the original depth features and Red Green Blue (RGB) features for training, resulting in enhanced precision of the model. To address this issue, we propose a depth-quality purification feature processing network for RGB-D SOD, named DQPFPNet. First, we design a depth-quality purification feature processing (DQPFP) module to filter the depth features in a multi-scale manner and fuse them with RGB features in a multi-scale manner. This module can control and enhance the depth features explicitly in the process of cross-modal fusion, avoiding injecting noise or misleading depth features. Second, to prevent overfitting and avoid neuron inactivation, we utilize the RReLU activation function in the training process. In addition, we introduce the pixel position adaptive importance (PPAI) loss, which integrates local structure information to assign different weights to each pixel, thus better guiding the network's learning process and producing clearer details. Finally, a dual-stage decoder is designed to utilize contextual information to improve the modeling ability of the model and enhance the efficiency of the network. Extensive experiments on six RGB-D datasets demonstrate that DQPFPNet outperforms recent efficient models and delivers cutting-edge accuracy.

Keywords: red green blue-depth salient object detection; convolutional neural network; cross-modal fusion; dual-stage decoder

Citation: Feng, S.; Zhao, L.; Hu, J.; Zhou, X.; Chan, S. Depth-Quality Purification Feature Processing for Red Green Blue-Depth Salient Object Detection. *Electronics* **2024**, *13*, 93. https://doi.org/10.3390/electronics13010093

Academic Editors: Haibin Wu, Aili Wang and Yuji Iwahori

Received: 25 November 2023
Revised: 12 December 2023
Accepted: 19 December 2023
Published: 25 December 2023

1. Introduction

Visual saliency refers to a human visual simulation system that uses algorithms to simulate human visual features and locate prominent areas in an image. Salient Object Detection (SOD) is designed to find the most appealing features of an image. It has rapidly developed and is widely used in many fields, including object tracking [1], object detection [2,3], object segmentation [4,5], and other computer vision tasks for pre-processing [6]. Deep learning has advanced considerably over the past few years, and many SOD methods have been proposed. However, the majority of current models for SOD can only handle RGB images.

Park et al. proposed a unique surface-defect detection method [7] that utilizes a deep nested convolutional neural network (NC-NET) with attention and guiding modules to segment defect regions from complicated backgrounds precisely and adaptively refine features.

To overcome the inherent limitations of convolution, SwinE-Net [8] effectively combines EfficientNet, driven by a CNN, and the Vision Transformer (VIT)-based Swin Transformer for segmentation. This combination preserves global semantics while maintaining low-level characteristics, demonstrating specific generalization and scalability. CoEg-Net [9] employs a shared attention projection technique to facilitate fast learning from public information, utilizing vast SOD datasets to significantly enhance the model's scalability and stability. DRFI [10] autonomously integrates regional saliency features of high dimensionality and selects the most discriminative cues. This inevitably creates challenges for SOD in intricate scenes, for example, backdrops with cluttered or low-contrast areas where color provides few clues.

To address the aforementioned problem, combining RGB and depth features for RGB-D SOD has received increasing attention. To learn the transferable representation of RGB-D partition tasks, Bowen et al. [11] proposed an RGB-D framework, DFormer. DFormer encodes RGB and depth information through a series of RGB-D blocks. The model is pre-trained on ImageNet-1K, so DFormer has the ability to encode RGB-D representations. To build a better global long-range dependence model with self-modality and cross-modality, Cong et al. [12] introduced the transformer architecture to create a new RGB-D SOD network called point-aware interaction and CNN-induced refinement (PICR-Net). The network explores the interaction of characteristics under different modules, alleviates the block effects, and details the destruction problems caused by the transformers. Wu et al. [13] designed HiDAnet, which includes a granularity-based attention strategy to enhance the fusion of RGB and depth features. Note that the accuracy depends greatly on the quality of the depth of information, as suggested by the previous work. Cong et al. [14] suggested a method for assessing the dependability of depth maps and utilizing it to minimize the impact of inferior depth maps on salient detection. DPA-Net [15] can recognize the potential value of depth information through a learning-based approach, preventing contamination by accounting for depth potentiality. Although BBS-Net [16] employs a module with improved depth to selectively extract informative regions of depth cues from both channel and spatial viewpoints, the quality of the depth features is still not great, resulting in the prediction accuracy not achieving adequate results. Although the above models consider the quality of depth features, they only perform single-scale filtering and fuse RGB and depth features at the coarsest filtering level without considering the mode of multi-scale filtering and fusion. This may lead to the roughness of features and the lack of feature utilization and fusion. In addition, Cong et al. [14] adapted a top-down UNet [17] architecture, which performs well in extracting and integrating local information, but it cannot effectively capture global information and has some limitations.

The above facts indicate that multi-scale filtering of depth features and multi-scale fusion with RGB features can improve feature utilization and fusion rates, thereby enhancing a model's accuracy. In addition, a decoder that can capture both global and local information has a significant impact on the performance of a model. Based on this, we propose a depth-quality purification feature processing (DQPFP) network for RGB-D SOD in this paper. Figure 1 shows the overall network architecture. The DQPFP module consists of three key sub-modules, namely a depth denoising module (DDM), depth-quality purification weighting (DQPW) module, and depth purification-enhanced attention (DPEA) module. The DDM filters multi-scale depth features through a channel attention mechanism and a spatial attention mechanism to achieve the initial filtering of the depth features. The DQPW module supplements the color features with purified depth features in a residual-connected manner to enhance feature characterization and then learns the weight factor α from the depth features and RGB features; By assigning smaller weights to poor-quality depth features, we obtain different weight factors on different scales. The DPEA module learns the global attention maps β from the purified depth features, which enhances the quality of the depth features from a spatial dimension. Then, α and β are integrated to obtain the final high-quality depth features. Then, the high-quality depth features and RGB features are fused in a multi-scale manner, and the final saliency map is generated through

a two-stage decoder. In addition, after experimental analysis, we utilize the Randomized Leaky Rectified Linear Unit (RReLU) activation function to prevent overfitting and avoid neuron inactivation, which introduces randomness into the neural network training process. Furthermore, we introduce the pixel position adaptive importance (PPAI) loss, which integrates local structure information to assign different weights to each pixel, thus better guiding the network's learning process and resulting in clearer details.

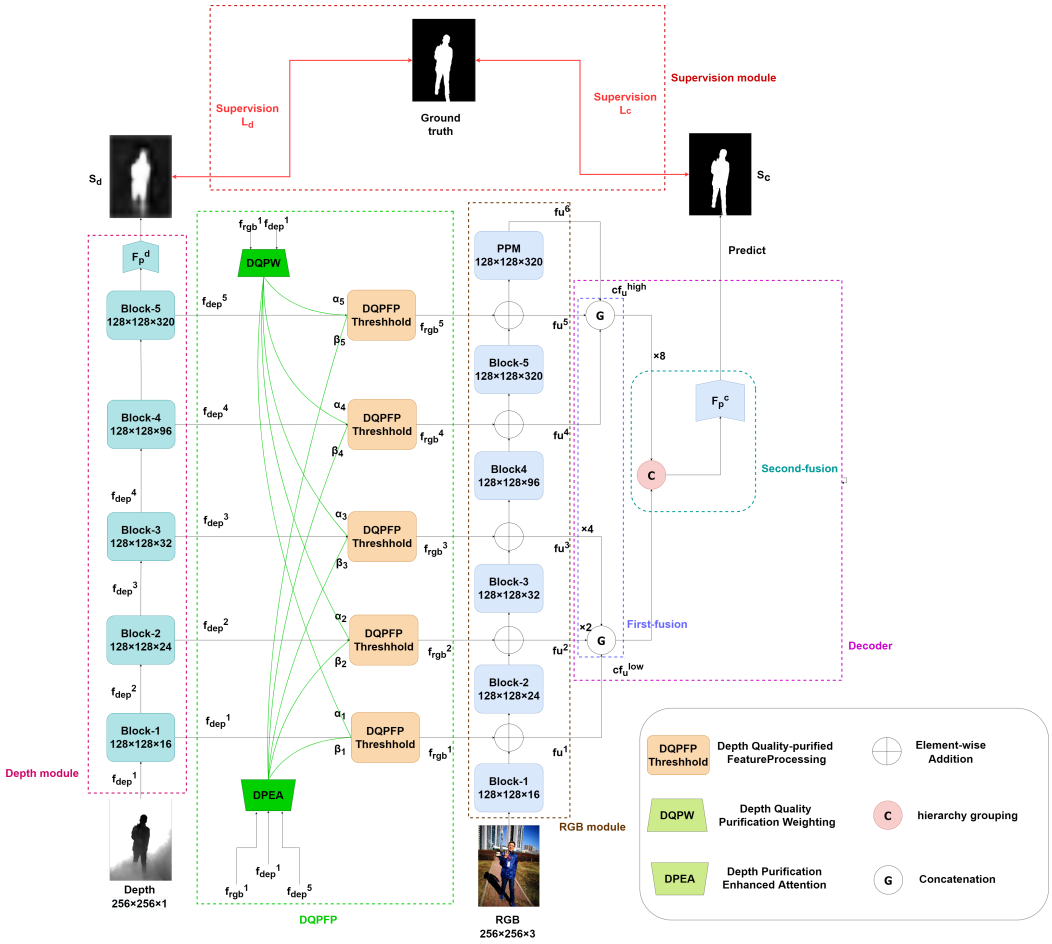

Figure 1. The overall structure of the proposed DQPFPNet.

Our contributions can be summarized as follows:

- We propose a DQPFP module, consisting of three sub-modules: DDM, DQPW, and DPEA. This module filters the depth features in a multi-scale manner and fuses them with RGB features in a multi-scale manner. It can also control and enhance the depth features explicitly in the process of cross-modal fusion, avoiding injecting noise or misleading depth features, which improves the feature utilization, fusion, and accuracy rates of the model.
- We design a dual-stage decoder as one of DQPFPNet's essential elements, which can fully utilize contextual information to improve the modeling ability of the model and enhance the efficiency of the network.
- We introduce the RReLU activation function to prevent overfitting and avoid neuron inactivation, thereby introducing randomness into the training process. Furthermore, the pixel position adaptive importance (PPAI) loss is utilized to integrate local structure information to assign different weights to each pixel, thus better guiding the network's learning process and resulting in clearer details.
- Extensive experiments on six RGB-D datasets demonstrate that DQPFPNet outperforms recent efficient models.

The remainder of this paper is structured as follows. The related research on general RGB-D SOD, effective RGB-D SOD, and depth-quality analysis in RGB-D SOD is covered in Section 2. Section 3 describes the proposed DQPFPNet in detail. Section 4 presents the experimental results, performance evaluation, and ablation analysis. Finally, some conclusions are provided in Section 5.

2. Related Works

For many years, researchers have been investigating the use of RGB-D data for SOD. Considering the objective of this paper, this section reviews common techniques for RGB-D SOD and the previous works on valid methods and depth-quality analysis.

2.1. Common RGB-D SOD Techniques

The effectiveness of traditional methods [18,19] mostly relies on how well made the hand-crafted features are. The first traditional RGB-D SOD method was proposed in 2012. Recently, deep learning-based techniques [20–24] have made great progress, gradually becoming mainstream, with the first deep learning-based RGB-D SOD starting in 2017. To investigate whether and how visual saliency is influenced by depth features, Lang et al. [18] presented the first RGB-D SOD work in 2012, where seven experimenters performed eye-movement experiments on 500 images, recording observation points. A Gauss mixed model was used to simulate the distribution of depth-induced saliency and observe the relationship between 2D saliency and 3D saliency. To investigate the efficacy of global priors for RGB-D data, Peng et al. [19] developed a multi-background contrast model, including local, global, and background contrast, to detect salient targets using depth maps. In addition, the first substantial RGB-D dataset for SOD was provided by this work. In order to accelerate inference speed and improve model training efficiency, GSCINet [21] was proposed with a series of carefully designed convolutions of different scales and attention-to-weight matrices, introducing a cyclic cooperation technique to reduce computing costs while optimizing compressed features, thereby achieving rapid and precise inference for Salient Object Detection. To explore how to combine low-level salient cues to generate master salience maps, DF [20] was created with a new convolutional neural network (CNN) that aggregates many low-level saliency indicators into hierarchical features to effectively find saliency regions in RGB-D images. Published in 2017, it was the first model to incorporate the deep learning technique into RGB-D SOD tasks. In order to make better use of complementary information in multi-modal data and reduce the negative effect of ambiguity between different modes, A2TPNet [24] was proposed to fuse cross-modal features, employing a cooperative technique that combines channel attention and spatial attention mechanisms to lessen the interference of irrelevant information and unimportant aspects in the interaction process. To apply uncertainty to RGB-D Salient Object Detection, UCNet [22], a probability-based RGB-D SOD network that simulates the uncertainty of human annotations through conditional variational automatic encoders, was proposed. In order to fully mine the information of cross-modal complementarity and cross-level continuity, ICNet [23] was proposed, offering a transformation of the information module for interactive high-level feature transformation.

As this research direction has flourished, other encouraging skills have recently been used in RGB-D SOD tasks, for instance, the use of RGB images, bottom-up and top-down depth maps of the multi-modal integration framework [25], co-attention mechanisms [26,27], model compression [28,29], shared networks [30], weak semi-supervised learning [31,32], and self-mutual attention modules [33]. A relatively comprehensive RGB-D SOD survey report can be found in [34].

Although the above-mentioned RGB-D SOD methods can improve detection accuracy, most of the models do not consider the impact of multi-scale depth quality on model accuracy.

2.2. RGB-D SOD Depth-Quality Assessments

As depth quality often affects the performance of a model, some researchers have considered using the RGB-D SOD depth mass to lessen the impact of depth at low mass. To forecast a hint map, in EF-Net [35], a module of a color hint map using RGB pictures was initially employed. The issue of poor-quality depth maps was then resolved, and the saliency detection process was improved thanks to the use of a depth-enhanced module. After removing the depth stream's feature encoder and creating a lightweight model, the authors of SSN [36] employed the depth map directly to guide the pre-fusion of RGB and depth features. The authors of A2dele [37] used network prediction and attention methods as conduits for transferring depth data from the depth stream to the RGB stream. In JL-DCF [30], depth adjustment and fusion mechanisms were used to explicitly solve depth quality issues. Based on this, the adjusted depth map was able to estimate the original depth map. Using hyperpixels of components created by SPSN [38], component prototypes were created from the input RGB picture and depth map. In addition, a reliability selection module was proposed to detect the quality of RGB feature maps and depth feature maps and weigh them adaptively according to the quality of the feature maps.

3. Proposed Method

3.1. Overview

Figure 1 presents the proposed *DQPFPNet* structure, consisting of the encoder, decoder, and supervision module. Our encoder adopts the architecture in [16], where the RGB module is in charge of both cross-module fusion between RGB and depth features and feature extraction for RGB to achieve great performance. To create the final saliency map, the decoder performs a dual-stage fusion, namely the first fusion and second fusion. The encoder itself is made up of an RGB-related module, whose backbone network is MobileNet-v2 [2]; a depth-related module, which is an efficient backbone; and the proposed DQPFP. The depth module and RGB module comprise five feature hierarchies, each with an output stride of 2, with the last one having an output stride of 1. The depth features are extracted within the given hierarchy, passed through the DQPFP threshold, added to the RGB module through simple element additions, and then sent to the next hierarchy. Moreover, a PPM (pyramid pooling module [39]) is introduced toward the end of the RGB module to acquire multi-scale semantic data. In practical coding, the DQPFP threshold consists of two operations: depth-quality purification weighting (DQPW) and depth purification-enhanced attention (DPEA). In order to facilitate a better understanding of the overall workflow of the network, Figure 2 shows the pipeline of the entire network.

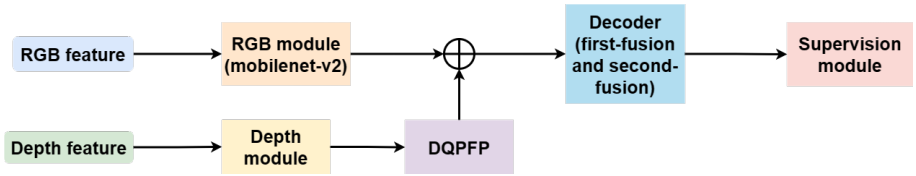

Figure 2. The pipeline of the network architecture.

The features extracted from the five depths/RGB hierarchies are represented as $f_n^i (n \in \{rgb, dep\}, i = 1, \ldots, 5)$, the fusion features are represented as $f_u^i (i = 1, \ldots, 5)$, and the features from the PPM are represented as f_u^6. This multi-modal feature fusion can be written as:

$$f_u^i = f_{rgb}^i + (\alpha_i \otimes \beta_i \otimes f_{dep}^i) \tag{1}$$

where α_i and β_i are calculated by DQPW and DPEA, respectively, to control the fusion of the depth features f_{dep}^i. $\otimes$ indicates element-by-element multiplication. After the encoding process shown in Figure 1, $f_u^i (i = 1, \ldots, 5)$ and f_u^6 are transferred to the next decoder module.

3.2. Depth-Quality Purification Feature Processing (DQPFP)

DQPFP includes two crucial modules: DQPW (depth-quality purification weighting) and DPEA (depth purification-enhanced attention). These two modules calculate α_i and β_i in Equation (1), respectively. $\alpha_i \in \mathbb{R}^i$ is a scalar that determines "how many" depth features are used, whereas $\beta_i \in \mathbb{R}^{s \times s}$ (s is the feature size for level i) is a spatial attention map, determining "which regions" to focus on within the depth characteristics. The internal structures of the DQPW and DPEA modules are described below.

3.2.1. Depth-Quality Purification Weighting (DQPW)

The paired color features and depth features in the RGB-D features are two different forms of the same object. Color images provide visual cues, and depth images provide 3D information. Considering the inadequate quality of depth maps, this paper proposes a depth de-noising module (DDM). The DDM first purifies the depth features using the attention mechanism, then complements the color features through a residual connection [40], and uses the shortcut connection section to retain more of the original color cues.

In the DDM, as shown in Figure 3, the RGB features are merged with the depth features and transmitted to the channel attention module to obtain the attention channel mask, which is employed to purify the depth features. Subsequently, the purified depth features are input into the spatial attention module to produce the attention space mask, purifying the depth features on a spatial level. This process can be represented as:

$$F_i^r = f_i^d \times SA(f_i^d \times CA(Cat(f_d^d, f_i^r))) + f_i^r \tag{2}$$

where f_i^r and f_i^d, respectively, represent the low-level color and depth features; Cat($\cdot$) represents the concatenation and subsequent convolution operations; CA($\cdot$) and SA($\cdot$) are channel and spatial attention operations proposed by CBAM [41], respectively; "$\times$" denotes the element-wise multiplication operation; and "$+$" denotes the element-by-element addition operation. This process purifies poor-quality depth features and then merges them into RGB features to produce a more accurate representation F_i^r.

In Figure 4, the low-level features f_{rgb}^1 and f_{dep}^1 first obtain $f_{rgb\_en}^1$ through the DDM, and DQPW adaptively learns the weighting term α_i from the features $f_{rgb\_en}^1$ and f_{dep}^1. We apply convolution to $f_{rgb\_en}^1 / f_{dep}^1$ to obtain the transformed features $f_{rt'} / f_{dt'}$, which are anticipated to obtain more activators associated with the edge:

$$f_{rt'} = \mathbf{BRRConv}_{1\times1}(f_{rgb\_en}^1), f_{dt'} = \mathbf{BRRConv}_{1\times1}(f_{dep}^1) \tag{3}$$

where $BRRConv_{1\times1}(\cdot)$ represents a 1×1 convolution with BatchNorm layers and the RReLU activation. To be able to assess the alignment of low-level features, the alignment feature vector V_{BA}, encoding the alignment between $f_{rt'}$ and $f_{dt'}$, is computed as follows, given the edge activations $f_{rt'}$ and $f_{dt'}$:

$$V_{BA} = \frac{\mathbf{GAP}(f_{rt'} \otimes f_{dt'})}{\mathbf{GAP}(f_{rt'} + f_{dt'})} \tag{4}$$

where $GAP(\cdot)$ means the global average pooling operation aggregating element-level details and $\otimes$ represents the element-level multiplication.

Additionally, to make V_{BA} robust to minor edge movements, this paper calculates V_{BA} on multiple scales and concatenates the results to produce the strengthened vector. Figure 4 shows that this multi-level computation is realized by downsampling the original features $f_{rt'} / f_{dt'}$ by max-pooling with a stride of 2, and then V_{BA}^1 and V_{BA}^2 are calculated in the same way as in Equation (4). Assuming that V_{BA}, V_{BA}^1, and V_{BA}^2 are aligned eigenvectors

calculated from the three scales shown in Figure 4, the strengthened vector V_{BA}^{ms} is calculated as follows:

$$V_{BA}^{ms} = [V_{BA}, V_{BA}^{1}, V_{BA}^{2}] \tag{5}$$

where $[\cdot\,]$ represents a channel cascade. Then, two completely linked layers are used to calculate $\alpha \in \mathbb{R}^5$ from V_{BA}^{ms} in the manner shown below:

$$\alpha = \mathbf{MLP}(V_{BA}^{ms}) \tag{6}$$

where $MLP(\cdot)$ represents a two-level perception with the Sigmoid function at the end. Then, $\alpha_i \in (0,1)(i = 1,2\ldots,5)$ is one of the elements of the vector α that is obtained. Note that this paper uses different weighting factors for different levels, and the effectiveness of this multivariable approach is verified in Section 4.4.

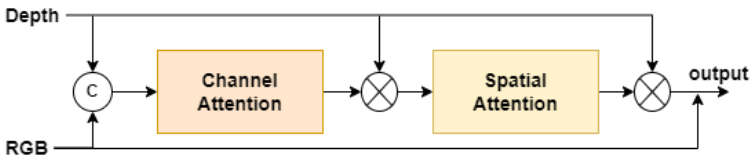

Figure 3. The structure of the DDM.

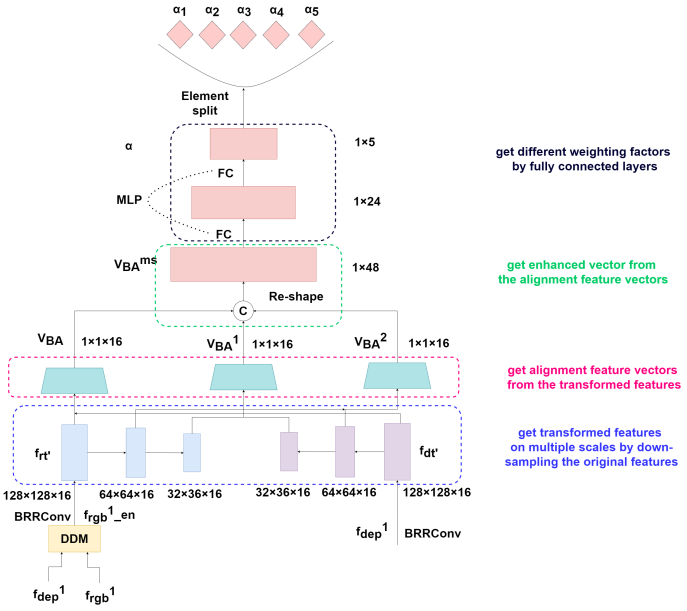

Figure 4. The organization of the DQPW module. The red arrows show the Equation (4) calculation process. The dashed lines indicate max-pooling with a stride of 2.

3.2.2. Depth Purification-Enhanced Attention (DPEA)

The DPEA enhances the depth features in the spatial dimension by deriving a global attention map β_i from the depth channel. As shown in Figure 5, the purified features $f_{dep}^{5}\_en$ are first obtained from f_{rgb}^{1} and f_{dep}^{5} through the DDM to locate the coarse-grained salient areas (with supervision cues shown in Figure 1). In order to simplify the next pixel-by-pixel processes, $f_{dep}^{5}\_en$ is compressed and then sampled up into f_{dht} in the same dimension as f_{rgb}^{1}/f_{dep}^{1}, as shown in the following formula:

$$f_{dht} = \mathbf{F}_{UP}^{8}(\mathbf{BRRConv}_{1\times1}(f_{dep}^{5})) \tag{7}$$

where $\mathbf{F}_{UP}^{8}(\cdot)$ represents 8 × bilinear upsampling. f_{dht} is then re-calibrated with the primary RGB and depth features. Like the calculation in DQPW, this paper first transfers f_{rgb}^{1}/f_{dep}^{1} to $f_{rt''}/f_{dt''}$. The result is that element-level multiplication generates the features f_{ec}, which somewhat emphasizes the general activation properties linked to the edge. The max-pooling operation and dilated convolution operation are used to rapidly expand the receptive field to simulate better long-term relationships between low- and high-level information (i.e., f_{ec} and f_{dht}) while preserving the effectiveness of the DPEA. This re-calibration process is represented as:

$$\mathbf{F}_{rec}(f_{dht}) = \mathbf{F}_{UP}^{2}\left(\mathbf{DConv}_{3\times3}(\mathbf{F}_{DN}^{2}(f_{dht} + f_{ec}))\right) \tag{8}$$

where $\mathbf{F}_{rec}(\cdot)$ is the input of the re-calibration process; $\mathbf{DConv}_{3\times3}(\cdot)$ represents the 3 × 3 dilated convolution with a stride of 1 and a dilation rate of 2, followed by BatchNorm layers and the RReLU activation; and $\mathbf{F}_{UP}^{2}(\cdot)/\mathbf{F}_{DN}^{2}(\cdot)$ indicates the bi-linear upsampling/downsampling operation to $2/(\frac{1}{2})$ times the initial dimensions. To achieve a balance between functionality and effectiveness, the following two re-calibrations are performed:

$$f'_{dht} = \mathbf{F}_{rec}(f_{dht}), f''_{dht} = \mathbf{F}_{rec}(f'_{dht}), \tag{9}$$

where f'_{dht} and f''_{dht} are the features re-calibrated once and twice, respectively. Finally, f''_{dht} is combined with f_{ec} to obtain global attention maps:

$$\beta = \mathbf{BRRConv}_{3\times3}(f_{ec} + f''_{dht}). \tag{10}$$

Be aware that the RReLU activation in $\mathbf{BRRConv}_{3\times3}$ is replaced with the Sigmoid activation to achieve the attention features of β. Eventually, By downsampling β, five depth global attention maps $\beta_1, \beta_2, \ldots, \beta_5$ are obtained, using spatial enhancement factors for the depth levels. Generally, background clutters that are unrelated to the depth features can be prevented by multiplying them with attention maps $\beta_1 \sim \beta_5$.

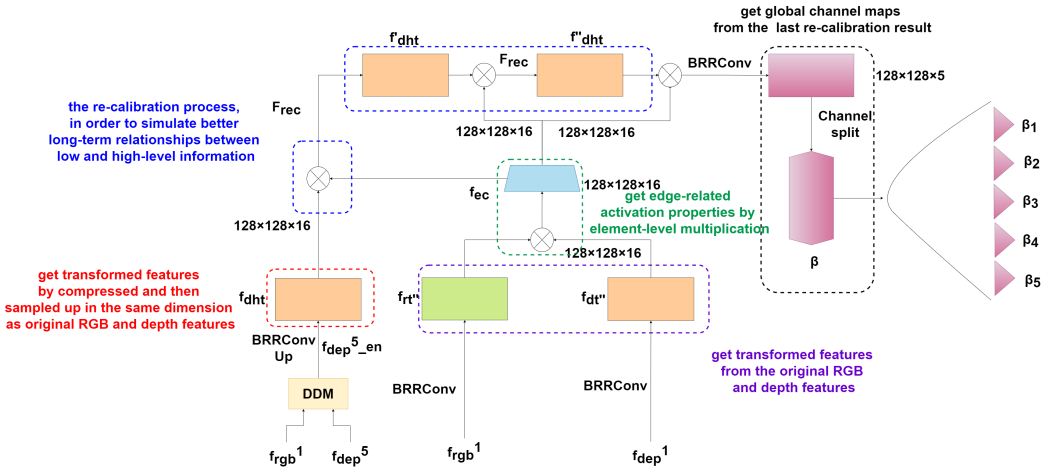

Figure 5. The structure of the DPEA (depth purification-enhanced attention) module.

3.3. Dual-Stage Decoder

This work suggests a simpler two-phase decoder that comprises first fusion and second fusion stages to further increase efficiency, in contrast to the well-known UNet [17], which uses a hierarchical top-down decoding technique. Hierarchical grouping is used, denoted in Figure 1 as "G". The first fusion aims to cut down on the feature channels and hierarchies. Based on the outputs of the first fusion stage, the low-level and high-level hierarchical structures are further aggregated to generate the final salient map. Note that in

our decoder, instead of ordinary convolutions, separable depth-wise convolutional filters are mainly used with many input channels.

3.3.1. First Fusion Stage

This paper first uses a 3×3 depth-by-depth separable convolution [42] with BatchNorm layers and the RReLU activation, represented as $\mathbf{DSConv}_{3\times3}(\cdot)$, to reduce the encoder's features during compression $(f_u^i, i = 1, 2 \ldots 6)$ into an integrated channel of size 16. Then, the popular channel attention operator [43] $\mathbf{F}_{CA}(\cdot)$ is used to improve the characteristics through channel weighting. The procedure described above can be expressed as:

$$cf_u^i = \mathbf{F}_{CA}(\mathbf{DSConv}_{3\times3}(f_u^i)), \tag{11}$$

where cf_u^i represents the features from the compression and enhancement processes. This work, which is motivated by [16], splits the six feature hierarchies into both high-level and low-level hierarchies, as follows:

$$cf_u^{low} = \sum_{i=1}^{3} \mathbf{F}_{UP}^{2^{i-1}}(cf_u^i), cf_u^{high} = \sum_{i=4}^{6} cf_u^i, \tag{12}$$

where $\mathbf{F}_{UP}^i$ is i times the original size of the bilinear upsampling.

3.3.2. Second Fusion Stage

Since the number of channels and hierarchies have been reduced in the first fusion phase, the high-level and low-level hierarchies are directly concatenated in the second fusion phase and then provided to a prediction head to acquire the ultimate full-resolution prediction map, which is expressed as follows:

$$S_c = \mathbf{F}_p^c\left([cf_u^{low}, \mathbf{F}_{UP}^8(cf_u^{high})]\right), \tag{13}$$

where S_c represents the final salient features, and $\mathbf{F}_p^c(\cdot)$ represents the prediction head consisting of two 3×3 separable depth-by-depth convolutions (followed by BatchNorm layers and the RReLU activation function): a 3×3 convolution with Sigmoid activation and a $2\times$ bilinear upsampling to restore the original input dimension.

3.4. RReLU Activation Function

The activation function plays an important role in computer vision tasks such as object segmentation, object tracking, and object detection. An important aspect of neural network design is the selection of the activation functions to be used in the different layers of the network. The activation function is used to introduce nonlinearity into the neural network calculation, and the correct selection of the activation function is very important for the effective performance of the network.

Common activation functions, such as Sigmoid, Tanh, and so on, have good properties, but with the advent of deep neural architectures, it is difficult for researchers to train very deep neural networks because they are saturated with activation functions. To solve this problem, the ReLU activation function was utilized, as shown in Figure 6. Although ReLU is not differentiable at zero, it is unsaturated, and it can keep the gradient constant in the positive interval. This method effectively alleviates the problem of gradient disappearance in the neural network, thereby speeding up the training of the neural model. However, when the input is negative, ReLU will have dead neurons, resulting in the corresponding weights not being updated, which may result in the loss of model information.

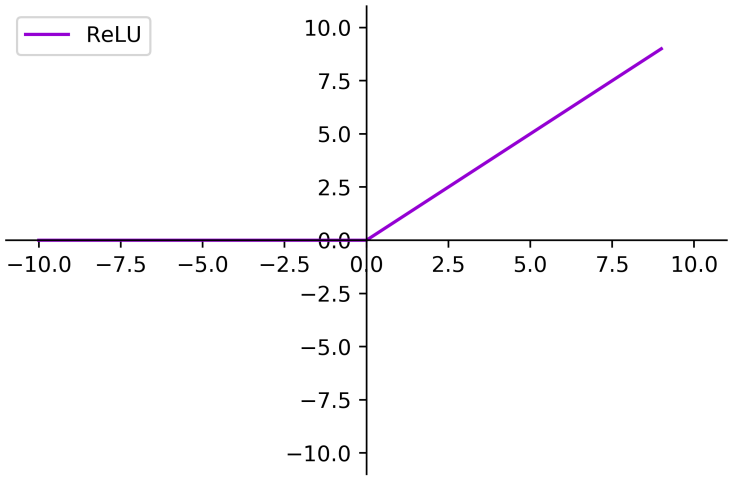

Figure 6. The ReLU activation function.

To address the problems with the ReLU activation function, in Section 4.4, we conduct a number of experiments to determine the optimal activation function to use in this model: RReLU. As shown in Figure 7, RReLU is a variant of ReLU that prevents overfitting by introducing randomness during model training while helping to resolve the issue of neuronal inactivation. When the input is positive, the gradient is a positive value, and when the input is negative, the gradient is a negative value. However, the slope of the negative value is randomly obtained during training and fixed in subsequent tests.

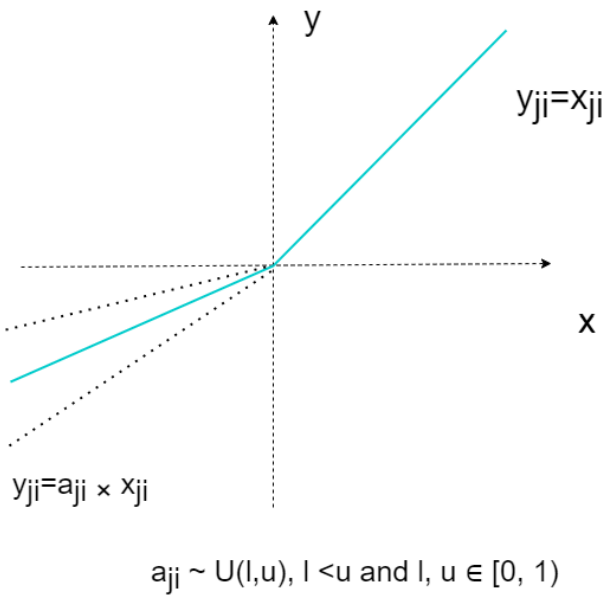

Figure 7. The RReLU activation function.

The beauty of RReLU is that during the training process, a_{ji} is randomly drawn from a uniform distribution of $U(l, u)$, which helps increase the robustness of the model and reduce the dependence on specific input patterns, thereby mitigating the risk of overfitting. By introducing randomness, RReLU allows the activation values of neurons to vary within a range, even with negative inputs, thus avoiding complete neuronal inactivation.

3.5. Pixel Position Adaptive Importance (PPAI) Loss

Despite having three flaws, binary cross-entropy (BCE) is the most popular loss function for RGB and RGB-D SOD. First, it disregards the image's overall structure and calculates each pixel's loss separately. Second, the loss of foreground pixels will be less

noticeable in photographs where the backdrop predominates. Third, it gives each pixel the same treatment. In actuality, pixels in cluttered or constrained locations (e.g., the pole and horn) are more likely to result in incorrect predictions and require additional effort, whereas pixels located in places like roadways and trees require less focus. So, this paper introduces the pixel position adaptive importance (PPAI) loss, which consists of two components, namely the weighted binary cross-entropy (wBCE) loss and the weighted IoU (wIoU) loss. The wBCE loss is shown in Equation (11)

$$L^s_{wbce} = -\frac{\sum_{i=1}^{H}\sum_{j=1}^{W}\left(1+\gamma\alpha_{ij}\right)\sum_{l=0}^{1}\mathbf{1}\left(g^s_{ij}=l\right)\log\mathbf{Pr}\left(p^s_{ij}=l\mid\Psi\right)}{\sum_{i=1}^{H}\sum_{j=1}^{W}\gamma\alpha_{ij}} \tag{14}$$

where $\mathbf{1}(\cdot)$ is the indicator function and γ is a hyperparameter. The symbol $l \in \{0,1\}$ denotes two types of labels. p^s_{ij} and g^s_{ij} are the prediction and the ground truth of the pixel at location (i,j) in an image. Ψ shows all the parameters of the model, and $\mathbf{Pr}(p^s_{i,j}=l\mid\Psi)$ represents the predicted probability.

In L^s_{wbce}, each pixel is given a weight α. A hard pixel corresponds to a larger α, whereas a simple pixel is assigned a smaller weight. α, which is determined based on the disparity between the central pixel and its surrounds, can be used as a measure of pixel significance, as shown in Equation (15).

$$\alpha^s_{ij} = \left|\frac{\sum_{m,n\in A_{ij}} gt^s_{mn}}{\sum_{m,n\in A_{ij}} 1} - gt^s_{ij}\right| \tag{15}$$

where A_{ij} denotes the area around the pixel (i,j). For all pixels, $\alpha^s_{ij} \in [0,1]$. If α^s_{ij} is big, the pixel at (i,j) is significant (e.g., an edge or hole) and stands out significantly from its surroundings. Therefore, it warrants extra attention. In contrast, if α^s_{ij} is small, the pixel is just an ordinary pixel and not worth attention.

L^s_{wbce} increases the emphasis on hard pixels compared to BCE. Meanwhile, the local structural information is encoded into L^s_{wbce} such that a greater receptive field rather than a single pixel is the model's primary focus. To further make the network focus on the overall structure, the weighted IoU (wIoU) loss is introduced, as shown in Equation (16).

$$L^s_{\text{wiou}} = 1 - \frac{\sum_{i=1}^{H}\sum_{j=1}^{W}\left(gt^s_{ij} * p^s_{ij}\right) * \left(1+\gamma\alpha^s_{ij}\right)}{\sum_{i=1}^{H}\sum_{j=1}^{W}\left(gt^s_{ij} + p^s_{ij} - gt^s_{ij} * p^s_{ij}\right) * \left(1+\gamma\alpha^s_{ij}\right)} \tag{16}$$

In the segmentation of images, the IoU loss is frequently employed. It is not affected by the uneven distribution of pixels, and the optimization of the global structure is the goal, which overcomes the limitation of a single pixel. In recent years, it has been included in SOD in order to address BCE's deficiencies. However, it still treats each pixel equally and ignores the differences between pixels. In contrast to the IoU loss, our WIoU loss gives harder pixels a higher weight to indicate their significance.

The pixel position adaptive importance (PPAI) loss is shown in Equation (14). It combines the information on local structures to assign different weights to each pixel and provide pixel restriction (L^s_{wbce}) and global restriction L^s_{wiou}, thus better guiding the network learning process and resulting in clearer details.

$$L^s_{ppai} = L^s_{wbce} + L^s_{wiou} \tag{17}$$

Eventually, the ultimate loss $\mathcal{L}^s_{c-ppai}$ and deep supervision for the loss of the depth branch $\mathcal{L}_d$ make up the total loss $\mathcal{L}$, which is formulated as follows:

$$\mathcal{L} = \mathcal{L}^s_{c-ppai}(S_c, G) + \mathcal{L}_d(S_d, G), \tag{18}$$

where G represents the ground truth (GT) and $\mathcal{L}^{s}_{c-ppai}$ and $\mathcal{L}_{d}$ denote the PPAI loss and the standard BCE loss, respectively.

4. Experiments and Results

This section introduces the datasets and metrics, the details of the implementation, and comparisons to SOTAs. The experiments include both quantitative and qualitative experiments. Ablation experiments are also conducted to demonstrate the effectiveness of our proposed module.

4.1. Datasets and Metrics

Experiments were performed on six public datasets, including *LFSD* [44] (100 samples), *NJU2K* [45] (1996 samples), *NLPR* [46] (1023 samples), *RGBD*135 [47] (142 samples), *SIP* [48] (910 samples), and *STERE* [49] (1000 samples).

Meanwhile, for evaluation, four widely used metrics were employed, including the S-measure (S_α) [50], maximum F-measure (F^m_β) [51], maximum E-measure (E^m_ε) [52,53], and mean absolute error (MAE, $\mathcal{M}$) [48]. A higher $S_\alpha, F^m_\beta, and\ E^m_\varepsilon$ and a lower $\mathcal{M}$ mean better performance.

4.2. Details of the Implementation

The experiments were carried out on a personal computer equipped with an Intel (R) Xeon (R) Gold 6248 CPU and an NVIDIA Tesla V100-SXM2 32GB GPU. DQPFPNet was implemented in Pytorch [54], and the RGB and depth features were both scaled to 256×256 as input. To extend the network to the limited training examples, following [16], this paper adopted a variety of data enhancement techniques, such as horizontal flipping, random cropping, color enhancement, etc. DQPFPNet was trained on a single Tesla v100 GPU for 300 epochs. The Adam optimizer's [55] initial learning rate was set to 1×10^{-4} with a batch size of 10. A multiple learning rate strategy was used, with the power set to 0.9.

4.3. Comparison to SOTAs

A total of 1700 samples from NJU2K and 800 samples from NLPR were used for training, and tests were performed on STERE, SIP, NLPR, LFSD, NJU2K, and RGBD135. The results of DQPFPNet were compared to those of 16 state-of-the-art (SOTA) models, including C2DF [56], S2MA [33], JL-DCF [30], CoNet [57], UCNet [22], CIRNet [58], SSLSOD [59], cmMS [60], DANet [36], DCF [61], ATSA [62], DSA2F [63], PGAR [64], A2dele [37], MSal [65], and DFMNet [66], as shown in Table 1. The salient maps for the other models were derived from their released predictions, if available, or produced from their public code.

As shown in Table 1, DQPFPNet outperformed some existing efficient models in terms of detection accuracy, e.g., MSal [65], A2dele [37], and PGAR [64]. Additionally, it is evident that DQPFPNet achieved SOTA performance, indicating that the method of filtering the depth features in a multi-scale manner, fusing the filtered depth features with RGB features in a multi-scale manner, and finally, obtaining the salient graph through a two-stage decoder is of practical significance, thereby proving the effectiveness of our model. Validation of the functionality of each module is performed in Section 4.4. Figure 8 presents a visual comparison of the results of our proposed method and those of the SOTA methods, and our results are closer to the GT.

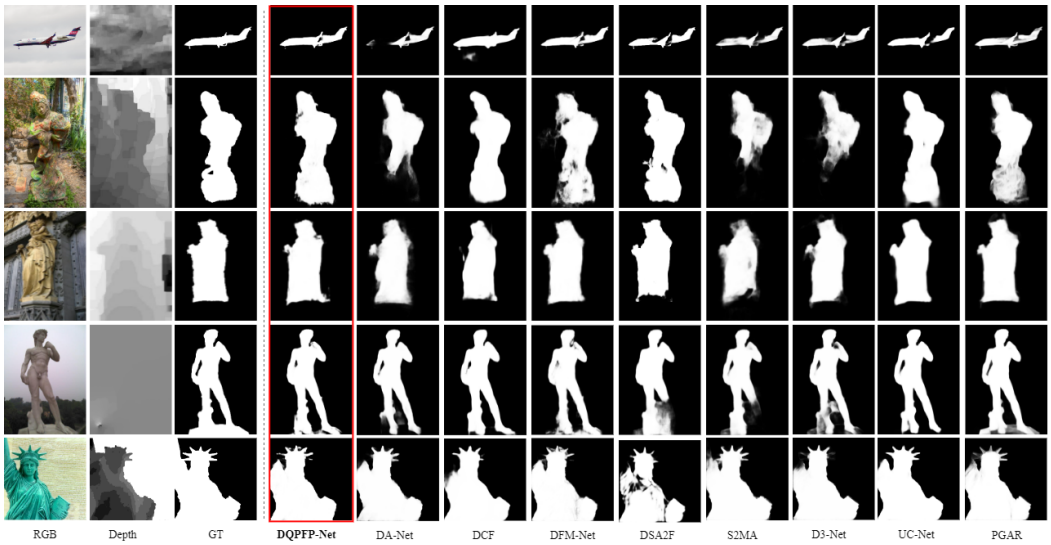

Figure 8. Qualitative comparison of DQPFPNet with SOTA RGB-D SOD methods.

4.4. Ablation Experiments

Thorough ablation experiments were performed on six classical datasets, including STERE, SIP, RGBD135, NLPR, LFSD, and NJU2K, by changing or deleting parts of the DQPFPNet implementation.

4.4.1. Effectiveness of DQPFP

DQPFP is made up of two essential components: DQPW and DPEA. Table 2 displays several configurations with DQPW/DPEA disabled. Specifically, configuration #1 represents the baseline model with DQPW and DPEA removed from DQPFPNet. Configurations #2 and #3 each introduce one of the components, whereas configuration #4 represents the complete model of DQPFPNet. It can be seen from Table 2 that merging DQPW and DPEA into the baseline model resulted in consistent improvements on almost all datasets. Meanwhile, when comparing configurations #2/#3 to #4, it can be seen that using DQPW and DPEA together further improved the results, demonstrating a synergistic effect between DQPW and DPEA. The possible reason is that although DPEA can enhance potential salient areas in the deep dimension, it is inevitable that certain errors (for example, emphasizing the wrong areas) will occur, especially in the case of poor depth quality. Fortunately, DQPW mitigates some of these mistakes because it allocates lower global weights to the depth features in this case. Hence, the two elements can cooperate to increase network resiliency, as mentioned in Section 3.2.

Figure 9 shows visual examples of configuration #3 (without DQPW) and configuration #4. Figure 9a,b illustrate that combining DQPW contributes to improved detection accuracy. In the first example of good quality (row 1, Figure 9a), in the RGB view, it is challenging to discern between shadows and people's legs, but this is simple to do in the depth view. The addition of DQPW enhances the depth feature and makes it easier to distinguish the full human body from the shadow. In the first example of bad quality (row 1, Figure 9b), although the boy on the skateboard boy much more blurry in the depth view, the impact of the incorrect depth is lessened, and precise detection of the entire object is still possible.

Table 1. Quantitative benchmark results. ↑/↓ for a metric denotes that a larger/smaller value is better. Our results are highlighted in **bold**. The best scores are shown in red. The second-best scores are shown in blue.

Metric		C2DF TMM 2022	JL-DCF CVPR 2020	UCNet CVPR 2020	SSLSOD AAAI 2022	S2MA CVPR 2020	CoNet ECCV 2020	cmMS ECCV 2020	DANet ECCV 2020	ATSA ECCV 2020	DCF CVPR 2022	DSA2F CVPR 2021	A2dele CVPR 2020	PGAR ECCV 2020	MSal TPAMI 2021	DFMNet CVPR 2022	CIRNet TIP 2022	DQPFPNet Ours -
SIP	$S_\alpha \uparrow$	0.871	0.879	0.875	0.870	0.878	0.858	0.867	0.878	0.864	0.876	0.862	0.829	0.875	0.873	0.873	0.861	**0.885**
	$F_\beta^m \uparrow$	0.865	0.885	0.879	0.862	0.884	0.867	0.871	0.884	0.873	0.884	0.875	0.834	0.877	0.883	0.878	0.840	**0.896**
	$E_\varepsilon^m \uparrow$	0.912	0.923	0.919	0.900	0.920	0.913	0.910	0.920	0.911	0.922	0.912	0.889	0.914	0.920	0.919	0.886	**0.943**
	$\mathcal{M} \downarrow$	0.053	0.051	0.051	0.059	0.054	0.063	0.061	0.054	0.058	0.052	0.057	0.070	0.059	0.053	0.055	0.069	**0.046**
NLPR	$S_\alpha \uparrow$	0.927	0.925	0.920	0.914	0.915	0.908	0.915	0.915	0.907	0.924	0.919	0.890	0.918	0.920	0.923	0.920	**0.931**
	$F_\beta^m \uparrow$	0.904	0.916	0.903	0.881	0.902	0.887	0.896	0.903	0.876	0.912	0.906	0.875	0.898	0.908	0.907	0.881	**0.930**
	$E_\varepsilon^m \uparrow$	0.955	0.962	0.956	0.941	0.950	0.945	0.949	0.953	0.945	0.963	0.952	0.937	0.948	0.961	0.956	0.937	**0.961**
	$\mathcal{M} \downarrow$	0.021	0.022	0.025	0.027	0.030	0.031	0.027	0.029	0.028	0.022	0.024	0.031	0.028	0.025	0.026	0.028	**0.022**
NJU2K	$S_\alpha \uparrow$	0.908	0.903	0.897	0.902	0.894	0.895	0.900	0.891	0.901	0.904	0.895	0.868	0.906	0.905	0.904	0.901	**0.906**
	$F_\beta^m \uparrow$	0.898	0.903	0.895	0.887	0.889	0.892	0.897	0.880	0.893	0.906	0.897	0.872	0.905	0.905	0.905	0.880	**0.910**
	$E_\varepsilon^m \uparrow$	0.936	0.944	0.936	0.929	0.930	0.937	0.936	0.932	0.921	0.950	0.936	0.914	0.940	0.942	0.945	0.917	**0.947**
	$\mathcal{M} \downarrow$	0.038	0.043	0.043	0.043	0.053	0.047	0.044	0.048	0.040	0.040	0.044	0.052	0.045	0.041	0.041	0.047	**0.036**
RGBD135	$S_\alpha \uparrow$	0.898	0.929	0.934	0.905	0.941	0.910	0.932	0.904	0.907	0.905	0.917	0.884	0.894	0.929	0.932	0.900	**0.941**
	$F_\beta^m \uparrow$	0.885	0.919	0.930	0.883	0.935	0.896	0.922	0.894	0.885	0.894	0.916	0.873	0.879	0.924	0.924	0.888	**0.942**
	$E_\varepsilon^m \uparrow$	0.946	0.968	0.976	0.941	0.973	0.945	0.970	0.957	0.952	0.951	0.954	0.920	0.929	0.970	0.969	0.927	**0.976**
	$\mathcal{M} \downarrow$	0.031	0.022	0.019	0.025	0.021	0.029	0.020	0.029	0.024	0.024	0.023	0.030	0.032	0.021	0.020	0.051	**0.019**
LFSD	$S_\alpha \uparrow$	0.863	0.862	0.864	0.859	0.837	0.862	0.849	0.845	0.865	0.842	0.883	0.834	0.833	0.847	0.863	0.822	**0.871**
	$F_\beta^m \uparrow$	0.859	0.866	0.864	0.867	0.835	0.859	0.869	0.846	0.862	0.842	0.889	0.832	0.831	0.841	0.864	0.803	**0.871**
	$E_\varepsilon^m \uparrow$	0.897	0.901	0.905	0.900	0.873	0.907	0.896	0.886	0.905	0.883	0.924	0.874	0.893	0.888	0.902	0.834	**0.906**
	$\mathcal{M} \downarrow$	0.065	0.071	0.066	0.066	0.094	0.071	0.074	0.083	0.064	0.075	0.055	0.077	0.093	0.078	0.071	0.096	**0.065**
STERE	$S_\alpha \uparrow$	0.899	0.905	0.903	0.893	0.890	0.908	0.895	0.892	0.897	0.902	0.898	0.885	0.903	0.903	0.898	0.835	**0.904**
	$F_\beta^m \uparrow$	0.891	0.901	0.899	0.890	0.882	0.904	0.891	0.881	0.884	0.901	0.900	0.885	0.893	0.895	0.891	0.847	**0.901**
	$E_\varepsilon^m \uparrow$	0.938	0.946	0.944	0.936	0.932	0.948	0.937	0.930	0.921	0.945	0.942	0.935	0.936	0.940	0.942	0.911	**0.947**
	$\mathcal{M} \downarrow$	0.046	0.042	0.039	0.044	0.051	0.040	0.042	0.048	0.039	0.039	0.039	0.043	0.044	0.041	0.044	0.066	**0.040**

Table 2. Ablation analysis of DQPFP to validate the effectiveness of DQPW and DPEA. √ below the module indicates that the model has used the module. Otherwise, the model has not used it. The best results are shown in red.

#	DQPW	DPEA	SIP				NLPR				NJU2K				RGBD135				LFSD				STERE			
			S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$
1			0.873	0.879	0.919	0.054	0.912	0.899	0.954	0.027	0.898	0.903	0.941	0.042	0.926	0.931	0.971	0.017	0.850	0.853	0.891	0.075	0.885	0.883	0.938	0.047
2	√		0.877	0.885	0.923	0.051	0.916	0.905	0.958	0.025	0.941	0.902	0.898	0.042	0.941	0.941	0.968	0.016	0.853	0.857	0.895	0.074	0.885	0.887	0.940	0.046
3		√	0.876	0.883	0.923	0.051	0.914	0.901	0.954	0.025	0.897	0.903	0.941	0.043	0.934	0.931	0.976	0.018	0.855	0.856	0.895	0.073	0.889	0.886	0.940	0.045
4	√	√	0.885	0.896	0.923	0.046	0.922	0.916	0.961	0.023	0.904	0.910	0.947	0.039	0.930	0.942	0.976	0.019	0.870	0.869	0.906	0.068	0.902	0.898	0.947	0.041

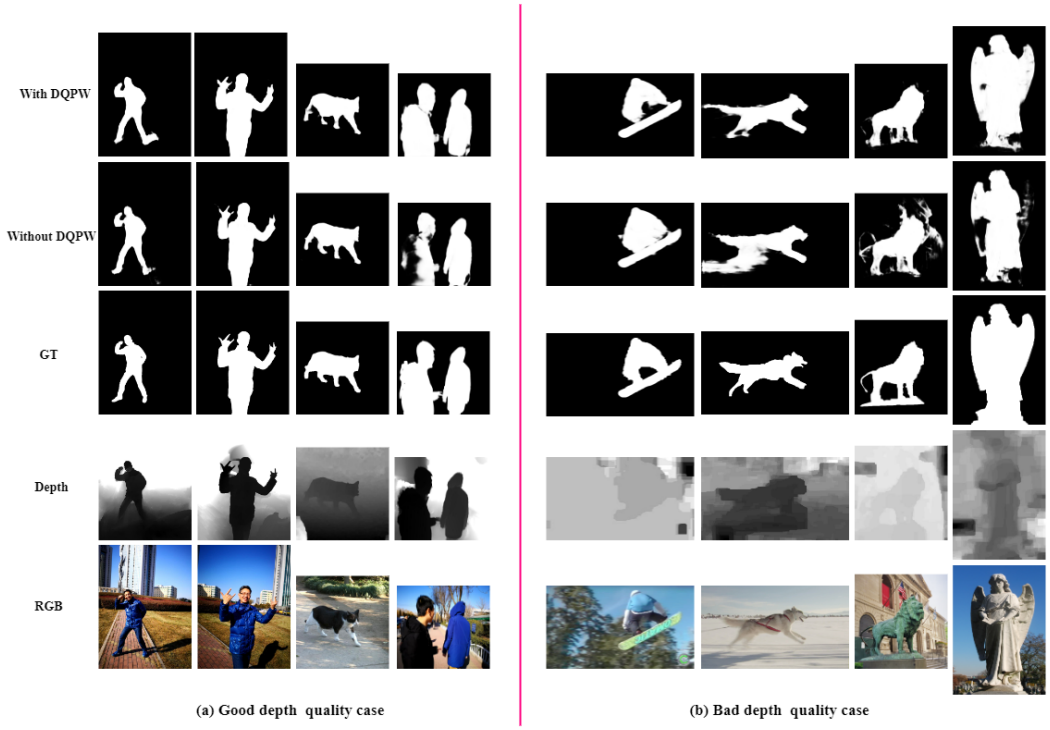

Figure 9. Visual examples of configuration #3 (without DQPW) and configuration #4 (with DQPW) for good (**a**) and bad (**b**) depth-quality cases.

Table 3 presents the results of the modular ablation experiments, demonstrating the positive effect of each module on detection accuracy. The baseline is the original model, with its precision as the benchmark. The modules are presented in order from the second to the fourth columns, with all other conditions remaining unchanged. In addition, all the experimental parameter configurations remained the same. Based on the detection outcomes, it is evident that the combination of the DQPFP module, RReLU activation function, and PPAI loss can greatly increase the model's detection accuracy.

Table 3. Quantitative module results. ↑/↓ for a metric denotes that a larger/smaller value is better. The best scores are shown in red.

Metric	Baseline	Baseline + DQPFP	Baseline + DQPFP + RReLU	Baseline + DQPFP + RReLU + PPAI
SIP $S_\alpha \uparrow$	0.8732	0.8751	0.8796	0.8850
SIP $F_\beta^m \uparrow$	0.8779	0.8816	0.8874	0.8960
SIP $E_\varepsilon^m \uparrow$	0.9191	0.9249	0.9372	0.9425
SIP $\mathcal{M} \downarrow$	0.0552	0.0515	0.0506	0.0460
NLPR $S_\alpha \uparrow$	0.9233	0.9265	0.9277	0.9311
NLPR $F_\beta^m \uparrow$	0.9074	0.9078	0.9111	0.9300
NLPR $E_\varepsilon^m \uparrow$	0.9562	0.9577	0.9583	0.9612
NLPR $\mathcal{M} \downarrow$	0.0258	0.0249	0.0244	0.0221
NJU2K $S_\alpha \uparrow$	0.9041	0.9042	0.9051	0.9066
NJU2K $F_\beta^m \uparrow$	0.9052	0.9061	0.9075	0.9100
NJU2K $E_\varepsilon^m \uparrow$	0.9456	0.9458	0.9455	0.9467
NJU2K $\mathcal{M} \downarrow$	0.0418	0.0411	0.0406	0.0364
RGBD135 $S_\alpha \uparrow$	0.9321	0.9325	0.9340	0.9411
RGBD135 $F_\beta^m \uparrow$	0.9241	0.9262	0.9277	0.9423
RGBD135 $E_\varepsilon^m \uparrow$	0.9690	0.9715	0.9738	0.9761
RGBD135 $\mathcal{M} \downarrow$	0.0207	0.0205	0.0202	0.0190
LFSD $S_\alpha \uparrow$	0.8639	0.8654	0.8700	0.8710
LFSD $F_\beta^m \uparrow$	0.8645	0.8652	0.8663	0.8710
LFSD $E_\varepsilon^m \uparrow$	0.9026	0.9032	0.9055	0.9063
LFSD $\mathcal{M} \downarrow$	0.0708	0.0734	0.0684	0.0654
STERE $S_\alpha \uparrow$	0.8986	0.8994	0.9011	0.9042
STERE $F_\beta^m \uparrow$	0.8916	0.8922	0.8937	0.9013
STERE $E_\varepsilon^m \uparrow$	0.9426	0.9425	0.9427	0.9472
STERE $\mathcal{M} \downarrow$	0.0439	0.0433	0.0427	0.0403

4.4.2. DQPFP Threshold Strategy

As described in Section 3.2, a multivariable strategy was used for α_i and β_i. To verify this strategy, it was compared to the single-variable strategy that uses the same (only one) α_i and β_i. Table 4 shows the results, and it is evident that the multi-factor approach used in this paper is better because it adds flexibility to the network, enabling it to render at different levels with different quality heuristic weights and attention maps.

4.4.3. Effectiveness of Loss and Activation

The loss function is one of the core components of deep learning, measuring the difference between the predicted results of the model and the true labels. By minimizing the value of the loss function, the model can gradually improve its performance during the training process. The loss function provides a clear optimization objective for neural networks and is an important bridge connecting data and model performance. It is necessary to choose a suitable loss function. Thus, we utilized the DQPFPNet to conduct comparative experiments on six datasets using the widely used BCE with the Sigmoid loss, MSE loss, Hinge loss, BCE loss, and PPAI loss to validate the effectiveness of PPAI loss, and the results are shown in Table 5. All other experimental settings remained the same, with only the loss function transformed each time. From the experimental results, it can be seen that the detection accuracy of the model was improved to some extent after using PPAI loss. This indicates that the PPAI loss can accelerate the convergence of the model and drive it toward better performance.

The activation function plays an important role in the backpropagation of neural networks. It introduces nonlinearity into the network, enabling it to learn complex patterns and make accurate predictions. Some activation functions have the problem of gradient disappearance during training, which leads to slow convergence and hinders the learning process. Therefore, the performance and training speed of neural networks can be greatly affected by choosing the appropriate activation function. We conducted ablation experiments and trained the DQPFPNet model using the ReLU, Sigmoid, Tanh, ELU, and RReLU activations, and the results are presented in Table 6. All other experimental configurations remained the same, with only the activation function changed for training each time. The experimental results show that compared with other activations, the RReLU activation enables the model to achieve higher accuracy. This may be related to the introduction of randomness in RReLU, which reduces the occurrence of neuronal "death" through a certain proportion of negative values, improves the stability of the network, and enhances its rich nonlinear expression ability.

4.4.4. Effectiveness of Dual-Stage Decoder

In Table 7, we present the results of ablation experiments on the decoder, where we used a single-stage decoder and a dual-stage decoder. All other conditions remained the same, with only the decoder architecture changing each time. Based on the outcomes of the experiment, it is evident that the resulting metrics when using the dual-stage decoder are better compared to the single-stage decoder across all six datasets, proving that the two-stage decoder is practical and effective. This may be due to the architectural advantages of the dual-stage decoder itself. The first fusion stage reduces the feature channel and hierarchical structure, and the second fusion stage further aggregates the low-level structure and the high-level structure to produce the final salient graph. This two-stage design can make full use of the context information and improve the modeling ability of the model.

Table 4. DQPFP threshold strategy: using identical (only one) α_i and β_i vs. using multiple α_i and β_i (five different values). The best scores are shown in red.

#	Strategy	SIP				NLPR				NJU2K				RGBD135				LFSD				STERE			
		S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$
5	Identical	0.876	0.884	0.923	0.051	0.916	0.905	0.955	0.025	0.900	0.902	0.941	0.041	0.931	0.927	0.968	0.019	0.853	0.852	0.895	0.074	0.890	0.891	0.941	0.044
4	Multiple	0.885	0.896	0.923	0.046	0.922	0.916	0.961	0.023	0.904	0.910	0.947	0.039	0.930	0.942	0.976	0.019	0.870	0.869	0.906	0.068	0.902	0.898	0.947	0.041

Table 5. Ablation analysis of DQPFPNet to validate the effectiveness of the PPAI loss. √ below the module indicates that the model has used the module. Otherwise, the model has not used it. The best results are shown in red.

#	BCE-Logits	MSE	Hinge	BCE	PPAI	SIP				NLPR				NJU2K				RGBD135				LFSD				STERE			
						S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$
6	√					0.8730	0.8790	0.9190	0.0540	0.9120	0.8990	0.9540	0.0270	0.8980	0.9030	0.9410	0.0420	0.9260	0.9310	0.9710	0.0170	0.8500	0.8530	0.8910	0.0750	0.8850	0.8830	0.9380	0.0470
7		√				0.5926	0.6462	0.5545	0.3250	0.7325	0.6517	0.6607	0.1140	0.6764	0.6801	0.7251	0.1260	0.7288	0.6603	0.6585	0.1195	0.7362	0.6134	0.7549	0.1225	0.7884	0.7531	0.6949	0.0980
8			√			0.4991	0.6450	0.5250	0.3420	0.6394	0.7517	0.6325	0.1324	0.7826	0.5801	0.6250	0.2684	0.7351	0.6684	0.7250	0.1107	0.6684	0.7134	0.6822	0.2463	0.6948	0.6531	0.7120	0.2310
9				√		0.8685	0.8715	0.9154	0.0578	0.9170	0.8976	0.9562	0.0270	0.8982	0.9011	0.9429	0.0424	0.9213	0.9084	0.9610	0.0248	0.8547	0.8469	0.8908	0.0746	0.8983	0.8919	0.9421	0.0443
10					√	0.8740	0.8810	0.9323	0.0532	0.9211	0.9048	0.9564	0.0254	0.9029	0.9040	0.9456	0.0405	0.9312	0.9411	0.9716	0.0217	0.8520	0.8558	0.8949	0.0721	0.9015	0.8919	0.9442	0.0423

Table 6. Ablation analysis of DQPFPNet to validate the effectiveness of the RReLU activation. √ below the module indicates that the model has used the module. Otherwise, the model has not used it. The best results are shown in red.

#	ReLU	Sigmoid	Tanh	ELU	RReLU	SIP				NLPR				NJU2K				RGBD135				LFSD				STERE			
						S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$
11	√					0.8730	0.8790	0.9190	0.0540	0.9070	0.8990	0.9540	0.0207	0.8693	0.9030	0.9410	0.0420	0.9260	0.9310	0.9710	0.0270	0.8500	0.8530	0.8910	0.0750	0.8850	0.8830	0.9380	0.0470
12		√				0.3921	0.2462	0.2573	0.2052	0.4316	0.2517	0.3655	0.1043	0.4752	0.5631	0.6581	0.0852	0.4279	0.4603	0.5653	0.1008	0.6605	0.6134	0.6569	0.065	0.5387	0.5624	0.6374	0.0837
13			√			0.4825	0.3462	0.4954	0.1196	0.6182	0.4517	0.6599	0.0725	0.6651	0.6801	0.6746	0.0638	0.5119	0.6603	0.6537	0.0730	0.5524	0.6334	0.5789	0.0863	0.6755	0.6531	0.7975	0.0625
14				√		0.8760	0.8830	0.9210	0.0510	0.9020	0.8891	0.9523	0.0360	0.8970	0.9030	0.9410	0.0430	0.9296	0.9320	0.9660	0.0180	0.8550	0.8560	0.8970	0.0739	0.8890	0.8860	0.9400	0.0450
15					√	0.8842	0.8816	0.9249	0.0506	0.9120	0.8992	0.9542	0.1013	0.8980	0.9036	0.9457	0.0411	0.9340	0.9429	0.9738	0.0170	0.8564	0.8621	0.8997	0.0734	0.8817	0.8901	0.9425	0.0413

Table 7. Ablation analysis of DQPFPNet to validate the effectiveness of the dual-stage decoder. √ below the module indicates that the model has used module, otherwise the model has not used it. The best results are shown in red.

#	Single-Stage	Dual-Stage	SIP				NLPR				NJU2K				RGBD135				LFSD				STERE			
			S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$	S_α	F_β^m	E_ε^m	$\mathcal{M}$
16	√		0.8685	0.8715	0.9154	0.0588	0.9211	0.9088	0.9565	0.0371	0.8979	0.9011	0.9326	0.0424	0.9213	0.9084	0.9610	0.0324	0.8547	0.8469	0.8908	0.0746	0.8983	0.8919	0.9269	0.0542
17		√	0.8850	0.8960	0.9425	0.0460	0.9311	0.9300	0.9612	0.0221	0.9066	0.9100	0.9467	0.0364	0.9411	0.9423	0.9761	0.0190	0.8710	0.8710	0.9063	0.0654	0.9042	0.9013	0.9472	0.0403

5. Conclusions

This paper proposed DQPFPNet, an RGB-D SOD model with high efficiency and good performance. The method models an efficient RGB-D SOD framework and DQPFP processing, greatly improving detection accuracy. DQPFP consists of three sub-modules: DDM, DQPW, and DPEA. The DDM filters multi-scale depth features through a channel attention mechanism and a spatial attention mechanism to achieve the initial filtering of the depth features. The DQPW module weights the depth features based on the alignment between the enhanced RGB features of the DDM module and the depth features, whereas the DPEA module focuses on the depth features spatially using multiple enhanced attention maps originating from the DDM-enhanced depth features refined with low-level RGB features. Additionally, the framework is built on a dual-stage decoder, which helps further increase efficiency. The pixel position adaptive importance (PPAI) loss is utilized to better explore the structural information in the features, making the network attach significance to detailed areas. In addition, the RReLU activation is used to solve the problem of neuronal "necrosis". Experiments conducted on six RGB-D datasets demonstrate that DQPFPNet performs well in terms of both metric values and visualizations. A limitation of the current model is that in the comparison experiments with existing models, it did not achieve the best performance across all metrics and datasets, indicating that the network structure needs to be improved. Furthermore, the behavior of the model in mobile or embedded devices is unknown. Hence, we will continue to explore new network architectures to optimize performance on common datasets in the future. In addition, we will attempt to deploy the DQPFP in embedded/mobile systems that handle RGB-D and video data and continue to optimize the model based on its performance metrics.

Author Contributions: Conceptualization, S.F., L.Z. and S.C.; investigation, S.F., L.Z., J.H., X.Z. and S.C.; methodology, S.F., L.Z., X.Z. and S.C.; code and validation, S.F., L.Z., J.H. and S.C.; writing—original draft preparation, S.F. and S.C.; writing—review and editing, S.F., L.Z., J.H., X.Z. and S.C.; data curation, S.F., L.Z., J.H. and X.Z.; funding acquisition, S.C., J.H. and X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the National Natural Science Foundation of China (Grant No. 61906168, 62101387, 62201400, and 62272267), the Zhejiang Provincial Natural Science Foundation of China (Grant No. LY23F020023, LZ23F020001), the Construction of Hubei Provincial Key Laboratory for Intelligent Visual Monitoring of Hydropower Projects (Grant No. 2022SDSJ01), the Hangzhou AI major scientific and technological innovation project (Grant No. 2022AIZD0061), the Project of Science and Technology Plans of Wenzhou City (Grant No. H20210001) and the Quzhou Science and Technology Projects(2022k91).

Data Availability Statement: This study did not report any data. We used public data for research. The URL and accessed date of the dataset are as follows: https://pan.baidu.com/s/1ckNlS0uEIPV-iCwVzjutsQ, training data, 2022-04-19 (Extracted code: eb2z). https://pan.baidu.com/s/1wI-bxarzdSrOY39UxZaomQ, test data, 2021-08-07 (Extracted code: 940i).

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Deep learning	DL
Red Green Blue	RGB
Red Green Blue-Depth	RGB-D
Salient Object Detection	SOD
Convolutional neural network	CNN
Depth-quality purification feature processing	DQPFP
Rectified Linear Unit	ReLU
Random ReLU	RReLU
Pixel position adaptive importance	PPAI
Depth-quality purification weighting	DQPW

Depth purification-enhanced attention	DPEA
Consumer Electronics	CE
Software-Defined Networking	SDN
Pyramid pooling module	PPM
Depth de-noising module	DDM
Channel attention	CA
Spatial attention	SA
Binary cross-entropy	BCE
Intersection over Union	IoU
Weighted binary cross-entropy	wBCE
Weighted IoU	wIoU
Ground truth	GT
State of the art	SOTA
Mean-square error	MSE
Hyperbolic tangent function	Tanh
Exponential Linear Unit	ELU

References

1. Chan, S.; Tao, J.; Zhou, X.; Bai, C.; Zhang, X. Siamese implicit region proposal network with compound attention for visual tracking. *IEEE Trans. Image Process.* **2022**, *31*, 1882–1894. [CrossRef] [PubMed]
2. Chan, S.; Yu, M.; Chen, Z.; Mao, J.; Bai, C. Regional Contextual Information Modeling for Small Object Detection on Highways. *IEEE Trans.Instrumentation and Measure.* **2023**, *72*, 1–13. [CrossRef]
3. Dilshad, N.; Khan, T.; Song, J.S. Efficient Deep Learning Framework for Fire Detection in Complex Surveillance Environment. *Comput. Syst. Sci. Eng.* **2023**, *46*, 749–764. [CrossRef]
4. Chan, S.; Wang, Y.; Lei, Y.; Cheng, X.; Chen, Z.; Wu, W. Asymmetric Cascade Fusion Network for Building Extraction. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–18. [CrossRef]
5. Javeed, D.; Saeed, M.S.; Ahmad, I.; Kumar, P.; Jolfaei, A.; Tahir, M. An Intelligent Intrusion Detection System for Smart Consumer Electronics Network. *IEEE Trans. Consum. Electron.* **2023**, 1. [CrossRef]
6. Yar, H.; Ullah, W.; Ahmad Khan, Z.; Wook Baik, S. An Effective Attention-based CNN Model for Fire Detection in Adverse Weather Conditions. *ISPRS J. Photogramm. Remote Sens.* **2023**, *206*, 335–346. [CrossRef]
7. Park, K.B.; Lee, J.Y. Novel industrial surface-defect detection using deep nested convolutional network with attention and guidance modules. *J. Comput. Des. Eng.* **2022**, *9*, 2466–2482. [CrossRef]
8. Park, K.B.; Lee, J.Y. SwinE-Net: Hybrid deep learning approach to novel polyp segmentation using convolutional neural network and Swin Transformer. *J. Comput. Des. Eng.* **2022**, *9*, 616–632. [CrossRef]
9. Fan, D.P.; Li, T.; Lin, Z.; Ji, G.P.; Zhang, D.; Cheng, M.M.; Fu, H.; Shen, J. Re-Thinking Co-Salient Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 4339–4354. [CrossRef]
10. Jiang, H.; Wang, J.; Yuan, Z.; Wu, Y.; Zheng, N.; Li, S. Salient Object Detection: A Discriminative Regional Feature Integration Approach. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2083–2090. [CrossRef]
11. Yin, B.; Zhang, X.; Li, Z.; Liu, L.; Cheng, M.M.; Hou, Q. DFormer: Rethinking RGBD Representation Learning for Semantic Segmentation. *arXiv* **2023**, arXiv:2309.09668 .
12. Cong, R.; Liu, H.; Zhang, C.; Zhang, W.; Zheng, F.; Song, R.; Kwong, S. Point-aware Interaction and CNN-induced Refinement Network for RGB-D Salient Object Detection. In Proceedings of the 31st ACM International Conference on Multimedia, Ottawa ON Canada, 29 October–3 November 2023; pp. 406–416. [CrossRef]
13. Wu, Z.; Allibert, G.; Meriaudeau, F.; Ma, C.; Demonceaux, C. HiDAnet: RGB-D Salient Object Detection via Hierarchical Depth Awareness. *IEEE Trans. Image Process.* **2023**, *32*, 2160–2173. [CrossRef] [PubMed]
14. Cong, R.; Lei, J.; Zhang, C.; Huang, Q.; Cao, X.; Hou, C. Saliency Detection for Stereoscopic Images Based on Depth Confidence Analysis and Multiple Cues Fusion. *IEEE Signal Process. Lett.* **2016**, *23*, 819–823. [CrossRef]
15. Chen, Z.; Cong, R.; Xu, Q.; Huang, Q. DPANet: Depth Potentiality-Aware Gated Attention Network for RGB-D Salient Object Detection. *IEEE Trans. Image Process.* **2020**, *30*, 7012–7024. [CrossRef] [PubMed]
16. Fan, D.P.; Yingjie, Z.; Ali, B.; Jufeng, Y.; Ling, S. *BBS-Net: RGB-D Salient Object Detection with a Bifurcated Backbone Strategy Network* ; Lecture Notes in Computer Science; Springer: Cham, Switzerland , 2020.
17. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer Assisted Intervention*; Springer: Cham, Switzerland, 2015.
18. Lang, C.; Nguyen, T.V.; Katti, H.; Yadati, K.; Kankanhalli, M.S.; Yan, S. Depth Matters: Influence of Depth Cues on Visual Saliency. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012.
19. Ren, J.; Gong, X.; Yu, L.; Zhou, W.; Yang, M.Y. Exploiting global priors for RGB-D saliency detection. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Boston, MA, USA, 7–12 June 2015.

20. Qu, L.; He, S.; Zhang, J.; Tian, J.; Tang, Y.; Yang, Q. RGBD Salient Object Detection via Deep Fusion. *IEEE Trans. Image Process.* **2017**, *26*, 2274–2285. [CrossRef] [PubMed]
21. Sun, Y.; Gao, X.; Xia, C.; Ge, B.; Duan, S. GSCINet: Gradual Shrinkage and Cyclic Interaction Network for Salient Object Detection. *Electronics* **2022**, *11*, 1964. [CrossRef]
22. Zhang, J.; Fan, D.P.; Dai, Y.; Anwar, S.; Saleh, F.S.; Zhang, T.; Barnes, N. UCNet: Uncertainty Inspired RGB-D Saliency Detection via Conditional Variational Autoencoders. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020.
23. Li, G.; Liu, Z.; Ling, H. ICNet: Information Conversion Network for RGB-D Based Salient Object Detection. *IEEE Trans. Image Process.* **2020**, *29*, 4873–4884. [CrossRef] [PubMed]
24. Duan, S.; Gao, X.; Xia, C.; Ge, B. A2TPNet: Alternate Steered Attention and Trapezoidal Pyramid Fusion Network for RGB-D Salient Object Detection. *Electronics* **2022**, *11*, 1968. [CrossRef]
25. Chen, H.; Li, Y.; Su, D. Multi-modal fusion network with multi-scale multi-path and cross-modal interactions for RGB-D salient object detection. *Pattern Recognit.* **2019**, *86*, 376–385. [CrossRef]
26. Lu, J.; Yang, J.; Batra, D.; Parikh, D. Hierarchical Question-Image Co-Attention for Visual Question Answering. In Proceedings of the 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, 5–10 December 2016.
27. Yu, Z.; Yu, J.; Cui, Y.; Tao, D.; Tian, Q. Deep Modular Co-Attention Networks for Visual Question Answering. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
28. He, Y.; Lin, J.; Liu, Z.; Wang, H.; Li, L.J.; Han, S. AMC: AutoML for Model Compression and Acceleration on Mobile Devices. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
29. Cheng, Y.; Wang, D.; Zhou, P.; Zhang, T. A Survey of Model Compression and Acceleration for Deep Neural Networks. *arXiv* **2017**, arXiv:1710.09282.
30. Fu, K.; Fan, D.P.; Ji, G.P.; Zhao, Q. JL-DCF: Joint Learning and Densely-Cooperative Fusion Framework for RGB-D Salient Object Detection. *arXiv* **2020**, arXiv:2004.08515.
31. Zeng, Y.; Zhuge, Y.; Lu, H.; Zhang, L.; Qian, M.; Yu, Y. Multi-source weak supervision for saliency detection. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
32. Zhang, D.; Meng, D.; Zhao, L.; Han, J. Bridging saliency detection to weakly supervised object detection based on self-paced curriculum learning. In Proceedings of the International Joint Conference on Artificial Intelligence, New York, NY, USA, 9–15 July 2016.
33. Liu, N.; Zhang, N.; Han, J. Learning Selective Self-Mutual Attention for RGB-D Saliency Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020.
34. Zhou, T.; Fan, D.P.; Cheng, M.M.; Shen, J.; Shao, L. RGB-D salient object detection: A survey. *Comput. Vis. Media* **2021**, *7*, 37–69. [CrossRef] [PubMed]
35. Chen, Q.; Fu, K.; Liu, Z.; Chen, G.; Du, H.; Qiu, B.; Shao, L. EF-Net: A novel enhancement and fusion network for RGB-D saliency detection. *Pattern Recognit.* **2021**, *112*, 107740. [CrossRef]
36. Zhao, X.; Zhang, L.; Pang, Y.; Lu, H.; Zhang, L. A Single Stream Network for Robust and Real-time RGB-D Salient Object Detection. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
37. Piao, Y.; Rong, Z.; Zhang, M.; Ren, W.; Lu, H. A2dele: Adaptive and Attentive Depth Distiller for Efficient RGB-D Salient Object Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020.
38. Sun, F.; Xu, Y.; Sun, W. SPSN: Seed Point Selection Network in Point Cloud Instance Segmentation. In Proceedings of the International Joint Conference on Neural Network, Glasgow, UK, 19–24 July 2020.
39. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid Scene Parsing Network. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
40. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
41. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
42. Howard, A.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
43. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
44. Li, N.; Ye, J.; Ji, Y.; Ling, H.; Yu, J. Saliency Detection on Light Field. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014.
45. Ju, R.; Ge, L.; Geng, W.; Ren, T.; Wu, G. Depth saliency based on anisotropic center-surround difference. In Proceedings of the International Conference on Image Processing, Paris, France, 27–30 October 2014.
46. Peng, H.; Li, B.; Xiong, W.; Hu, W.; Ji, R. RGBD Salient Object Detection: A Benchmark and Algorithms. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014.
47. Cheng, Y.; Fu, H.; Wei, X.; Xiao, J.; Cao, X. Depth Enhanced Saliency Detection Method. In Proceedings of the International Conference on Internet Multimedia Computing and Service, Xiamen, China, 10–12 July 2014.

48. Fan, D.P.; Lin, Z.; Zhang, Z.; Zhu, M.; Cheng, M.M. Rethinking RGB-D Salient Object Detection: Models, Data Sets, and Large-Scale Benchmarks. *IEEE Trans. Neural Netw.* **2021**, *32*, 2075–2089. [CrossRef] [PubMed]
49. Niu, Y.; Geng, Y.; Li, X.; Liu, F. Leveraging stereopsis for saliency analysis. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012.
50. Fan, D.P.; Cheng, M.M.; Liu, Y.; Li, T.; Borji, A. Structure-measure: A New Way to Evaluate Foreground Maps. *arXiv* **2017**, arXiv:1708.00786.
51. Achanta, R.; Hemami, S.S.; Estrada, F.J.; Süsstrunk, S. Frequency-tuned salient region detection. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009.
52. Fan, D.P.; Gong, C.; Cao, Y.; Ren, B.; Cheng, M.M.; Borji, A. Enhanced-alignment Measure for Binary Foreground Map Evaluation. In Proceedings of the International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018.
53. Fan, D.P.; Ji, G.P.; Qin, X.; Cheng, M.M. Cognitive vision inspired object segmentation metric and loss function. *Sci. Sin. Inf.* **2021**, *51*, 1475–1489. [CrossRef]
54. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Proceedings of the Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019.
55. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2014**, arXiv:1412.6980.
56. Zhang, M.; Yao, S.; Hu, B.; Piao, Y.; Ji, W. C^2 DFNet: Criss-Cross Dynamic Filter Network for RGB-D Salient Object Detection. *IEEE Trans. Multimed.* **2022**, *25*, 5142–5154. [CrossRef]
57. Ji, W.; Li, J.; Zhang, M.; Piao, Y.; Lu, H. Accurate RGB-D Salient Object Detection via Collaborative Learning. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
58. Cong, R.; Lin, Q.; Zhang, C.; Li, C.; Cao, X.; Huang, Q.; Zhao, Y. CIR-Net: Cross-modality Interaction and Refinement for RGB-D Salient Object Detection. *IEEE Trans. Image Process.* **2022**, *31*, 6800–6815. [CrossRef]
59. Zhao, X.; Pang, Y.; Zhang, L.; Lu, H.; Ruan, X. Self-Supervised Pretraining for RGB-D Salient Object Detection. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 22 February–1 March 2022; pp. 3463–3471. [CrossRef]
60. Li, C.; Cong, R.; Piao, Y.; Xu, Q.; Loy, C.C. RGB-D Salient Object Detection with Cross-Modality Modulation and Selection. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
61. Ji, W.; Li, J.; Yu, S.; Zhang, M.; Piao, Y.; Yao, S.; Bi, Q.; Ma, K.; Zheng, Y.; Lu, H.; et al. Calibrated RGB-D Salient Object Detection. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021.
62. Zhang, M.; Fei, S.X.; Liu, J.; Xu, S.; Piao, Y.; Lu, H. Asymmetric Two-Stream Architecture for Accurate RGB-D Saliency Detection. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
63. Sun, P.; Zhang, W.; Wang, H.; Li, S.; Li, X. Deep RGB-D Saliency Detection with Depth-Sensitive Attention and Automatic Multi-Modal Fusion. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021.
64. Chen, S.; Fu, Y. Progressively Guided Alternate Refinement Network for RGB-D Salient Object Detection. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
65. Wu, Y.H.; Liu, Y.; Xu, J.; Bian, J.W.; Gu, Y.C.; Cheng, M.M. MobileSal: Extremely Efficient RGB-D Salient Object Detection. *arXiv* **2020**, arXiv:2012.13095.
66. Zhang, W.; Ji, G.P.; Wang, Z.; Fu, K.; Zhao, Q. Depth Quality-Inspired Feature Manipulation for Efficient RGB-D Salient Object Detection. *arXiv* **2021**, arXiv:2107.01779.