MHGAN: A Multi-Headed Generative Adversarial Network for Underwater Sonar Image Super-Resolution

Journal:	*Transactions on Geoscience and Remote Sensing*
Manuscript ID	TGRS-2023-01858
Manuscript Type:	Regular paper
Date Submitted by the Author:	28-Apr-2023
Complete List of Authors:	Ma, Zhengda; Fudan University, Department of Electronic Engineering Li, Sensen; Fudan University, Department of Electronic Engineering Ding, Jie; Fudan University, Department of Electronic Engineering Zou, Binbin; Chinese Academy of Sciences, Shanghai Acoustic Laboratory
Keywords:	Oceans and Water

MHGAN: A Multi-Headed Generative Adversarial Network for Underwater Sonar Image Super-Resolution

Zhengda Ma, Sensen Li, Jie Ding, Binbin Zou

***Abstract*—Super-Resolution is a technique for recovering image details based on available information, avoiding image quality degradation by increasing an image's resolution. Despite the great success of deep learning-based SR models, they are rarely applied to underwater sonar scenarios due to the lack of underwater sonar datasets and the difficulty in recovering texture details. To address these challenges, we propose a multi-headed generative adversarial network (MHGAN) for the super-resolution of underwater sonar images. The main body of the generator is designed with the introduced Residual Simple-dense Self-calibrated Block (RSSB), which combines Self-calibrated Attention and Pixel Attention to improve generalization capability. Additionally, the discriminator employs a novel multi-headed U-net structure to acquire detailed texture information at multiple patch sizes of different scales, along with the generated image and High-Resolution(HR) image details. Moreover, we introduce the Correction loss, which acts as a pixel-level correction to help improve the quality of the output images, and multi-perceptual loss based on VGG19 to extract multi-scale features for comparison. Meanwhile, we also introduce the Underwater Sonar Dataset for Super-Resolution (USDSR). We perform the visual and quantitative comparison, ablation experiments, and model analysis with state-of-the-art methods on both the public KLSG dataset and the built USDSR dataset. Experimental results demonstrate the effectiveness and superiority of our proposed MHGAN model.**

***Index Terms*—Generative adversarial network (GAN), Underwater sonar image, Single-image super-resolution (SISR)**

I. Introduction

With the increasing importance of exploiting ocean resources, sonar is an indispensable sensor in the field of remote sensing that provides rich underwater observation for the rational development and utilization of marine resources. However, due to the principles of underwater acoustics, the energy of sonar acoustic waves attenuates in water, which leads to distorted and low-resolution underwater sonar images [1]. Therefore, it is necessary to enhance the resolution of sonar images. Super-resolution (SR), as a powerful technique, can not only improve the quality of underwater sonar images but also plays a critical role in further processing underwater classification [2], object detection [3]–[5], instance segmentation [6]–[8], and other sonar image scenes [9]–[12].

Super-resolution is a method for reconstructing high-resolution (HR) images from one or more low-resolution (LR) images [13]. Depending on the number of input LR images, SR algorithms can be classified as single-image SR (SISR) and multi-image SR (MISR). SISR [14]–[17] is a rapidly developing research topic. It is widely used in computer vision and is crucial for the real world, such as environmental monitoring [18], medical image processing [19]–[21], and surveillance and security [22], [23]. However, SISR is a typical ill-posed problem, as multiple HR images can be generated for any LR image. Therefore, exploring the mapping relationship between HR and LR images is the key to solving this problem. Previous research has proposed several methods to learn such mappings. Traditional methods use filtering and interpolation methods, such as linear, Lanczos filtering [24], and bicubic interpolation [25], to generate high-resolution images based on the neighborhood information. Although these methods are computationally efficient, they over-simplify the mapping relationships in SISR, resulting in overly smooth generated images that ignore some important details, especially edge and texture information.

Because of successful applications of deep convolutional neural networks in vision tasks such as image classification, object detection, target recognition, and semantic segmentation, Dong et al. [26] introduce deep convolutional neural network (CNN) into the SR domain, to learn the mapping relationship between LR and HR images. Through the development of deep learning neural networks, some neural network models with powerful non-linear fitting and learning capabilities are designed, e.g., ResNet [27] and DenseNet [28], leading to the emergence of deep learning-based super-resolution models. These models use deep networks to extract higher-level features, achieving significant fitting performance, including VDSR [29], DRCN [30], EDSR [31], ESPCN [32], and DBPN [33]. Swin Transformer is introduced into the field of super-resolution by Liang et al. [34], and SwinIR achieves advanced performance through shallow feature extraction, deep feature extraction, and high-quality image reconstruction.

Although complex convolutional neural networks make significant breakthroughs in accuracy and speed for image super-resolution, they still suffer from the problem of generating SR images that are often blurry, lacking high-frequency details, and unsatisfactory in terms of perceptual quality, which cannot

This work was supported by National Natural Science Foundation of China under Grant 62103107, Shanghai Sailing Program under Grant 21YF1403000, Shanghai Science and Technology Committee under Grant 22dz1204002, and Young Potential Program of Shanghai Acoustics Laboratory, Chinese Academy of Sciences under Grant YXJH202203.

Z. Ma, S. Li, and J. Ding are with the Department of Electronic Engineering, School of Information Science and Engineering, Fudan University, Shanghai 200433, China. E-mail: fyuan9634@gmail.com (Z. Ma), liss21@m.fudan.edu.cn (S. Li), dingjie@fudan.edu.cn (J. Ding).

B. Zou is with Shanghai Acoustic Laboratory, Chinese Academy of Science, Shanghai 200032, China. Email: zoubb@mail.ioa.ac.cn (B. Zou)

match the expected fidelity at higher resolutions. Therefore, Ledig et al. [35] introduce adversarial generative network (GAN) into CNN-based super-resolution methods for recovering realistic textures in severely downsampled images. Since the pioneering work of SRGAN [35], GAN has brought prosperity to the SR field. ESRGAN [36] extends the generator network with residual in residual dense block (RRDB) without batch normalization and utilizes the discriminator to reduce unpleasant artifacts. The Fine-grained Attention Generative Adversarial Network (FASRGAN) [37] improves the ability to generate high-quality images using generated image scores and image score maps.

Despite the successful applications of deep neural networks in SR for optical images, there are few studies for implementing SR with limited samples of underwater sonar images. This is because sonar images are generated based on acoustic signals with different textures and details from optical images, and the underwater background is complicated. Due to wide variation in the texture and structural information in underwater sonar images, existing algorithms such as EDSR [31], DBPN [33], RCAN [34], or SwinIR [35] often produce blurred underwater sonar images with limited ability to recover their details and textures. Moreover, the noise in sonar images is complex, such as Speckle noise, making it challenging to generalize current SR methods to find the mapping relationships between high-resolution and low-resolution underwater sonar images.

In order to cope with the above problems and further recover realistic texture details of underwater sonar images, in this paper, we propose a multi-headed adversarial generative network (MHGAN) for underwater sonar images. Specifically, the main body of the generator consists of Residual Simple-dense Self-calibrated Block (RSSB) and an Upsampling Attention Block (UAB). Self-calibrated Attention Block (SCAB) and Self-calibrated Net are introduced in RSSB on the basis of Self-calibrated Convolution [38] to improve the generalization ability of the model. The Self-calibrated module enables any spatial position to adaptively encode information from distant regions for further extraction of discriminative features. Pixel Attention [39], in combination with pixel shuffle in the upsampling module, improves the ability to generate image details while maintaining the original image features. For the discriminator, U-net is used to combine multi-level image information. Moreover, a multi-head structure is designed to capture and integrate multi-level, multi-scale (different patch sizes) features. At the same time, we redefine GAN loss to include multi-headed loss and correction loss, which guides the learning process of the generator in order to reconstruct underwater sonar images with rich details and textures.

In summary, the main contributions of this paper are listed as follows.

1) Firstly, we propose the MHGAN Single Image Super Resolution model with application in underwater sonar images. With the proposed MHGAN, detailed features of the whole images can be accurately captured while discriminant multi-scaled patches are incorporated. As a result, MHGAN can achieve super-resolution of images with small datasets and complex backgrounds while keeping the number of parameters small.
2) Secondly, in the generator, we design the Simple-dense Net with the Self-calibrated Net to extend the perceptual field of each convolutional layer and obtain rich output features. The Self-calibrated Attention Block is constructed based on Self-calibrated Convolutions to establish remote spatial and inter-channel attention calibration for each location in space. The Upsampling Attention Block (UAB) is also used to focus on pixel-level changes in the upsampling process. Integrating the above modules gives the generator a solid ability to reconstruct details and realistic textures.
3) Thirdly, an Unet-based discriminator with a Multi-headed Module is introduced to obtain multi-level, multi-scale features for integration. The output of the discriminator is used for pixel-level corrections between the SR and HR images. This correction mechanism enables the generator to focus on the challenging areas of the image for reconstruction. The constructed discriminator can distinguish SR from HR images at the level of detail and is used to guide the generator to produce realistic textures in complex contexts.
4) Finally, in the loss function, we redefine the GAN loss to include correction loss. At the same time, we choose multiple feature extraction layers in the VGG19 model to combine their outputs as a multi-perceptual loss. For datasets, we build an Underwater Sonar Dataset for Super-Resulotion (USDSR) dataset based on publicly available underwater sonar images. The code will be published at: https://github.com/white-1118/MHGAN for future studies.

The remainder of the paper is organized as follows. Section II describes the related work. A description of the MHGAN model structure, loss functions, and other details are presented in Section III. The datasets used in the article, details of the experimental implementation, and discussions of the experimental results and ablation studies analysis are presented in Section IV. Finally, conclusions are drawn in Section V.

II. RELATED WORK

A. CNN-based SISR

The first deep learning-based SISR model, SRCNN [26], outperforms previous traditional SR models in terms of accuracy due to its end-to-end structure, which uses a shallow model with fewer than five layers to learn the mapping between low- and high-resolution images. FSRCNN is developed in [40] with a compact hourglass-shaped CNN structure to achieve faster and better SR. As deep learning models are evolving, SR results are improved by increasing the number of convolutional layers in models such as VDSR [29] with 20 layers and DRCN [30] with 16 recursive layers. Tai et al. [41] introduce a deep recursive residual network DRRN with 52 convolutional layers, which recursively uses residual learning to increase depth in both global and local aspects. SRResNet, introduced in [35], achieves excellent performance by stacking multiple residual blocks. EDSR [31] improves

model efficiency by removing the traditional residual network in the batch normalization module for optimizing and scaling up the model size, resulting in significant performance improvements. RCAN [42], with residual-in-residual (RIR) to enable the leading network to focus on learning high-frequency information, implements a channel attention (CA) mechanism to adaptively adjust channel characteristics by considering the interdependencies between channels.

In recent years, based on deep learning models, Lee et al. [43] construct a distillation-based teacher-student network framework based on FSRCNN [40] network structure, showing promising results with a few parameters. The authors in [44] adopt scale-aware feature adaptive blocks and scale-aware upsampling layers to construct a scale-arbitrary image SR network that achieves desirable results at many scales. Furthermore, some works apply attention modules achieving promising SR performance. The holistic attention network (HAN) [45] is designed to model the interdependence between layers, channels, and locations through the layer attention module and channel-space attention module (CSAM). Zhang et al. [46] propose Contextual Reasoning Attention Network (CRAN) to adaptively modify the convolutional kernel according to the global context by introducing channel and spatial interactions. Gu et al. [47] introduce the local attribution map, and demonstrate that features can be extracted from a broader range of input pixels with attentional and non-local networks. It is also found that textures with regular stripes and grids are easily noticed by SR networks, while relatively complex textures are difficult to exploit. It is pointed out in [48] that channel representation capability and generalization of the Super Resolution network can be improved by an appropriate Dropout and the performance is significantly improved in a multi-degenerate setting. Wang et al. [49] propose adaptive patch exiting (APE), which trains a regressor to predict the incremental capacity of each patch layer to trade-off performance and efficiency.

B. GAN-based SISR

Most super-resolution (SR) algorithms achieve state-of-the-art results on two general criteria: Peak Signal-to-Noise Ratio (PSNR) [50] and Structural Similarity Index Measure (SSIM) [51]. However, high PSNR and SSIM values do not always guarantee satisfactory and realistic visual results. To overcome this limitation, researchers utilize adversarial generative networks to produce perceptually satisfying images. Generative Adversarial Network is introduced into SR models by Ledig et al. [35]. Similarly, ESRGAN [36] combines the Residual-in-Residual Dense Block (RRDB) without batch normalization, which provides substantial supervision for luminance consistency and texture recovery. Despite being effective in generating texture details and producing perceptually realistic results, GAN-based SR models can improve their performance in terms of PSNR and SSIM. Lee et al. [52] propose a neural architecture search (NAS) method that combines recent advances in GAN and perceptual SR to improve the efficiency of small perceptual SR models. These modifications lead to more realistic and naturalistic textures, improving the realism of the reconstructed images.

In recent years, GAN-based super-resolution research achieves several significant advancements. Researchers in [53] utilize a pre-trained GAN to capture a rich and diverse prior to improve the quality of image super-resolution recovery while maintaining the texture fidelity of the generated images. Another approach proposed in [54] is best-buddy GANs (BebyGAN), generating rich detail and texture information by dynamic supervision and region-aware adversarial learning strategies. Furthermore, the authors in [55] introduce locally discriminative learning (LDL) and a framework to distinguish generated artifacts from realistic details, which leads to the suppression of visual artifacts while stably generating perceptually realistic details. GAN-based SR models are also commonly used in remote sensing image restoration. MA-GAN [56] and SRAGAN [57] are two state-of-the-art models incorporating different Attention modules to achieve the high-resolution reconstruction of remote-sensing images. MA-GAN adopts an Attention Pyramid Convolution (AttPConv) operator in the Generator network to generate high-resolution remote sensing images. SRAGAN uses local and global attention mechanisms to capture different feature levels and reconstruct images with realistic details.

III. METHODOLOGY

In order to generate high-quality and pleasing underwater sonar images, we design MHGAN, which contains a Self-calibrated Attention Generator and Multi-headed Discriminator with correction, as shown in Figs. 1 and 2, respectively. In this section, firstly, we present the specific structures of the two frameworks separately. Subsequently, we introduce GAN loss of MHGAN. Finally, we present the implementation of multi-perceptual loss and give the overall loss functions used in this paper.

A. Generator network architecture of MHGAN

Our proposed MHGAN generator is composed of four parts, which are illustrated in Fig. 1. The input layer is a single convolutional layer, responsible for broadening the number of channels of the input image and extracting the shallow features F_0 of the LR. The output layer adopts a Convolution-LeakyReLU-Convolution structure to produce the super-resolved image. The generator body is composed of a Deep Feature Extraction Module and an Image Reconstruction Module. The Deep Feature Extraction Module combines N Residual Simple-dense Self-calibrated Blocks (RSSB) with a signal Convolutional layer, featuring a shortcut connection structure. Meanwhile, the Image Reconstruction Module consists of M Upsampling Attention Blocks (UAB). The overall generator can be formulated as follows:

$$I^{SR} = G(I^{LR}) \tag{1}$$

where I^{LR} is low-resolution image, I^{SR} is a super-resolution image generated by the generator and $G(\cdot)$ represents the

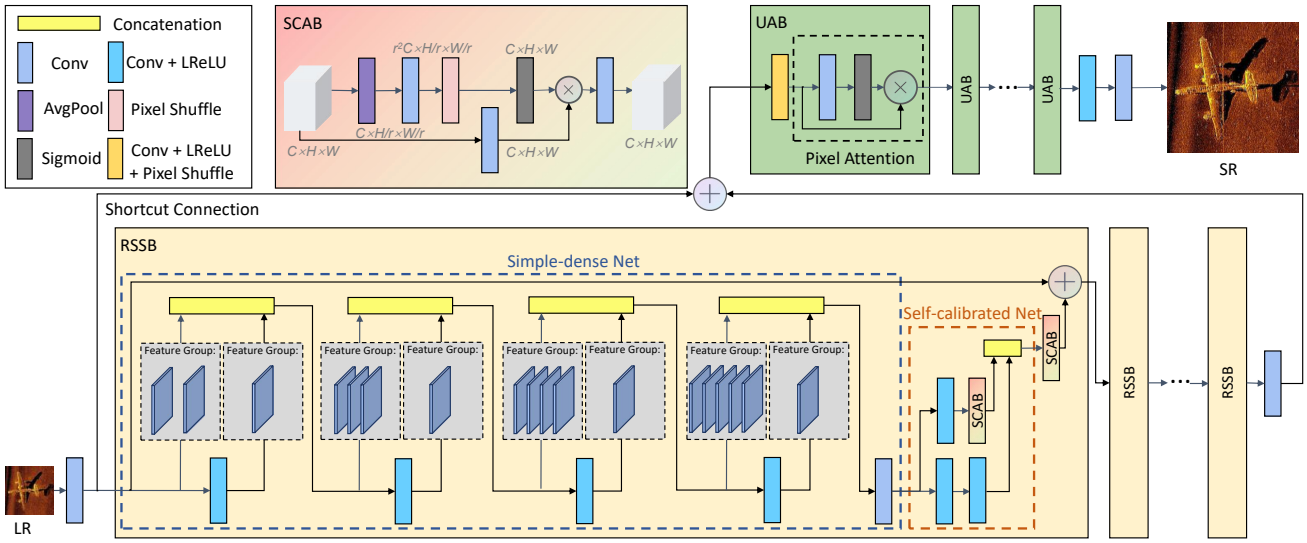

Fig. 1. Architecture of the Self-calibrated Attention Generator network of the proposed MHGAN.

generator's function. The four parts of the generator are formulated as follows:

$$F_0 = Conv_3(I^{LR}) \tag{2}$$
$$F_n = \mathcal{F}_n^{\mathrm{RSSB}}(F_{n-1}), n = 1, 2, \ldots, N \tag{3}$$
$$F_0^u = Conv_3(F_N) + F_0 \tag{4}$$
$$F_m^u = \mathcal{F}_m^{\mathrm{UAB}}(F_{m-1}^u), m = 1, 2, \ldots, M \tag{5}$$
$$I^{SR} = Conv_3(\sigma(Conv_3(F_M^u))) \tag{6}$$

where $Conv_k(\cdot)$ denotes convolution with a kernel size of $k \times k$. The output of the n-th RSSB is represented by F_n, and $\mathcal{F}_n^{\mathrm{RSSB}}(\cdot)$ is the function of the n-th RSSB. Similarly, F_m^u denotes the output of the m-th UAB, and $\mathcal{F}_m^{\mathrm{UAB}}(\cdot)$ refers to the function of the m-th UAB. $\sigma(\cdot)$ is the Leaky ReLU layer [58] in the proposed method. Finally, F_0^u is the output of the Deep Feature Extraction Module and also serves as the input of the Image Reconstruction Module.

1) Residual Simple-dense Self-calibrated Block (RSSB): We adopt the RSSB to extract intermediate features (F_n) from the LR. The RSSB involves a Simple-dense Net, Self-calibrated Net, and Self-calibrated Attention Block (SCAB). Furthermore, we implement residual learning by introducing a shortcut connection between the start section and the output of the SCAB module, which helps reduce learning difficulties.

$$\mathcal{F}_n^{\mathrm{RSSB}}(F_{n-1}) = \mathcal{F}_n^{\mathrm{SCAB}}(\mathcal{F}_n^{\mathrm{SC}}(\mathcal{F}_n^{\mathrm{SD}}(F_{n-1}))) + F_{n-1}$$

where $\mathcal{F}_n^{\mathrm{SD}}(\cdot)$, $\mathcal{F}_n^{\mathrm{SC}}(\cdot)$, and $\mathcal{F}_n^{\mathrm{SCAB}}(\cdot)$ represent the Simple-dense Net function, Self-calibrated Net function, and Self-calibrated Attention Block function of the n-th RSSB, respectively.

Specifically, in RSSB, the Simple-dense Net is composed of multiple modules of Convolution-LeakyReLU (Conv-LReLU) and is connected to the Self-calibrated Net. The Simple-Dense Net differs from the DenseNet network structure proposed by [28] and the Resnet network structure with cross-layer element-wise addition. Actually, the DenseNet tends to produce smoother decision boundaries when training data is insufficient. After numerous experimental comparisons, we find that Simple-dense Net, as a simplified version of DenseNet, is suited for underwater sonar images, as it maintains a particular texture structure. Therefore, Simple-dense Net is designed as shown in Fig. 1, in which cross-channel concatenation is used but the input of each layer only comes from the previous layer.

$$F_{n,d} = \mathbb{C}(F_{n,d-1}, \sigma(Conv_3(F_{n,d-1})))$$
$$\mathcal{F}_n^{\mathrm{SD}}(F_{n,0}) = Conv_3(F_{n,4})$$

where $F_{n,d}$ refers to the output of the d-th concatenation operator in n-th RSSB, $d = 1, 2, \ldots, 4$. $\mathbb{C}(\cdot, \cdot)$ represents the operator of concatenation along the channel.

In this work, we employ a Self-calibrated Net to efficiently capture contextual information for each spatial location. The Self-calibrated Net, inspired by the Self-calibrated Convolutions structure proposed in [38], performs a convolutional feature transformation in two different scale spaces: the feature maps in the original scale space and the feature maps in a small potential area. As illustrated in Fig. 1, the Self-calibrated Net splits the output of the Simple-dense Net into two feature maps with the same number of channels using convolution. One feature map is processed through a convolutional layer to generate the feature map in the original scale space, while the other is through a convolutional layer to create a feature map in a downsampled latent area by the SCAB module. The two feature maps are then concatenated across channels to form the output of the Self-calibrated Net. The downsampled feature map has a larger receptive field, which serves as a reference to guide the transformation of the elements in the original feature space. The Self-calibrated Net process is defined as:

$$\mathcal{F}_n^{\mathrm{SC}}(X) = \mathbb{C}(\mathcal{F}_n^{\mathrm{SCAB}}(Conv_3(X)), Conv_3(Conv_3(X)))$$

where X denotes the feature maps.

In the Self-calibrated Net, we utilize SCAB as an Attention-like structure to generate a downsampled feature map that

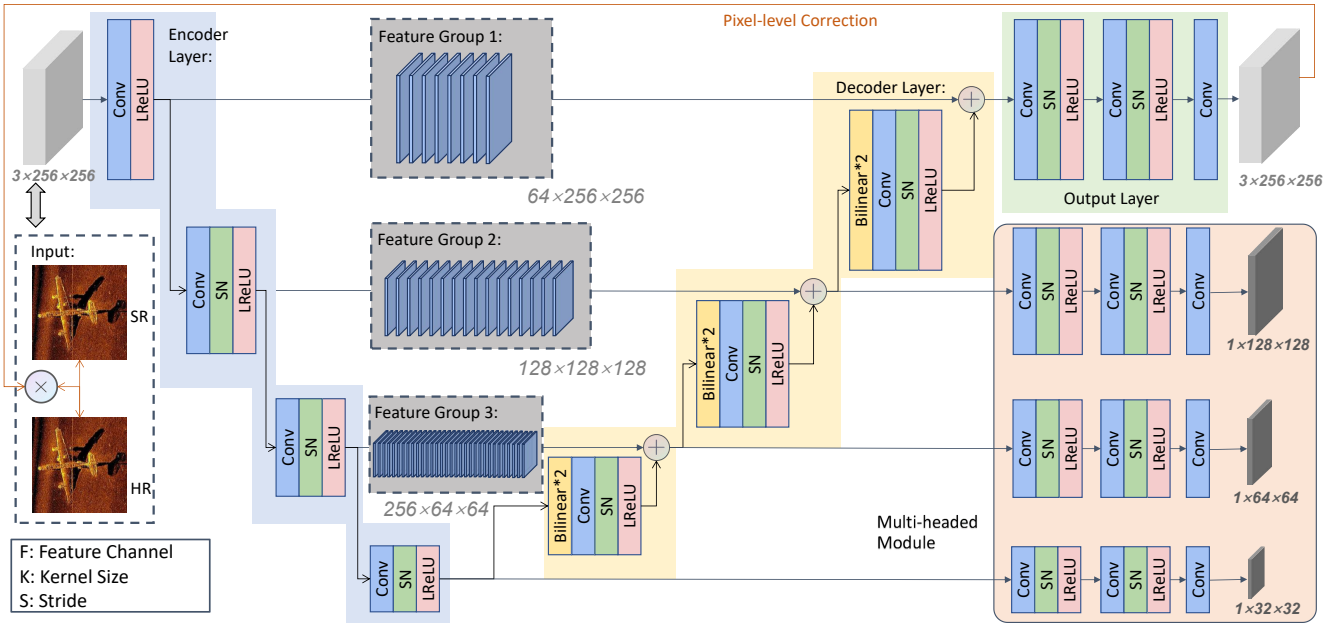

Fig. 2. Architecture of the Multi-headed Discriminator with correction network of the proposed MHGAN.

enhances the field of perception. This approach enables any spatial location to adaptively encode context information from distant regions, instead of being limited to a small 3×3 kernel size. As depicted in Fig. 1, the feature map generated by Avgpool-Conv-PixelShuffle is passed through the Sigmoid function. Then, we multiply the output of Sigmoid by the input feature map passing through a single convolutional layer, followed by another convolutional layer. This process enriches the contextual relationships within the feature map and improves the representation. The SCAB formula can be expressed as follows:

$$\mathcal{F}_n^{\mathrm{SCAB}}(X) = Conv_3(Conv_3(X) \otimes \delta(ACP(X)))$$

where δ denotes the sigmoid activation function. $\otimes$ is an elementwise multiplication operator. $ACP(\cdot)$ represents the operator of $AvgPool\_Conv_3\_PixelShuffle$.

2) Upsampling Attention Block (UAB): In the upsampling process, we employ the Upsampling Attention Block (UAB), which combines Pixel Attention [39] with Pixel Shuffle for efficient upsampling, as illustrated in Fig. 1. Each UAB has an upsampling multiplier of 2. Unlike interpolation, we use Pixel Shuffle [32] for upsampling, as it enhances the non-linear representation and improves the fitting capability by merging feature maps in a recombination manner. To assign a weight to each pixel based on its feature value, we adopt a simple Pixel Attention structure, where the input is passed through Conv-Sigmoid operation, followed by an element-wise multiplication to generate the output feature map. This adaptive learning approach enables the model to perceive local information effectively by assigning different degrees of importance to each pixel during feature extraction. The UAB is defined as:

$$\mathcal{F}_n^{\mathrm{UAB}}(X) = CLP(X) \otimes \delta(Conv_1(CLP(X)))$$

where $CLP(\cdot)$ represents the operator of $Conv_3\_LeakyReLU\_PixelShuffle$.

B. Discriminator network architecture of MHGAN

In the field of SR, GAN is adopted as a way to restore texture details. Previous methods use a discriminator to generate a score value [35], [36], [56], or a score map [57], [59] for the input image. However, with regard to the methods with a single score for the entire image, it is too rough in making decisions as local features are ignored. On the other hand, the methods that generate a score map for each pixel in an image are too meticulous, making it difficult to maintain consistency in judging both local and global features. Moreover, underwater sonar images are rich in background, and it is difficult to use evaluations of the entire image or individual pixels separately as a reference for accurately restoring texture details. To address the above issue, the discriminator designed in MHGAN involves a multi-head module that outputs two parts: Patch Map, which discriminates against images of different patch sizes, and Pixel Map, which serves as fine-grained discrimination of each pixel in a single image. The discriminator proposed in this paper includes four parts: Encoder, Decoder, output layers, and Multi-headed Module, whose specific structure is shown in Fig. 2.

1) Encoder: The Encoder component is designed based on the discriminator structure of ESRGAN with VGG style [36]. Throughout the Encoder process, convolution is utilized to reduce the feature map size while increasing the number of feature maps. To mitigate training instability and avoid the introduction of sharp artifacts, we adopt the spectral normalization regularization approach proposed in [59], [60], which also stabilizes the training dynamics. The feature map generated by the Encoder process has 512 channels and covers 1/64 of the area of the input image. The output of the i-th layer

of the Encoder is denoted as F_i^E, corresponding to the number of feature maps of 64, 128, 256, and 512, respectively.

$$F_0^E = \sigma(Conv_3(I_{input}^D))$$
$$F_i^E = \sigma(SN(Conv_4(F_{i-1}^E))), i = 1, 2, 3$$

where $SN(\cdot)$ represents spectral normalization regularization function.

2) Decoder and Output layers: In the Decoder section, we utilize Bilinear interpolation to up-sample the feature maps and reduce the number of feature maps through convolution layers, thereby achieving the same spatial size and the number of feature maps as the output of each Encoder layer. Afterward, we perform feature fusion by connecting the output of each layer of the Encoder with equal size and the number of feature maps in the Decoder. The output of the j-th decoder layer is denoted as F_j^D.

$$F_0^D = F_3^E$$
$$F_j^D = CSL(Bilinear(F_{j-1}^D)) + F_{3-j}^E, j = 1, 2, 3$$

where $Bilinear(\cdot)$ means Bilinear function. $CSL(\cdot)$ represents the operator of $Conv_3\_SN\_LeakyReLU$.

Finally, we feed F_3^D into the output layers that contain three convolutional layers. In addition, the output of the discriminator is a pixel-level score map $M \in \mathbb{R}^{W \times H \times 3}$, which has the same spatial size as the input image. The final output of the discriminator is denoted as M_{out}^D.

$$M_{out}^D = Conv_3(CSL(CSL(F_3^D)))$$

3) Multi-headed Module: In the proposed method, we aim to improve the discrimination accuracy of the generated image by providing true and false judgments at different patch sizes, rather than a global judgment for the entire image. This is achieved by combining the Encoder and Decoder output feature maps of different spatial sizes and passing them through separate convolution layers to generate three single-channel feature maps of patch resolution $3 \times 8 \times 8$, $3 \times 4 \times 4$, and $3 \times 2 \times 2$. The resulting patch score maps M_8^P, M_4^P, and M_2^P provide accurate feedback for local textures, which improves the ability to recover details and textures at different patch sizes. This approach is particularly effective in scenarios with complex backgrounds, such as underwater sonar image super-resolution.

$$M_8^P = Conv_3(CSL(CSL(F_3^E)))$$
$$M_4^P = Conv_3(CSL(CSL(F_1^D)))$$
$$M_2^P = Conv_3(CSL(CSL(F_2^D)))$$

C. GAN loss

Our proposed GAN loss has two components: Multi-headed Loss and Correction Loss. The Multi-headed Loss is for the Multi-headed Module outputs in order to estimate the input image at different patch levels. The Correction Loss is for the output of the discriminator and provides pixel-level estimation.

1) Multi-headed Loss: In this case, the Multi-headed GAN Loss is based on different patch sizes, M_s^P. The method of relativistic GAN [61], which considers both the quality of the SR image and the distribution of the HR image, leads to improved effectiveness of the generated model. Specifically, with three score maps M_2^P, M_4^P, M_8^P of different patch sizes, Multi-headed GAN loss is defined as:

$$\begin{aligned} L_s^G = &-\mathbb{E}[\log(1 - M_s^P(I^{HR}) + \mathbb{E}[M_s^P(G(I^{LR}))])] \\ &- \mathbb{E}[\log(M_s^P(G(I^{LR})) - \mathbb{E}[M_s^P(I^{HR})])] \\ L_s^D = &-\mathbb{E}[\log(M_s^P(I^{HR}) - \mathbb{E}[M_s^P(G(I^{LR}))])] \\ &- \mathbb{E}[\log(1 - M_s^P(G(I^{LR})) + \mathbb{E}[M_s^P(I^{HR})])] \end{aligned}$$

$$L_{Multi-headed}^G = \frac{1}{3}\sum_{s=2}^{8} L_s^G$$

$$L_{Multi-headed}^D = \frac{1}{3}\sum_{s=2}^{8} L_s^D$$

where s = 2, 4, 8, $L_{Multi-headed}^G$ and $L_{Multi-headed}^D$ means the multi-headed loss of the generator and discriminator, respectively. $\mathbb{E}(\cdot)$ represents the average over a mini-batch.

2) Correction Loss: Correction Loss utilizes the output of the discriminator, which is a pixel-level score map of the same size as the input image, with each score between 0 and 1 representing the pixel proximity of the input image to the corresponding HR image. The score map determines which parts of the image are challenging. To achieve fine-grained correction at the pixel level, we combine the pixel-level score maps with the difference between the SR image and the HR image. This allows us to focus on the challenging parts of the image and guide the generator to generate accurate pixels. Correction Loss of the generator is defined as

$$\begin{aligned} L_{Correction}^G = \frac{1}{WHC}\sum_{w=1}^{W}\sum_{h=1}^{H}\sum_{c=1}^{C}&|I^{SR}(w,h,c) - I^{HR}(w,h,c)| \\ &\times(1 - M_{out}^D(I^{SR})(w,h,c)) \end{aligned}$$

where H, W, and C are the height, width, and number of channels of the input image. For the discriminator, we have

$$\begin{aligned} L_{Correction}^D = &-\mathbb{E}[\log(M_{out}^D(I^{HR}) - \mathbb{E}[M_{out}^D(G(I^{LR}))])] \\ &-\mathbb{E}[\log(1 - M_{out}^D(G(I^{LR})) + \mathbb{E}[M_{out}^D(I^{HR})])]. \end{aligned}$$

In summary, the generator's GAN loss is defined as

$$L_{adv}^G = L_{Multi-headed}^G + L_{Correction}^G. \tag{7}$$

D. Multi-perceptual Loss and Loss Function

1) Pixel Loss: Pixel loss is an important evaluation metric in image processing tasks as it measures the difference in details between SR image and HR image. In the first stage of generator training, a pre-trained model is generated using L1 loss as Pixel Loss, which is robust to outliers and can avoid noise interference.

$$L_{Pixel} = \frac{1}{WHC}\sum_{w=1}^{W}\sum_{h=1}^{H}\sum_{c=1}^{C}|G(I^{LR})(w,h,c) - I^{HR}(w,h,c)|$$

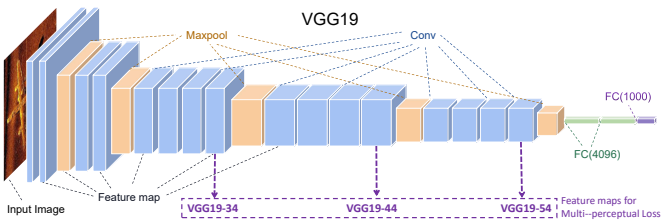

Fig. 3. Multi-perceptual implementation.

2) Multi-perceptual Loss: Pixel loss only considers the error between image pixels and may produce perceptually unsatisfactory images. To address this issue, we introduce Perceptual Loss [62] to help generate more realistic images by considering the errors between features. In previous works such as [35]–[37], [57] the feature maps of the 11th convolutional layer (before the ReLU activation layer) are chosen, VGG19-54 [63] are chosen. Here, VGG19-ij indicates the feature map of the j-th convolution before activation and the i-th maxpooling layer for the VGG19 network. However, deep feature maps alone may not reflect the similarity between the compared images well. Therefore, in this paper, we select multiple feature maps including shallow features after VGG19-34, medium features after VGG19-44, and deep features after VGG19-54, as shown in Fig. 3.

$$L_{Perc}^{f} = \frac{1}{W_f H_f C_f} \sum_{w=1}^{W_f} \sum_{h=1}^{H_f} \sum_{c=1}^{C_f} |F_{f4}^{VGG}(G(I^{LR}))(w,h,c) - F_{f4}^{VGG}(I^{HR})(w,h,c)|$$

where $F_{f4}^{VGG}(\cdot)$ is the output feature map of VGG19-$f4$ layer(f = 3, 4, 5). H_f, W_f, and C_f are the height, width, and number of channels of F_{f4}^{VGG}, respectively. In this paper, Multi-Perceptual Loss is defined as

$$L_{Multi-Perc} = \lambda_1 L_{Perc}^{3} + \lambda_2 L_{Perc}^{4} + \lambda_3 L_{Perc}^{5} \quad (8)$$

where λ_1, λ_2, and λ_3 are the weighting parameters.

3) Loss function: The overall loss functions for the generator and discriminator are defined as

$$L^{G} = \lambda L_{Multi-Perc} + \mu L_{Pixel} + \gamma L_{adv}^{G} \quad (9)$$

$$L^{D} = L_{Multi-headed}^{D} + L_{Correction}^{D} \quad (10)$$

Where λ, μ, and γ are the weighting parameters to balance different loss terms.

IV. EXPERIMENTS

A. Datasets

Due to the limited availability of publicly accessible underwater sonar datasets, we collect underwater sonar images from various scenarios [5], [11], [64] and create the dataset named Underwater Sonar Dataset for Super-Resolution (USDSR). Therefore, to demonstrate the broad applicability of our method to underwater sonar datasets, we select two datasets KLSG–II underwater side-scan sonar image dataset [2] and USDSR. Due to the low quality of underwater sonar images,

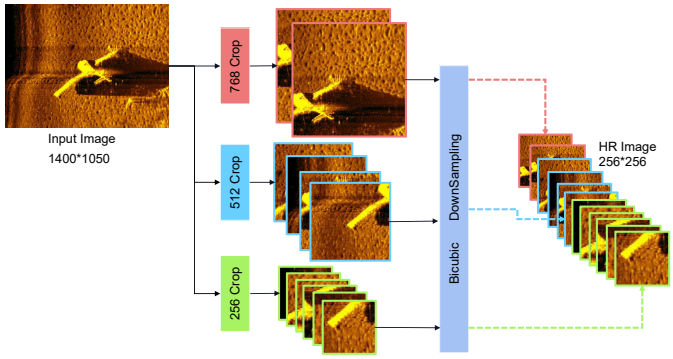

Fig. 4. Multi-scale cropping with downsampling.

we applied several pre-processing techniques to ensure adequate training samples and image clarity. Firstly, for KLSG-II, by unsharp masking (USM) to increase edge contrast and multi-scale cropping with downsampling, as illustrated in Fig. 4, we obtain 1500 128×128 images, including aircraft, seafloor, and shipwreck. Secondly, with similar pre-processing operations, the USDSR contains 1500 images, comprising three classes of aircraft, shipwreck, and human, each with a size of 256×256. Finally, we respectively select 6 and 60 images from each dataset as Set6 and Set60 for testing.

B. Implementation Details

Experiments are carried out by three scale factors: r = 2, 4, and 8, respectively, which increase the number of pixels by 4, 16, and 64. We obtain LR images by bicubic downsampling. Additionally, for each dataset, we normalize both the input HR and the LR images to [0,1], select 1400 images for training, and test the proposed method with Set6 and Set60, respectively.

The proposed MHGAN method is implemented on the Pytorch framework and trained on three NVIDIA GeForce RTX 3090 GPUs. Specifically, the training process is divided into two phases. In the first stage, the generator is trained with pixel loss only, generating a PSNR-oriented model, resulting in the $\mathrm{MHGAN_{PSNR}}$ model. In the second stage, the MHGAN is trained by combining the pixel loss, multi-perceptual loss, and GAN loss introduced in Section III, initializing the second-stage generator based on the per-trained $\mathrm{MHGAN_{PSNR}}$ model. The training process is with Adam [65] for optimization, where β_1 = 0.9 and β_2 = 0.999. The learning rate is initialized as 0.0002. $\lambda_1 = 0.2, \lambda_2 = 0.3$, and $\lambda_3 = 0.5$ in (8) and $\lambda = 1$, $\mu = 1$, and $\gamma = 1e-2$ are set for training in (9). The batch size is set to 8.

We select three widely used evaluation metrics in the field of super-resolution images for this paper, including PSNR [50], SSIM [51], and learned perceptual image patch similarity (LPIPS) [66]. A higher PSNR value indicates better image quality. The closer the SSIM value is to 1, the more similar the image is. A lower LPIPS value indicates better SR results.

C. Experimental Results

1) Parameters Comparison With State-of-Art SR Methods: Table I presents the results of our experiments on the

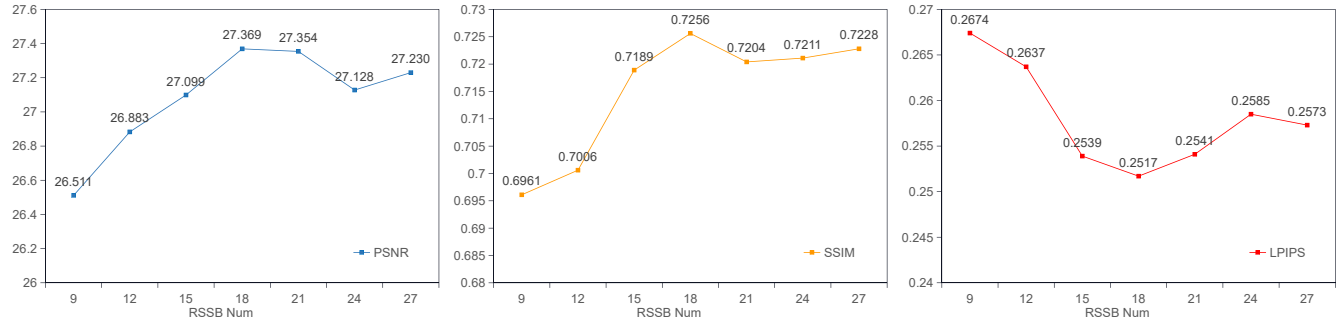

Fig. 5. Comparison results of PSNR, SSIM, and LPIPS with different numbers of RSSBs.

TABLE I
QUANTITATIVE COMPARISON OF PARAMETERS FOR DIFFERENT METHODS IN EACH SCALE FACTOR

Methods	Scale factor r		
	r=2	r=4	r=8
SRResNet	1,369,859	1,517,571	1,665,283
EDSR	40,729,603	43,089,923	45,450,243
RRDB	16,661,059	16,697,987	16,734,915
RCAN	15,444,643	15,592,355	15,740,067
SwinIR	11,752,487	11,900,199	12,047,911
$\text{MHGAN}_{\text{PSNR}}$	7,347,395	7,495,107	7,642,819
SRGAN	1,369,859 21,055,049	1,517,571 21,055,049	1,665,283 21,055,049
ESRGAN	16,661,059 21,055,049	16,697,987 21,055,049	16,734,915 21,055,049
MHGAN(Ours)	7,347,395 11,275,715	7,495,107 11,275,715	7,642,819 11,275,715

$\text{MHGAN}_{\text{PSNR}}$ (PSNR-oriented model), MHGAN, and state-of-the-art methods at scale factors of r = 2, 4, and 8, with a focus on the number of parameters used. For GAN-based methods, we report the number of generator parameters in the upper part and the number of discriminator parameters in the lower part. Table I demonstrates that our $\text{MHGAN}_{\text{PSNR}}$ model requires more parameters than SRResNet, but significantly fewer parameters than other methods. In addition, the number of discriminator parameters used in MHGAN is considerably less than those in SRGAN and ESRGAN.

2) Evaluation The Number Of RSSBs: In our proposed method, RSSB is a critical component of the generator model and has a significant impact on the quality of the generated SR images. Therefore, experiments are conducted to explore the effect of network depth by comparing the SR performance of the generator with varying numbers of RSSBs in the generator. Specifically, The number of RSSBs is set to $N \in \{9, 12, 15, 18, 21, 24, 27\}$, and PSNR, SSIM, and LPIPS are used as metrics for quantitative comparison. As shown in Fig. 5, the model's ability to extract features increases as N increases, leading to better results on the three metrics when N is less than 18. However, as the model's depth continues to increase, the difficulty of training increases, resulting in no significant improvement in the performance when N is more than 18. The best results in the three metrics of PSNR, SSIM, and LPIPS are obtained using 18 RSSBs. Therefore, N is set to 18 for the following experiments.

3) Comparison With State-of-the-Art SR Methods: We present a comparison of our proposed MHGAN model with several state-of-the-art SR networks, including EDSR [31], SRResNet [35], SRGAN [35], RRDB [36], ESRGAN [36], RCAN [42], and SwinIR [34], using bicubic interpolation [67] as the baseline. While models based on deep convolutional neural networks, such as EDSR, MSRResNet, RRDB, and RCAN, minimize L1 loss, SRGAN, and ESRGAN are GAN-based models that incorporate perceptual and adversarial generative losses in their loss functions. SwinIR, based on Swin Transformer [68], uses both L1 loss and a transformer-based architecture.

For a fair comparison, all the above models are trained and tested on the same dataset and environment. It is worth noting that the models that do not utilize the GAN approach produce better results for measures such as PSNR, but adding GAN to the model restores some perceptually pleasing texture detail, resulting in lower PSNR values. In our experiments, we compare the $\text{MHGAN}_{\text{PSNR}}$ method (PSNR-oriented) of the MHGAN model with all the above methods, as well as the MHGAN model with SRGAN and ESRGAN.

Table II presents a quantitative comparison of our $\text{MHGAN}_{\text{PSNR}}$ method on the USDSR dataset with scale factors of 2, 4, and 8, respectively. We also report the average results for the three evaluation measures, PSNR, SSIM, and LPIPS, across Set6 and Set60. The best values for each evaluation metric for each scene are marked in bold. Notably, in each scaling, our $\text{MHGAN}_{\text{PSNR}}$ method outperforms other advanced methods for Set6 and Set60 the USDSR dataset and achieves a significantly better mean of each evaluation criterion. Moreover, our model significantly outperforms other methods for Set6 at $r = 2$, achieving the average PSNR of 32.42, and producing results that are almost indistinguishable from corresponding HR images. When the scale factor is 8, which is a more challenging task, the performance of all the methods declines significantly. But our model still performs well and achieves a PSNR value of 27.74 on Set6. The LPIPS score is also reduced to nearly 0.2, demonstrating a significant advantage over other advanced algorithms. Table II clearly demonstrates the advantages of our proposed $\text{MHGAN}_{\text{PSNR}}$ method for each test set on the USDSR dataset.

Tables III-IV indicate the quantitative comparison between

TABLE II
QUANTITATIVE COMPARISON FOR EACH SCALING ON USDSR DATASET

Scale factor	Testing Sets	Evaluation Metrics	Bicubic	MSRResNet	SRGAN	EDSR	RRDB	ESRGAN	RCAN	SwinIR	$\mathrm{MHGAN}_{\mathrm{PSNR}}$
×2	Set6	PSNR	28.425	29.774	26.061	29.688	29.837	28.196	29.819	30.179	**32.418**
		SSIM	0.7327	0.7674	0.6144	0.7657	0.7783	0.6886	0.7905	0.8084	**0.8816**
		LPIPS	0.2672	0.2506	0.2255	0.2479	0.2320	0.1867	0.2075	0.2119	**0.1227**
	Set60	PSNR	27.541	28.859	25.335	28.785	28.833	26.682	28.745	28.877	**29.613**
		SSIM	0.7407	0.7739	0.6287	0.7720	0.7775	0.6787	0.7787	0.7844	**0.7920**
		LPIPS	0.2995	0.2862	0.2417	0.2826	0.2709	0.2173	0.2500	0.2570	**0.1811**
×4	Set6	PSNR	24.925	26.145	23.765	26.126	26.258	23.925	26.517	28.003	**30.811**
		SSIM	0.5501	0.5748	0.4742	0.5725	0.5921	0.4896	0.6177	0.7846	**0.8598**
		LPIPS	0.4614	0.4336	0.2984	0.4417	0.3977	0.3015	0.3696	0.2330	**0.1240**
	Set60	PSNR	24.839	26.042	23.308	25.966	25.875	23.575	25.924	26.544	**26.956**
		SSIM	0.5599	0.5816	0.4765	0.5800	0.5891	0.4945	0.5929	0.6440	**0.6673**
		LPIPS	0.4976	0.4769	0.3097	0.4797	0.4522	0.3251	0.4167	0.3565	**0.2528**
×8	Set6	PSNR	23.289	24.329	22.736	24.347	24.542	23.505	24.728	25.762	**27.740**
		SSIM	0.4077	0.4237	0.3451	0.4236	0.4369	0.4054	0.4619	0.5934	**0.7247**
		LPIPS	0.6009	0.5390	0.3794	0.5440	0.4987	0.3736	0.4532	0.3054	**0.2037**
	Set60	PSNR	21.996	23.180	20.878	23.037	23.242	21.628	23.172	23.502	**24.051**
		SSIM	0.4118	0.4324	0.3442	0.4296	0.4381	0.3968	0.4417	0.4827	**0.5126**
		LPIPS	0.6410	0.5685	0.3728	0.5713	0.5405	0.3694	0.4886	0.4159	**0.3234**

TABLE III
QUANTITATIVE COMPARISON FOR GAN BASED METHODS ON KLSG-II DATASET

KLSG		SRGAN			ESRGAN			MHGAN(Ours)		
Scale factor	Class	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
r=2	Shipwreck	28.209	0.7966	0.1600	29.145	0.8276	0.1530	**29.641**	**0.8356**	**0.1520**
	Aircraft	28.555	0.7800	0.1458	29.244	0.8038	0.1501	**35.054**	**0.9364**	**0.0464**
	Seafloor	22.707	0.5652	0.2510	23.023	0.5625	0.2433	**24.514**	**0.6765**	**0.2256**
	Set6	24.604	0.6961	0.1909	24.809	0.7028	0.1934	**28.127**	**0.8180**	**0.1276**
	Set60	26.433	0.7167	0.1880	27.121	0.7353	0.1826	**28.834**	**0.7994**	**0.1589**
r=4	Shipwreck	23.866	0.5667	0.4035	24.516	0.5843	**0.3140**	**25.595**	**0.6321**	0.3141
	Aircraft	24.812	0.5625	0.2613	25.136	0.5648	0.2932	**30.386**	**0.8326**	**0.1013**
	Seafloor	19.962	0.2297	0.3691	20.082	0.2254	**0.3664**	**21.261**	**0.3122**	0.3780
	Set6	21.187	0.3971	0.3114	21.478	0.3911	0.3288	**23.993**	**0.5635**	**0.2547**
	Set60	22.722	0.4537	0.3185	23.141	0.4614	0.3280	**24.949**	**0.5589**	**0.2999**
r=8	Shipwreck	21.267	0.4310	**0.4035**	22.060	0.4466	0.4171	**22.958**	**0.4844**	0.4327
	Aircraft	22.804	0.4571	0.3431	22.278	0.4247	0.4130	**27.617**	**0.7236**	**0.1482**
	Seafloor	18.920	0.1439	**0.4152**	19.261	0.1520	0.4214	**20.155**	**0.1685**	0.4208
	Set6	19.714	0.2718	0.3908	20.019	0.2722	0.4172	**21.874**	**0.3997**	**0.3514**
	Set60	20.741	0.3396	0.3973	21.163	0.3448	0.4178	**22.800**	**0.4190**	**0.3813**

our MHGAN method and the GAN-based approach on the KLSG and USDSR datasets with scale factors of 2, 4, and 8, respectively. The best values for each evaluation metric for each scene are marked in bold. In Table III, our proposed MHGAN method achieves significantly better results for each evaluation metric on the Set6 and Set60 test set than both SRGAN and ESRGAN methods. In the quantitative comparison, our method outperforms other GAN-based methods. Similarly, in Table IV, our MHGAN method significantly outperforms other methods on each evaluation criterion in each category on the USDSR dataset. At $r = 8$, the mean PSNR of Set60 can be 1.2 higher. These tables clearly demonstrate the superiority of our MHGAN method in qualitative performance on the KLSG and USDSR datasets.

4) Visual Comparison With State-of-the-Art SR Methods: Fig. 6 displays the SR visualization of MHGAN and other compared methods on the KLSG dataset with a scale factor of $r = 4$. Images belonging to the Aircraft and Shipwreck categories are selected, as they contain obvious targets and backgrounds. Upon inspection, the images generated by bicubic interpolation are found to be excessively smooth, resulting in the loss of high-frequency information. ESRGAN, a GAN-based method, is able to restore texture to some extent, but the resulting images contained unexpected artifacts that are not present in the original image. Moreover, the overall color of the ESRGAN images is somewhat different from that of the other images. The SwinIR method is able to restore certain parts of the target, but some regions appeared blurry. In contrast, the images generated by MHGAN are effective in restoring both the target and background, and the overall texture is closer to the Ground Truth. Grayscale images are easier to recover compared to colored ones, thus the disparity between the Ground Truth and the SR images is minimal.

Figs. 7-9 present the visual comparison of our proposed $\mathrm{MHGAN}_{\mathrm{PSNR}}$ and MHGAN with other methods at different scale factors of 2, 4, and 8, respectively. Each generated

TABLE IV
QUANTITATIVE COMPARISON FOR GAN BASED METHODS ON USDSR DATASET

USDSR		SRGAN			ESRGAN			MHGAN(Ours)		
Scale factor	Class	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
r=2	Shipwreck	24.242	0.6084	0.2498	25.394	0.6457	0.2296	**26.681**	**0.7103**	**0.2176**
	Aircraft	25.258	0.6322	0.2370	26.830	0.6860	0.2047	**28.082**	**0.7769**	**0.1839**
	Human	30.413	0.7130	0.2147	32.177	0.8130	0.1870	**34.923**	**0.9496**	**0.0921**
	Set6	26.061	0.6144	0.2255	28.196	0.6886	0.1867	**29.876**	**0.8221**	**0.1772**
	Set60	25.335	0.6287	0.2417	26.682	0.6787	0.2173	**28.154**	**0.7600**	**0.1918**
r=4	Shipwreck	22.177	0.4460	0.3177	22.428	0.4632	0.3371	**24.313**	**0.5463**	**0.3134**
	Aircraft	23.268	0.4767	0.3115	23.761	0.4962	0.3212	**25.472**	**0.6141**	**0.2904**
	Human	28.278	0.5138	0.2701	28.365	0.5324	0.2792	**32.613**	**0.8782**	**0.1062**
	Set6	23.765	0.4743	0.2984	23.925	0.4896	0.3015	**28.601**	**0.7004**	**0.2016**
	Set60	23.308	0.4765	0.3097	23.575	0.4945	0.3251	**25.729**	**0.6086**	**0.2797**
r=8	Shipwreck	19.661	0.3115	0.3818	20.420	0.3662	0.3729	**21.430**	**0.3883**	**0.3683**
	Aircraft	20.932	0.3400	0.3783	22.203	0.4068	0.3765	**22.522**	**0.4425**	**0.3477**
	Human	26.247	0.5003	0.3215	25.912	0.5143	0.3395	**29.551**	**0.7054**	**0.1326**
	Set6	22.736	0.3451	0.3794	23.505	0.4054	0.3736	**24.451**	**0.4987**	**0.2483**
	Set60	20.878	0.3443	0.3728	21.628	0.3968	0.3694	**22.804**	**0.4450**	**0.3314**

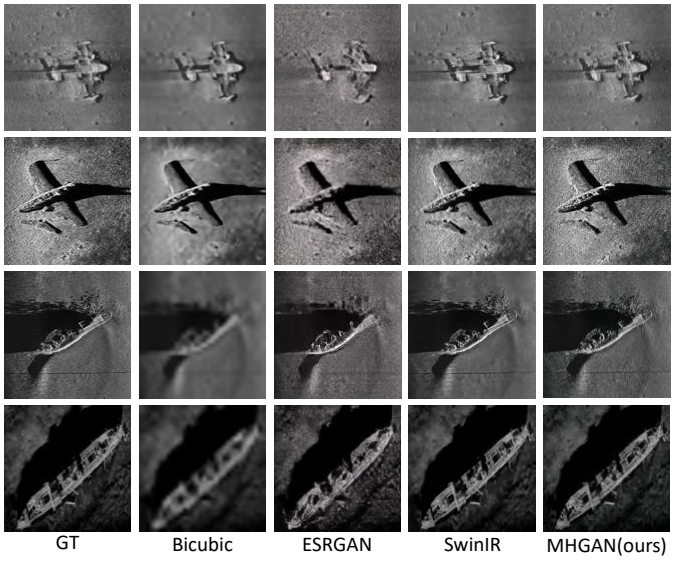

Fig. 6. Comparison of SR visualization for different methods on KLSG dataset with scale factor $r = 4$.

image is highlighted with a red box and a blue box. The red box denotes the focused region of the object, showing the superiority of the proposed method in detail in the recovery of the object compared to other methods. The blue box represents the focused area of the background, emphasizing the difference between our approach and others in recovering complex underwater background textures.

For the scale factor $r = 2$, we select an aircraft and a shipwreck with high-level object details. Other methods that are not based on GAN tend to produce blurring in small regions due to the slight magnification. Their focus is limited to the convolutional kernel size, resulting in nonuniform object and background recovery. In contrast, our proposed $\mathrm{MHGAN}_{\mathrm{PSNR}}$ and MHGAN methods use SCAB to fuse features for each image, acquire long-range features, and expand the perceptual field. MHGAN uses Multi-headed to guide the image recovery at multiple scales, enabling the recovery of object details and background textures and producing images that more closely resemble the HR image in terms of textures. Moreover, due to the smaller magnification, $\mathrm{MHGAN}_{\mathrm{PSNR}}$ generates detailed information easily by using global information guided by the L1 loss. Therefore, the difference between the generated images of $\mathrm{MHGAN}_{\mathrm{PSNR}}$ and MHGAN is negligible when $r = 2$.

For the scale factor of $r = 4$, the difficulty in recovery increases, and the generated images of some methods become more blurred. The overall recovery of objects becomes more challenging, making the details harder to recover. The methods which are not based on GAN show obvious difficulties in generating background in detail. SRGAN and ESRGAN can recover object and background details appropriately, but they often produce unexpected artifacts due to the complex background of sonar images. In contrast, our proposed method generates appropriate textures with fewer unexpected artifacts. Additionally, the background recovery of MHGAN is more effective than the $\mathrm{MHGAN}_{\mathrm{PSNR}}$ method, which demonstrates the effectiveness of the discriminator with the multi-headed module.

For the scale factor of $r = 8$, as image generation becomes more complex, we choose shipwreck and human with less detail for comparison. Other methods produce severely blurred images with unwanted details due to the complex data. In contrast, our proposed method is closer to the actual image, but recovering tiny details of the object and the overall texture of the background is still challenging. For the reconstruction of the target and background, GAN-based methods show their ability in recovering texture and details.

D. Ablation Studies

We conduct ablation experiments on our proposed generator, discriminator, and loss function. For the generator, our proposed MHGAN contains three main modules: Self-calibrated Net (SC-net), SCAB, and UAB. To illustrate the role of these modules, we explore various combinations of SC-net, SCAB, and UAB in scaling factor $r = 4$, and the

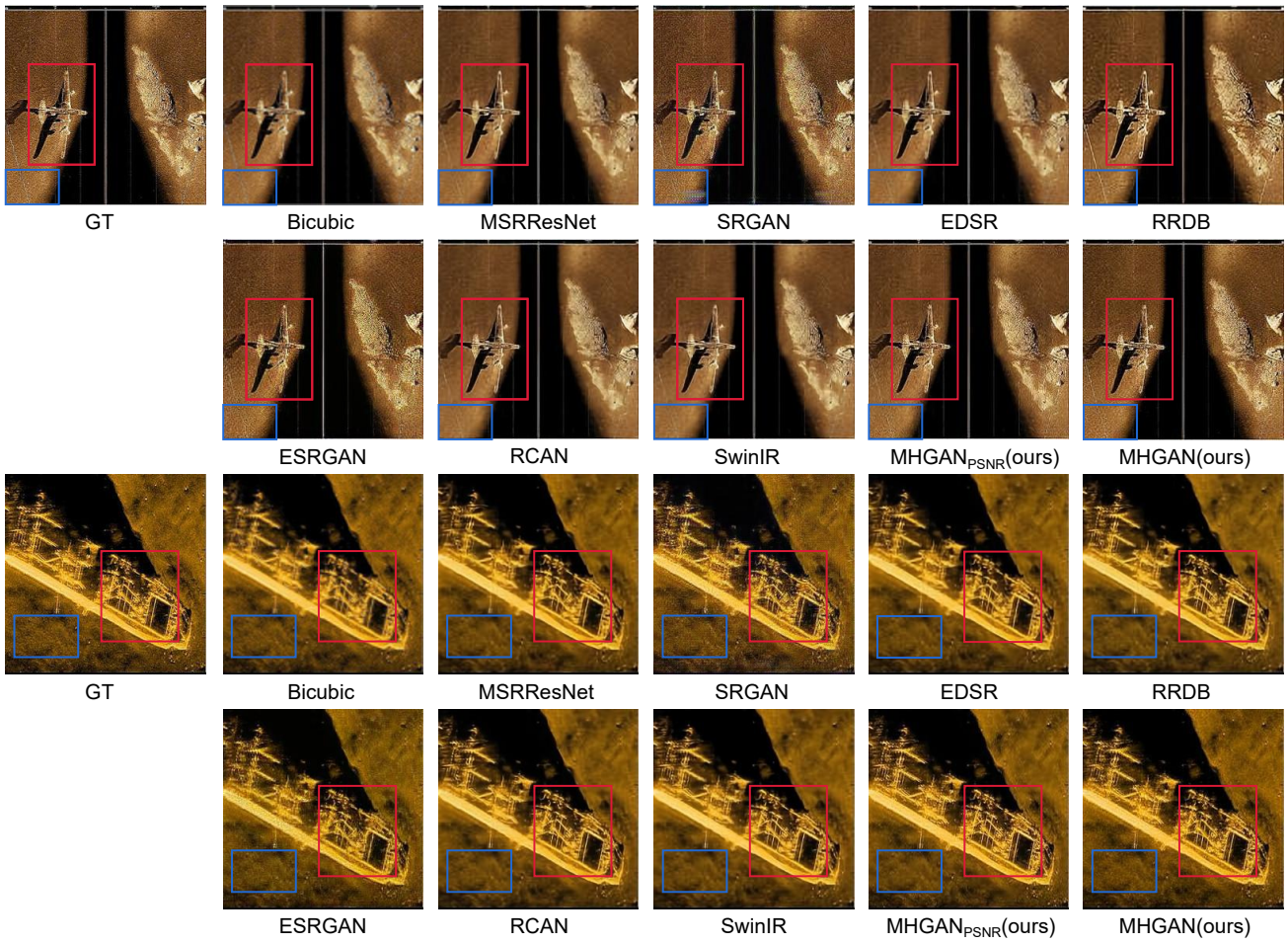

Fig. 7. Comparison of SR visualization for different methods on USDSR dataset with scale factor $r = 2$.

TABLE V
QUANTITATIVE COMPARISON OF THE EFFECT OF EACH COMPONENT IN SELF-CALIBRATED ATTENTION GENERATOR

SC-net	SCAB	UAB	PSNR	SSIM	LPIPS
			26.991	0.7099	0.3145
✓			27.087	0.7112	0.3097
	✓		27.160	0.7135	0.3042
		✓	27.105	0.7138	0.3040
✓	✓		27.499	0.7243	0.3067
✓		✓	27.230	0.7127	**0.2950**
	✓	✓	27.592	0.7233	0.3032
✓	✓	✓	**27.709**	**0.7259**	0.3002

TABLE VI
QUANTITATIVE COMPARISON OF THE EFFECT OF DIFFERENT LOSS FUNCTION

Configuration	PSNR	SSIM	LPIPS
VGG Style(Normal)	24.610	0.5383	0.2909
VGG Style_Multi-perc	25.770	0.6109	**0.2603**
Unet Style	25.663	0.5880	0.2896
Unet_Multi-perc	26.044	0.6379	0.2777
Unet_Multihead	26.312	0.6123	0.3115
Unet_Multihead_Multi-perc	26.532	0.6574	0.2795
Unet_Multihead(Correct)_Multi-perc	**26.729**	**0.6586**	0.2797

results are evaluated based on three criteria, as presented in Table V. The findings indicate that all three modules contribute to enhancing the performance of the SR task. This ablation experiment demonstrates the effectiveness of our proposed SC-net, SCAB, and UAB. In addition, adding our designed SCAB module significantly improves the model compared to the other components. And the combination of these modules in our MHGAN leads to a great improvement in three criteria and better visual performance.

Furthermore, to validate the effectiveness of the proposed Multi-headed loss, Correction loss, and Multi-perceptual loss, ablation studies are performed on the loss function. In the experiment, we also replace the VGG Style with U-net in the discriminator as a baseline. Table VI summarizes the results, where we compare the discriminator based on the traditional VGG Style and examine various combinations of the three components in the loss function. Each innovative component demonstrates some improvement in all three criteria. Thus, the ablation experiments confirm the effectiveness of the discriminator and loss function utilized in MHGAN. Among them, Multi-perceptual loss has a significant improvement effect on the LPIPS criterion. Furthermore, Multi-headed loss plays an important role in PSNR and SSIM improvements.

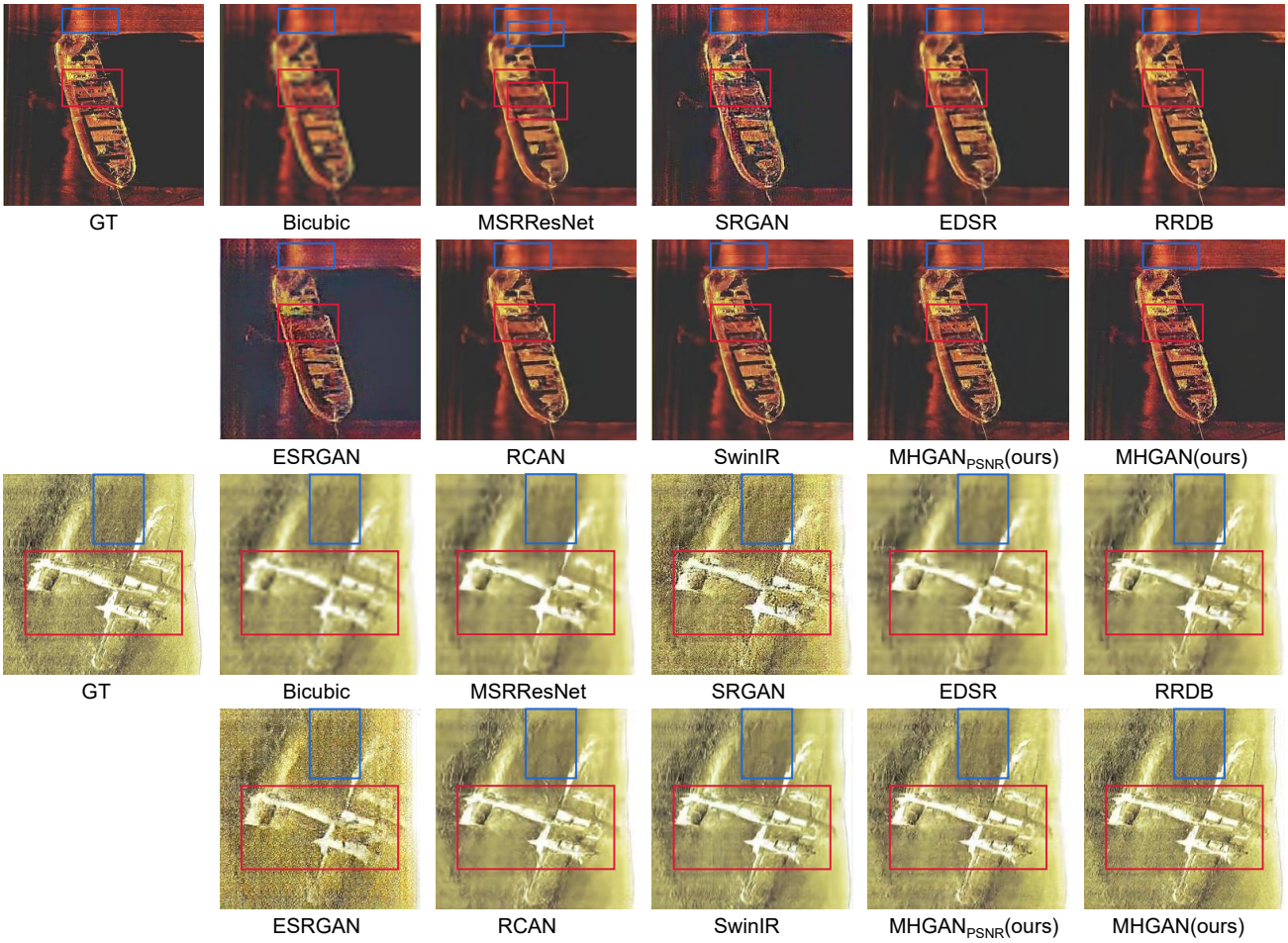

Fig. 8. Comparison of SR visualization for different methods on USDSR dataset with scale factor $r = 4$.

V. CONCLUSION

This paper proposes a novel super-resolution method called MHGAN for enhancing underwater sonar images, which can be extended to other small datasets with complex backgrounds. For the generator, we design a novel architecture, RSSB containing SD-net, SC-net, and SCAB, as the main body of the generator to improve its generalization ability. And we embed Pixel Attention into the upsampling module. Additionally, we propose a Multi-headed Discriminator that facilitates restoring accurate details and realistic textures from multiple scales. To guide the learning process of the generator, we also introduce the Multi-perceptual loss and GAN loss with correction in the loss function. Our experimental results show that our proposed $\mathrm{MHGAN}_{\mathrm{PSNR}}$ and MHGAN methods outperform current state-of-the-art methods both quantitatively and visually. In the future, this work can be further extended to tasks such as image denoising with GAN and blind super-resolution.

REFERENCES

[1] X. Ye, X. Ge, and H. Yang, "A gray scale correction method for side-scan sonar images based on gan," in *Global Oceans 2020: Singapore – U.S. Gulf Coast*, 2020, pp. 1–5.

[2] G. Huo, Z. Wu, and J. Li, "Underwater object classification in sidescan sonar images using deep transfer learning and semisynthetic training data," *IEEE Access*, vol. 8, pp. 47 407–47 418, 2020.

[3] Z. Wang, J. Guo, L. Zeng, C. Zhang, and B. Wang, "Mlffnet: Multilevel feature fusion network for object detection in sonar images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–19, 2022.

[4] T. Zhou, J. Si, L. Wang, C. Xu, and X. Yu, "Automatic detection of underwater small targets using forward-looking sonar images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–12, 2022.

[5] P. Zhang, J. Tang, H. Zhong, M. Ning, D. Liu, and K. Wu, "Self-trained target detection of radar and sonar images using automatic deep learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2022.

[6] H. Xu, L. Zhang, M. J. Er, and Q. Yang, "Underwater sonar image segmentation based on deep learning of receptive field block and search attention mechanism," in *2021 4th International Conference on Intelligent Autonomous Systems (ICoIAS)*, 2021, pp. 44–48.

[7] Z. Wang, J. Guo, W. Huang, and S. Zhang, "Side-scan sonar image segmentation based on multi-channel fusion convolution neural networks," *IEEE Sensors Journal*, vol. 22, no. 6, pp. 5911–5928, 2022.

[8] J. Li, P. Jiang, and H. Zhu, "A local region-based level set method with markov random field for side-scan sonar image multi-level segmentation," *IEEE Sensors Journal*, vol. 21, no. 1, pp. 510–519, 2021.

[9] M. Machado Dos Santos, G. G. De Giacomo, P. L. J. Drews, and S. S. C. Botelho, "Matching color aerial images and underwater sonar images using deep learning for underwater localization," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6365–6370, 2020.

[10] T. Zhou, Y. Wang, B. Chen, J. Zhu, and X. Yu, "Underwater multitarget tracking with sonar images using thresholded sequential monte carlo probability hypothesis density algorithm," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.

[11] T. T. C, A. M. Nambiar, and A. Mittal, "A gan-based super resolution

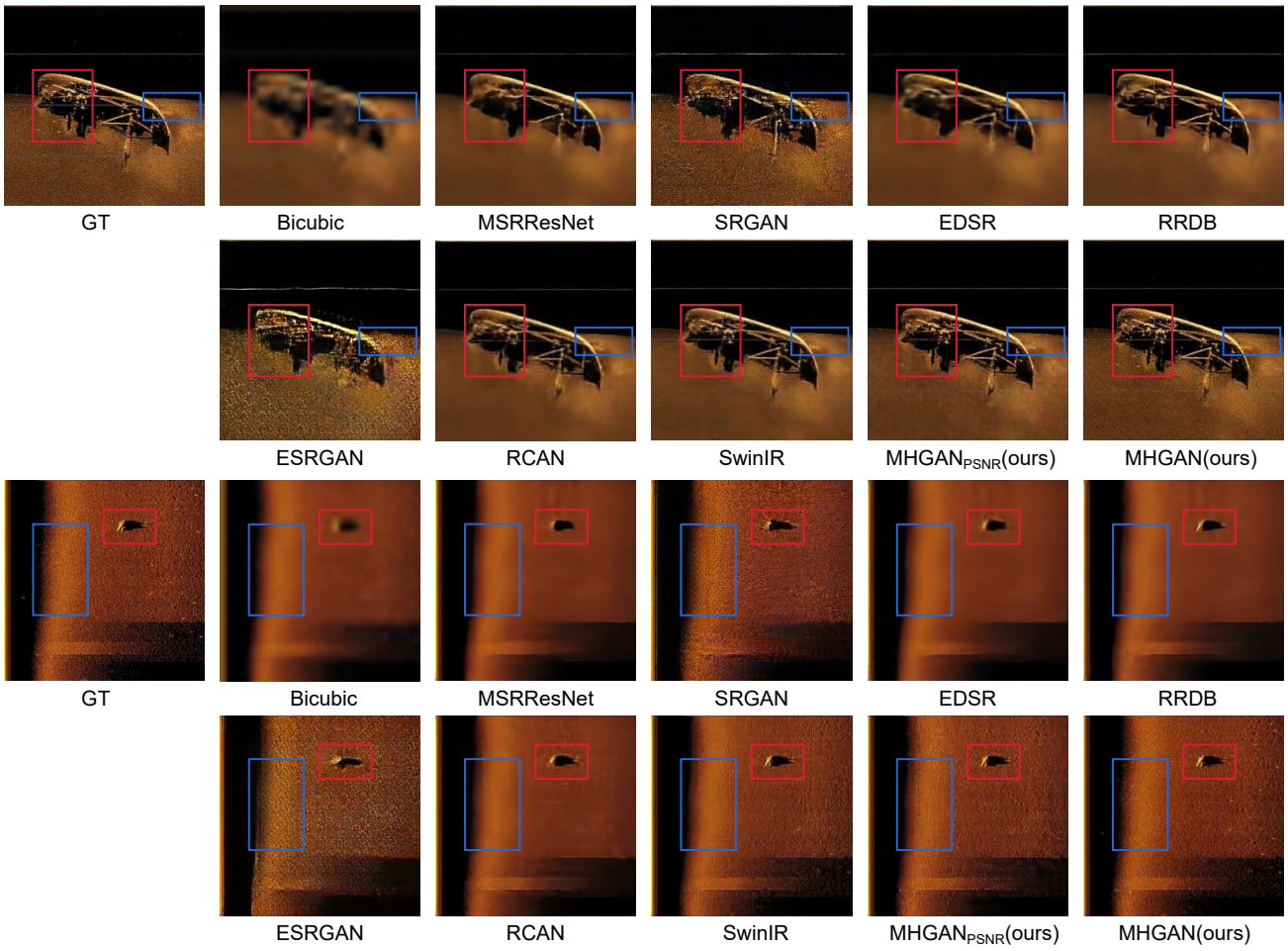

Fig. 9. Comparison of SR visualization for different methods on USDSR dataset with scale factor $r = 8$.

model for efficient image enhancement in underwater sonar images," in *OCEANS 2022 - Chennai*, 2022, pp. 1–8.

[12] W. Chen, K. Gu, W. Lin, F. Yuan, and E. Cheng, "Statistical and structural information backed full-reference quality measure of compressed sonar images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 2, pp. 334–348, 2020.

[13] W. Freeman and E. Pasztor, "Learning low-level vision," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, 1999, pp. 1182–1189 vol.2.

[14] Y. Yu, X. Li, and F. Liu, "E-dbpn: Enhanced deep back-projection networks for remote sensing scene image superresolution," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 8, pp. 5503–5515, 2020.

[15] Q. Cai, J. Li, H. Li, Y.-H. Yang, F. Wu, and D. Zhang, "Tdpn: Texture and detail-preserving network for single image super-resolution," *IEEE Transactions on Image Processing*, vol. 31, pp. 2375–2389, 2022.

[16] Q. Jiang, Z. Liu, K. Gu, F. Shao, X. Zhang, H. Liu, and W. Lin, "Single image super-resolution quality assessment: A real-world dataset, subjective studies, and an objective metric," *IEEE Transactions on Image Processing*, vol. 31, pp. 2279–2294, 2022.

[17] D. Jin, M. Ji, L. Xu, G. Wu, L. Wang, and L. Fang, "Boosting single image super-resolution learnt from implicit multi-image prior," *IEEE Transactions on Image Processing*, vol. 30, pp. 3240–3251, 2021.

[18] J. Cheng, Q. Kuang, C. Shen, J. Liu, X. Tan, and W. Liu, "Reslap: Generating high-resolution climate prediction through image super-resolution," *IEEE Access*, vol. 8, pp. 39 623–39 634, 2020.

[19] X. Bing, W. Zhang, L. Zheng, and Y. Zhang, "Medical image super resolution using improved generative adversarial networks," *IEEE Access*, vol. 7, pp. 145 030–145 038, 2019.

[20] P. G. Daneshmand, H. Rabbani, and A. Mehridehnavi, "Super-resolution of optical coherence tomography images by scale mixture models," *IEEE Transactions on Image Processing*, vol. 29, pp. 5662–5676, 2020.

[21] D.-H. Trinh, M. Luong, F. Dibos, J.-M. Rocchisani, C.-D. Pham, and T. Q. Nguyen, "Novel example-based method for super-resolution and denoising of medical images," *IEEE Transactions on Image Processing*, vol. 23, no. 4, pp. 1882–1895, 2014.

[22] J. Chen, J. Chen, Z. Wang, C. Liang, and C.-W. Lin, "Identity-aware face super-resolution for low-resolution face recognition," *IEEE Signal Processing Letters*, vol. 27, pp. 645–649, 2020.

[23] C. Chen, D. Gong, H. Wang, Z. Li, and K.-Y. K. Wong, "Learning spatial attention for face super-resolution," *IEEE Transactions on Image Processing*, vol. 30, pp. 1219–1231, 2021.

[24] C. E. Duchon, "Lanczos filtering in one and two dimensions," *Journal of Applied Meteorology and Climatology*, vol. 18, no. 8, pp. 1016–1022, 1979.

[25] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 6, pp. 1153–1160, 1981.

[26] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV 13*. Springer, 2014, pp. 184–199.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[28] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[29] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1646–1654.

[30] Kim, Jiwon and Lee, Jung Kwon and Lee, Kyoung Mu, "Deeply-recursive convolutional network for image super-resolution," in *2016*

IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 1637–1645.

[31] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 136–144.

[32] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1874–1883.

[33] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep back-projection networks for super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1664–1673.

[34] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, "Swinir: Image restoration using swin transformer," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 1833–1844.

[35] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4681–4690.

[36] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy, "ESRGAN: Enhanced super-resolution generative adversarial networks," in *Proceedings of the European conference on computer vision (ECCV) workshops*, 2018, pp. 0–0.

[37] Y. Yan, C. Liu, C. Chen, X. Sun, L. Jin, X. Peng, and X. Zhou, "Fine-grained attention and feature-sharing generative adversarial networks for single image super-resolution," *IEEE Transactions on Multimedia*, vol. 24, pp. 1473–1487, 2021.

[38] J.-J. Liu, Q. Hou, M.-M. Cheng, C. Wang, and J. Feng, "Improving convolutional networks with self-calibrated convolutions," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 096–10 105.

[39] H. Zhao, X. Kong, J. He, Y. Qiao, and C. Dong, "Efficient image super-resolution using pixel attention," in *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*. Springer, 2020, pp. 56–72.

[40] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer, 2016, pp. 391–407.

[41] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3147–3155.

[42] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 286–301.

[43] W. Lee, J. Lee, D. Kim, and B. Ham, "Learning with privileged information for efficient image super-resolution," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*. Springer, 2020, pp. 465–482.

[44] L. Wang, Y. Wang, Z. Lin, J. Yang, W. An, and Y. Guo, "Learning a single network for scale-arbitrary super-resolution," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 4801–4810.

[45] B. Niu, W. Wen, W. Ren, X. Zhang, L. Yang, S. Wang, K. Zhang, X. Cao, and H. Shen, "Single image super-resolution via a holistic attention network," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*. Springer, 2020, pp. 191–207.

[46] Y. Zhang, D. Wei, C. Qin, H. Wang, H. Pfister, and Y. Fu, "Context reasoning attention network for image super-resolution," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 4278–4287.

[47] J. Gu and C. Dong, "Interpreting super-resolution networks with local attribution maps," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9199–9208.

[48] X. Kong, X. Liu, J. Gu, Y. Qiao, and C. Dong, "Reflash dropout in image super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 6002–6012.

[49] S. Wang, J. Liu, K. Chen, X. Li, M. Lu, and Y. Guo, "Adaptive patch exiting for scalable single image super-resolution," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVIII*. Springer, 2022, pp. 292–307.

[50] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electronics letters*, vol. 44, no. 13, pp. 800–801, 2008.

[51] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *2010 20th international conference on pattern recognition*. IEEE, 2010, pp. 2366–2369.

[52] R. Lee, Ł. Dudziak, M. Abdelfattah, S. I. Venieris, H. Kim, H. Wen, and N. D. Lane, "Journey towards tiny perceptual super-resolution," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI*. Springer, 2020, pp. 85–102.

[53] K. C. Chan, X. Wang, X. Xu, J. Gu, and C. C. Loy, "GLEAN: Generative latent bank for large-factor image super-resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 14 245–14 254.

[54] W. Li, K. Zhou, L. Qi, L. Lu, and J. Lu, "Best-buddy GANs for highly detailed image super-resolution," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 2, 2022, pp. 1412–1420.

[55] J. Liang, H. Zeng, and L. Zhang, "Details or artifacts: A locally discriminative learning approach to realistic image super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5657–5666.

[56] S. Jia, Z. Wang, Q. Li, X. Jia, and M. Xu, "Multiattention generative adversarial network for remote sensing image super-resolution," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[57] Y. Li, S. Mavromatis, F. Zhang, Z. Du, J. Sequeira, Z. Wang, X. Zhao, and R. Liu, "Single-image super-resolution for remote sensing images using a deep generative adversarial network with local and global attention mechanisms," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–24, 2021.

[58] A. L. Maas, A. Y. Hannun, A. Y. Ng *et al.*, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, vol. 30, no. 1. Atlanta, Georgia, USA, 2013, p. 3.

[59] X. Wang, L. Xie, C. Dong, and Y. Shan, "Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1905–1914.

[60] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," *arXiv preprint arXiv:1802.05957*, 2018.

[61] A. Jolicoeur-Martineau, "The relativistic discriminator: a key element missing from standard GAN," *arXiv preprint arXiv:1807.00734*, 2018.

[62] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer, 2016, pp. 694–711.

[63] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[64] X. Ye, C. Li, S. Zhang, P. Yang, and X. Li, "Research on side-scan sonar image target classification method based on transfer learning," in *OCEANS 2018 MTS/IEEE Charleston*. IEEE, 2018, pp. 1–6.

[65] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[66] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

[67] K. Turkowski, "Filters for common resampling tasks," *Graphics gems*, pp. 147–165, 1990.

[68] 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," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.