Super-Resolution of Underwater Sonar Image based on Generative Adversarial Network

Zhengda Ma, Jie Ding, Sensen Li, Binbin Zou

***Abstract*—As an indispensable sensor for obtaining ocean resources, sonar can provide rich underwater observation information. However, due to the limitations of sonar equipment, the resolution of underwater sonar images collected is always low, resulting in some inexplicit underwater targets. To obtain high-resolution underwater information, image super-resolution reconstruction, as an efficient method, can significantly reduce hardware requirements. Super-resolution refers to the technology of restoring image details based on known image information. To achieve high-quality super-resolution for underwater sonar images, we propose a ConvGAN image super-resolution reconstruction model. We design the generator model with the improved ConvNeXt Block which enables the generator to generate rich details. At the same time, we use a VGG-Style discriminator and make its output a patch score map, which is used to guide the generator to recover complex texture details for sonar images. In addition, we compare with other state-of-the-art super-resolution networks on two different datasets. Quantitative and visual comparisons show the effectiveness of our proposed model.**

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

I. Introduction

With the increasing importance of ocean resource utilization, sonar technology, which utilizes sound wave propagation characteristics for detection, has been widely applied in detecting underwater targets, measuring water depth, and analyzing the marine environment [1]. However, the quality of underwater sonar imaging is not only limited by the hardware equipment of the sonar but also affected by environmental factors in the ocean [2]. Therefore, underwater sonar images are deeply affected by noise, resulting in low image quality. In order to conduct further processing of the obtained underwater sonar images, such as underwater target localization and recognition [3], [4], underwater organisms classification and tracking [5], [6], and submarine pipeline detection [7], it is necessary to perform image enhancement and super-resolution (SR) reconstruction on sonar images.

Super-resolution refers to the methods of reconstructing a high-resolution (HR) image from one or more low-resolution (LR) images [8]. Deep learning is commonly used to learn the mapping relationship between HR and LR images. SRCNN [9] introduces the convolutional neural network (CNN) into the SR field. With the development of deep learning neural networks, a series of SR algorithms based on deep neural networks, such as VDSR [10], EDSR [11], ESPCN [12], and RCAN [13], achieve good performance. However, the SR images generated by the SR methods based on the above CNN models often lack high-frequency details, which are unsatisfactory perceptions. Therefore, SRGAN [14] adopt the adversarial generative network (GAN) into the SR method to restore realistic textures. Since the introduction of SRGAN, GAN has brought prosperity to the SR field. ESRGAN [15] improves the generator network to the Residual in Residual Dense Block (RRDB) based on SRGAN. In remote sensing, MA-GAN [16] and SRAGAN [17] use different attention mechanisms to generate high-resolution remote sensing images. In addition, [18] brings the Swin Transformer into the SR field, using the Swin transformer layer for deep feature extraction, and achieving advanced performance in high-quality image reconstruction.

In recent years, super-resolution reconstruction of underwater sonar images mainly relies on the transfer learning approach, which involves fine-tuning existing pre-trained models on underwater sonar datasets. The pre-trained ESRGAN model is adopted as a basis for achieving super-resolution of sonar images [19]. However, underwater sonar images exhibit different textures and details compared with optical images, making the transfer learning technique less suitable for generating perceptually pleasing super-resolution images. Moreover, due to the scarcity of underwater sonar datasets, achieving super-resolution on limited image data presents significant challenges.

To address these issues, we propose a novel ConvGAN model tailored for the super-resolution reconstruction of underwater sonar images. First, we design a new generator network that leverages the ConvNeXt block [20] to extract deep features from underwater sonar images, thereby improving the model's generalization capability. Second, we employ an adversarial generative network that uses an improved relativistic discriminator to generate a patch score map. The patch score map helps to generate rich details and texture information by comparing features at the patch level. Additionally, we utilize a series of pre-processing operations, such as random cropping, multi-scale down-sampling, and unsharp masking (USM), to augment the underwater sonar dataset and enhance individual image quality for subsequent training. Finally, we achieve superior results to state-of-the-art super-resolution methods on underwater sonar datasets across three metrics.

The structure of this paper is as follows. Section II introduces the structure of the generator that we design. Section III

Z. Ma, J. Ding, and S. Li 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), dingjie@fudan.edu.cn (J. Ding), liss21@m.fudan.edu.cn (S. Li).

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

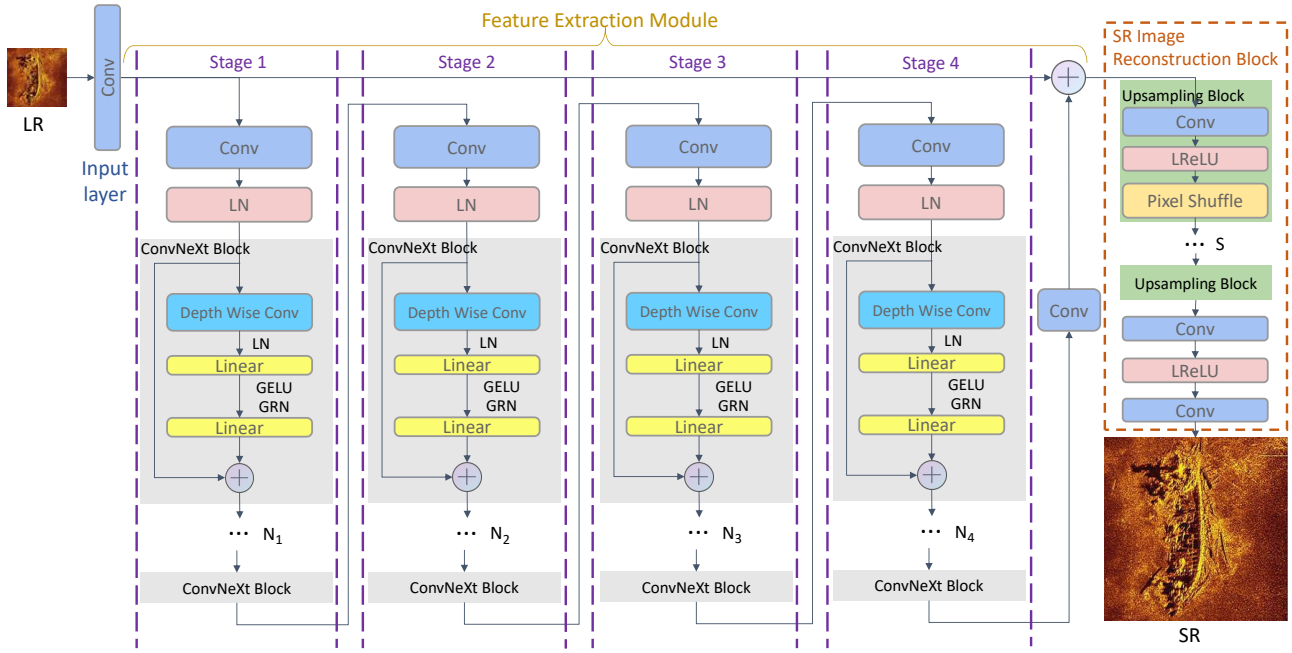

Fig. 1. Architecture of the Generator network of the proposed ConvGAN.

describes the structure of our improved discriminator and the loss function used in the model. The datasets and experimental comparisons are presented in Section IV. Conclusions are drawn in Section V.

II. Method

A. Generator Network Architecture

This section describes the main structure of the generator network we design. The generator has three main components: Input Layer, Feature Extraction Module, and SR Image Reconstruction Block. Given a low-resolution image as input, $I^{LR} \in \mathbb{R}^{h \times w \times c}$ (h, w, and c are the height, width, and channel number of I^{LR}, respectively). I^{SR} is the super-resolution image generated by the generator, $I^{SR} \in \mathbb{R}^{H \times W \times C}$($H$, W, and C are the height, width, and channel number of I^{SR}, respectively). $G(\cdot)$ is the generator function. The network of the generator is shown in Fig. 1. The overall generator is formulated as follows:

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

*1) **Input Layer and Feature Extraction Module**:* We use a convolutional layer with a kernel size 3×3, i.e. $Conv_3(\cdot)$, as the input layer to extract shallow features F_0. The input image space is mapped to a high-dimensional feature space by widening the channel number using the input convolutional layer.

$$F_0 = Conv_3(I^{LR}) \tag{2}$$

In our design, the feature extraction module contains four stages and a shortcut connection structure for extracting deep features, F_E. And each stage contains a 3×3 convolutional layer, a layer normalization, and N_i ConvNeXt Blocks (N_i denotes the number of ConvNeXt Blocks (CB) of the i-th Stage, $i = 1, 2, 3, 4$). We get intermediate features $F_1, \ldots, F_4$ from each stage. Since the feature extraction module in the super-resolution task needs to maintain a specific spatial size of the feature map, the downsampling structure leads to an inevitable loss of detailed features in the space of the feature map. Therefore, in our model design, we use a 3×3 convolutional layer instead of the downsampling structure and add them into each stage.

$$F_i = \mathcal{F}_i^{\mathrm{CB_n}}(\ldots(\mathcal{F}_i^{\mathrm{CB_1}}(LN(Conv_3(F_{i-1}))))) \tag{3}$$

where $\mathcal{F}_i^{\mathrm{CB_n}}(\cdot)$ denotes the n-th ConvNeXt Block function ($n = 1, \ldots, N_i$) of the i-th stage. $LN(\cdot)$ denotes layer normalization.

In the ConvNeXt Block, the input and output of the CB are connected by a shortcut connection. The input passes through an inverted bottleneck structure and first goes through a 7×7 depthwise convolution and layer normalization. Depthwise convolution dramatically reduces the number of operations and parameters by setting the group to the number of channels. Then, the 1×1 convolution is replaced by a fully connected layer for dimensionality enhancement, followed by the GELU activation function and Global Response Normalization (GRN) [21]. Finally, a fully connected layer is processed for dimensionality reduction. GRN improves the representation quality by enhancing feature diversity. The whole ConvNeXt Block process can be represented as:

$$F_{i,0} = LN(Conv_3(F_{i-1})) \tag{4}$$

$$F_{i,n} = \mathrm{CB}_n(F_{i,n-1}) = LGGL(LN(DW(F_{i,n-1}))) + F_{i,n-1} \tag{5}$$

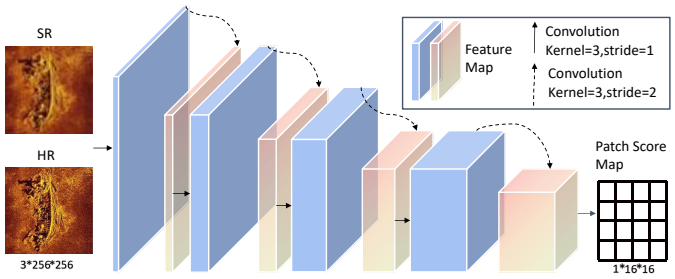

Fig. 2. Architecture of the Discriminator network of the proposed ConvGAN.

where $F_{i,n}$ denotes the output feature at the i-th Stage after the n-th ConvNeXt Block. $LGGL(\cdot)$ represents the operator of $Linear\_GELU\_GRN\_Linear$. DW is depthwise convolution. The final output F_4 of the 4-th stage passes through a 3×3 convolutional layer and connects the shallow features F_0 by a shortcut connection structure. After the above process, we finally obtain the feature map F_E after feature extraction

$$F_E = F_0 + Conv_3(F_4) \tag{6}$$

*2) **SR Image Reconstruction**:* We use the feature F_E after deep extraction as input to reconstruct the high-quality image I^{SR}. SR Image Reconstruction Block consists of two parts, S upsampling blocks, and an output layer. The upsampling block consists of the operator of $Conv\_LReLU\_PixelShuffle$ and PixelShuffle which obtains high-resolution feature maps by convolution and multi-channel recombination. An upsampling block achieves 2-fold up-sampling. The output layer consists of the operator of $Conv\_LReLU\_Conv$, which further outputs the generated SR image.

$$F_{u,0} = F_E \tag{7}$$

$$F_{u,s} = PixelShuffle(LReLU(Conv_3(F_{u,s-1}))) \tag{8}$$

where $F_{u,s}$ denotes the output feature map after the s-th upsampling block, $s = 1, \ldots, S$.

$$I^{SR} = Conv_3(LReLU(Conv_3(F_{u,S}))) \tag{9}$$

B. Discriminator Network Architecture

We choose the discriminator in VGG style. The structure of the discriminator is shown in Fig. 2. In the discriminator, the number of channels is increased from 64 to 512 by a series of convolutions of the feature map, and finally, the number of channels is reduced to 1 by a single-layer convolution. While widening the number of channels, we minimize the space size of the feature map. We output each image (HR or SR) through the discriminator as a patch map. Each value in the patch map corresponds to whether the input image is an HR image in a region of $3 \times 16 \times 16$ pixels.

C. Loss Function

*1) **Generator GAN Loss and Discriminator Loss**:* Specifically, during training, we use the method of Relativistic GAN [22]. This makes the generated image close to the real image. In training the generator, the SR image generated by the generator is fed into the discriminator at the same time as the HR image, and the GAN loss of the generator is defined as

$$\begin{aligned} L_{adv} = &- \mathbb{E}[\log(1 - M^P(I^{HR}) + \mathbb{E}[M^P(G(I^{LR}))])] \\ &- \mathbb{E}[\log(M^P(G(I^{LR})) - \mathbb{E}[M^P(I^{HR})])] \end{aligned} \tag{10}$$

where L_{adv} and L_D represent generator GAN loss and discriminator loss, respectively. $M^P(\cdot)$ represents the patch map of the image generated by the Discriminator. $\mathbb{E}(\cdot)$ represents the average over a mini-batch.

In training the discriminator, we also use both SR and HR images to gradually improve the ability of the discriminator to distinguish between SR and HR images. Loss of discriminator is defined as

$$\begin{aligned} L_D = &- \mathbb{E}[\log(M^P(I^{HR}) - \mathbb{E}[M^P(G(I^{LR}))])] \\ &- \mathbb{E}[\log(1 - M^P(G(I^{LR})) + \mathbb{E}[M^P(I^{HR})])]. \end{aligned} \tag{11}$$

*2) **Pixel Loss and Perceptual Loss**:* Pixel loss is used as a loss function to measure the image generation model to evaluate the consistency of the generated image with the HR image in detail. In the first phase, only pixel loss is used to guide the generator for training to develop a pre-trained model. As L1 loss has relatively good robustness, we choose L1 loss as pixel loss. We use Pixel loss defined as:

$$\begin{aligned} L_{pixel} = \frac{1}{W \times H \times C} \sum_{w=1}^{W} \sum_{h=1}^{H} \sum_{c=1}^{C} |G(I_j^{LR})(w,h,c) - \\ I_j^{HR}(w,h,c)| \end{aligned} \tag{12}$$

where H, W, and C are the height, width, and number of channels of the generated images. I_j denotes the j-th image, and G(·) still represents the function of generator.

Perceptual loss measures the similarity of generated and real images by comparing their differences in deep features. Perceptual loss can improve the visual quality of the generated images and make the generated ones more realistic. We use VGG19 to extract the deep features of SR and HR images. The extracted feature maps combine with the L1 loss for comparison. Perceptual loss is defined as

$$\begin{aligned} L_{perc} = \frac{1}{W \times H \times C} \sum_{w=1}^{W} \sum_{h=1}^{H} \sum_{c=1}^{C} |F^{VGG}(G(I_j^{LR}))(w,h,c) \\ - F^{VGG}(I_j^{HR})(w,h,c)| \end{aligned} \tag{13}$$

where H, W, and C are the height, width, and number of channels of the feature map of VGG19-54, which indicates the feature map of the 4-th convolution before activation and the 5-th max-pooling layer for the VGG19 network.

Finally, the total loss of the generator is:

$$L_G = \lambda L_{perc} + \mu L_{pixel} + \gamma L_{adv} \tag{14}$$

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

TABLE I
QUANTITATIVE COMPARISON FOR EACH SCALE ON THE KLSG DATASET

Method	Evaluation Metrics	Scale ×2	×4	×8
SRGAN	PSNR	30.1366	26.0125	24.6925
	SSIM	0.80642	0.49031	0.33705
	LPIPS	0.19551	0.38129	0.50751
ESRGAN	PSNR	30.2264	26.1122	24.7683
	SSIM	0.80539	0.50759	0.34318
	LPIPS	0.17558	0.34360	0.51048
RCAN	PSNR	30.5153	26.4709	24.8178
	SSIM	0.81143	0.52811	0.36806
	LPIPS	0.15297	0.30443	0.41083
SwinIR	PSNR	30.3836	27.8083	25.6881
	SSIM	0.80553	0.58715	0.43134
	LPIPS	0.15597	0.23533	0.34168
ConvGAN	PSNR	**30.8906**	**28.0835**	**26.4855**
	SSIM	**0.81992**	**0.60603**	**0.51956**
	LPIPS	**0.11218**	**0.21606**	**0.30628**

TABLE II
QUANTITATIVE COMPARISON FOR EACH SCALE ON THE SCTD DATASET

Method	Evaluation Metrics	Scale ×2	×4	×8
SRGAN	PSNR	27.3345	24.7418	22.9105
	SSIM	0.62586	0.42138	0.36446
	LPIPS	0.29933	0.47636	0.56968
ESRGAN	PSNR	27.4019	24.8559	23.1339
	SSIM	0.63948	0.44587	0.37628
	LPIPS	0.27146	0.44041	0.48922
RCAN	PSNR	27.3899	25.0977	23.2899
	SSIM	0.65259	0.52670	0.40363
	LPIPS	0.26265	0.40708	0.45609
SwinIR	PSNR	27.7461	25.1704	24.2996
	SSIM	0.67129	0.54429	0.49738
	LPIPS	0.24872	0.39067	0.43967
ConvGAN	PSNR	**28.3148**	**26.2788**	**25.0460**
	SSIM	**0.72896**	**0.66151**	**0.61745**
	LPIPS	**0.19620**	**0.24678**	**0.28289**

III. EXPERIMENTS

A. Datasets

We select two underwater sonar datasets for our experiments, the SeabedObjects-KLSG–II [23] dataset containing grayscale images and the SCTD [24] with color images. In order to improve the image quality of the datasets, we improve the sharpness of the images through a series of preprocessing operations. We split each dataset into training, validation, and test sets. Specifically, the training set contains 500 HR images. The validation and test sets contain 50 HR images, respectively. We use multi-scale random cropping and downsampling operations to set the images of the KLSG dataset and SCTD dataset to 128×128 and 256×256, respectively. HR images are bicubically downsampled to generate corresponding LR images. The KLSG dataset contains three classes: aircraft, seafloor, and shipwreck, and the STCD contains three classes: aircraft, shipwreck, and human.

B. Implementation Details

The experiments are conducted on the two datasets mentioned above. Super-resolution experiments are performed with scale factor $\times 2$, $\times 4$, and $\times 8$, i.e., the number of up-sampling blocks s in the generator is set to 1, 2, and 3, respectively. Different ConvNeXt blocks are set in each stage at a different scaling to complete the experiments. We implement the proposed ConvGAN method using the Pytorch framework and train it on two NVIDIA GeForce RTX 3090 GPUs. The training process is divided into two phases. In the first phase, only the generator is trained, and only pixel loss is used in the loss function to guide the generator to generate images that are close to HR as a whole, and the trained model of the generator is saved. In the second phase, the generator is based on the pre-trained model and the loss function involves pixel loss, GAN loss, and perceptual loss for training to generate the final model ConvGAN.

We also train four state-of-the-art methods, including SRGAN, ESRGAN, RCAN, and SwinIR, on the same datasets and compare them quantitatively and visually with our ConvGAN, respectively. During training, we use Adam [25] for optimization, where β_1 = 0.9 and β_2 = 0.999, and the learning rate is initialized to 0.0002. N_1 = 3, N_2 = 3, N_3 = 9, and N_4 = 3 are set in (3) and (5). Setting λ = 1, μ = 1, and γ = 1e-2 are set for training in (14). The batch size is set to 8.

In this paper, we have chosen three evaluation metrics that are widely used in the field of super-resolution images, including the peak signal-to-noise ratio (PSNR), the structural similarity index (SSIM), and learned perceptual image patch similarity (LPIPS). 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. Quantitative Comparison

Tables I and II present the quantitative comparison results of our proposed method and state-of-the-art methods at different scaling factors on the KLSG and SCTD datasets. The best values for three evaluation metrics in each scenario are highlighted in bold. Overall, all models perform significantly better on the KLSG dataset than the SCTD dataset, as the images in the SCTD dataset have more complex lighting, backgrounds, and richer details than the single-color images in the KLSG dataset.

Furthermore, the performance gap between models is smaller at lower scaling factors, indicating that our model performs well in generating fine details in challenging scenarios compared to other models. Additionally, the bold highlighted

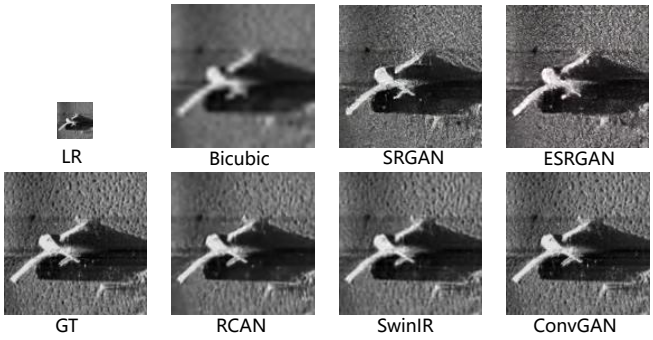

Fig. 3. SR visualization of ConvGAN with other compared methods on KLSG Dataset (Scale factor = 4).

values demonstrate that our model generally outperforms other advanced algorithms on all three metrics, proving the superiority of our model on the sonar dataset and its potential advantages in small datasets.

D. Visual Results

In Fig. 3, we choose the images from the KLSG dataset and compare the SR images generated by our method with those obtained by other advanced methods at scaling factors of r= 4. We use bicubic interpolation as the baseline. Since the overall performance on the KLSG dataset is satisfactory, we select the challenging aircraft image from this dataset, which has a complex background and visible detailed target, making it difficult to restore the particle details of the background and target. As a result, our ConvGAN method performs relatively better than other methods in background and target recovery.

In Fig. 4, we show a highly detailed image from the SCTD dataset and compare the SR images generated by our method with those produced by other state-of-the-art methods at a scaling factor of 4. First, regarding object restoration, ConvGAN outperforms other advanced methods in recovering more object details, especially for the shipwreck. Moreover, the restored background by SwinIR and RCAN tends to be blurry and lacks some texture, while the texture details recovered by SRGAN and ESRGAN often differ from those in the actual images. In contrast, the textures generated by the proposed ConvGAN are more authentic at a certain level.

In Fig. 5, we display two images of two classes from the SCTD dataset and compare the red square area of SR images generated by our method with those generated by other state-of-the-art methods when the scaling factor is 4. This detailed comparison shows that the bicubic method fails to produce details. SRGAN and ESRGAN recover realistic textures but produce HR images with some artifacts due to their poor utilization of features. RCAN and SwinIR can produce more texture details, but some parts of their results are still blurry. Our proposed ConvGAN achieves better visual quality. Moreover, the background restored by ConvGAN can involve clearer and sharper details than SRGAN and ESRGAN. The target (Shipwreck and Aircraft) generated by

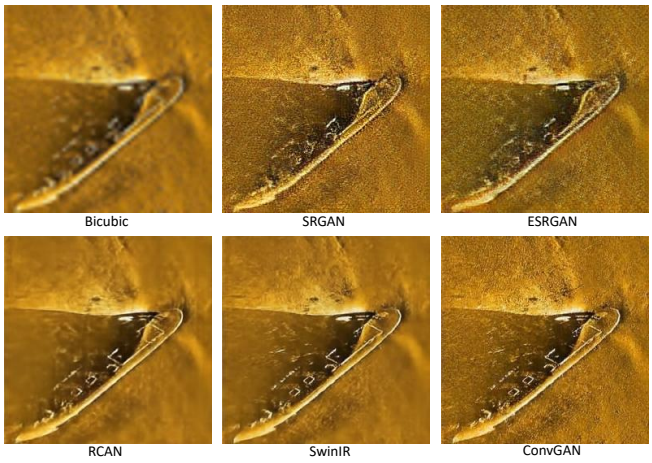

Fig. 4. SR visualization of ConvGAN and other compared methods on SCTD(Scale factor = 4).

ConvGAN is more similar to the ground-truth HR image than the other models.

IV. Conclusions

To achieve super-resolution reconstruction of underwater sonar images, we propose a super-resolution reconstruction method based on an adversarial generative network, named ConvGAN. We design a new four-stage generator structure and use the ConvNeXt module to extract deep features and improve the generalization ability of the generator. Moreover, we use the adversarial generative network to recover the details of underwater sonar images. We also improve the discriminator output as the score map of the patch so that the GAN loss guides the generator to generate the image details. In addition, we conduct experiments with ConvGAN on two different datasets in comparison with other state-of-the-art models at different scaling and difficulty levels. Quantitative and visual comparisons consistently demonstrate the effectiveness of our proposed ConvGAN method. In the future, we plan to extend our work to tasks such as image denoising and blind super-resolution for the processing of speckle noise in sonar images.

Acknowledgment

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, the Chinese Academy of Sciences under Grant YXJH202203.

References

[1] Xiufen Ye, Haibo Yang, Chuanlong Li, Yunpeng Jia, and Peng Li. A gray scale correction method for side-scan sonar images based on retinex. *Remote Sensing*, 11(11):1281, 2019.

[2] Huipu Xu, Lin Zhang, Meng Joo Er, and Qiushi 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)*, pages 44–48. IEEE, 2021.

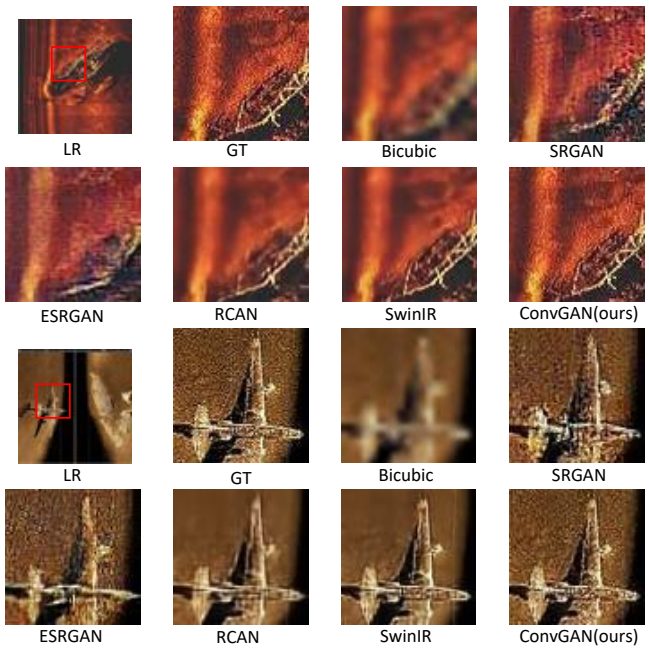

Fig. 5. Detail SR visualization of ConvGAN and other compared methods on SCTD (Scale factor = 4).

[3] Zhen Wang, Shanwen Zhang, Wenzhun Huang, Jianxin Guo, and Leya Zeng. Sonar image target detection based on adaptive global feature enhancement network. *IEEE Sensors Journal*, 22(2):1509–1530, 2021.

[4] Xiufen Ye, Weizheng Zhang, Yuanzi Li, and Wenyang Luo. Mobilenetv3-yolov4-sonar: Object detection model based on lightweight network for forward-looking sonar image. In *OCEANS 2021: San Diego–Porto*, pages 1–6. IEEE, 2021.

[5] Fan Yin, Chao Li, Haibin Wang, and Fan Yang. Automatic underwater acoustic target tracking by using image processing methods with jamming targets. In *2021 China Automation Congress (CAC)*, pages 4613–4616. IEEE, 2021.

[6] Pengfei Shao, Qing Li, and Bin Zhou. An improved tracking method for distributed passive sonar system. In *2022 7th International Conference on Signal and Image Processing (ICSIP)*, pages 100–105. IEEE, 2022.

[7] Trevi Jayanti Puspasari, Athur Yordan, and Iyod Suherman. Geoscience application for marine pipe line identification and validation. In *2021 IEEE Ocean Engineering Technology and Innovation Conference: Ocean Observation, Technology and Innovation in Support of Ocean Decade of Science (OETIC)*, pages 1–5. IEEE, 2021.

[8] William T Freeman and Egon C Pasztor. Learning low-level vision. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1182–1189. IEEE, 1999.

[9] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou 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*, pages 184–199. Springer, 2014.

[10] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654, 2016.

[11] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017.

[12] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan 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*, pages 1874–1883, 2016.

[13] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 286–301, 2018.

[14] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan 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*, pages 4681–4690, 2017.

[15] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018.

[16] Sen Jia, Zhihao Wang, Qingquan Li, Xiuping Jia, and Meng Xu. Multiattention generative adversarial network for remote sensing image super-resolution. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2022.

[17] Yadong Li, Sebastien Mavromatis, Feng Zhang, Zhenhong Du, Jean Sequeira, Zhongyi Wang, Xianwei Zhao, and Renyi 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*, 60:1–24, 2021.

[18] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1833–1844, 2021.

[19] Athira M Nambiar, Anurag Mittal, et al. A gan-based super resolution model for efficient image enhancement in underwater sonar images. In *OCEANS 2022-Chennai*, pages 1–8. IEEE, 2022.

[20] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022.

[21] Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. Convnext v2: Co-designing and scaling convnets with masked autoencoders. *arXiv preprint arXiv:2301.00808*, 2023.

[22] Alexia Jolicoeur-Martineau. The relativistic discriminator: a key element missing from standard gan. *arXiv preprint arXiv:1807.00734*, 2018.

[23] Guanying Huo, Ziyin Wu, and Jiabiao Li. Underwater object classification in sidescan sonar images using deep transfer learning and semisynthetic training data. *IEEE access*, 8:47407–47418, 2020.

[24] Peng Zhang, Jinsong Tang, Heping Zhong, Mingqiang Ning, Dandan Liu, and Ke Wu. Self-trained target detection of radar and sonar images using automatic deep learning. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2021.

[25] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.