2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) | 979-8-3503-0249-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/CVPRW59228.2023.00175

Lightweight Real-Time Image Super-Resolution Network for 4K Images

Ganzorig Gankhuyag*, Kihwan Yoon*, Jinman Park, Haeng Seon Son, Kyoungwon Min
Korea Electronics Technology Institute
South Korea
gnzrg25@gmail.com, rlghksdbs@gmail.com, jmpark@keti.re.kr, hsson@keti.re.kr, minkw@keti.re.kr

Abstract

Single-image super-resolution technology has become a topic of extensive research in various applications, aiming to enhance the quality and resolution of degraded images obtained from low-resolution sensors. However, most existing studies on single-image super-resolution have primarily focused on developing deep learning networks operating on high-performance graphics processing units. Therefore, this study proposes a lightweight real-time image super-resolution network for 4K images. Furthermore, we applied a reparameterization method to improve the network performance without incurring additional computational costs. The experimental results demonstrate that the proposed network achieves a PSNR of 30.15 dB and an inference time of 4.75 ms on an RTX 3090Ti device, as evaluated on the NTIRE 2023 Real-Time Super-Resolution validation scale X3 dataset. The code is available at https://github.com/Ganzooo/LRSRN.git.

1. Introduction

The single-image super-resolution (SISR) technology is a necessary image processing technique that aims to enhance the visual quality of low-resolution (LR) images by transforming them into high-resolution (HR) images with increased pixel density and more detailed information. The technology has a wide range of applications in computer vision, including remote sensing [35], underwater imaging [9], medical image analysis [31], mobile phones [21], and multimedia applications [17], autonomous vehicle [34] [8]. However, transforming LR images to HR images presents considerable challenges due to multiple potential HR images. The classic computer vision methods addressed this problem, such as interpolation-based [42] and representation-based [33] problems.

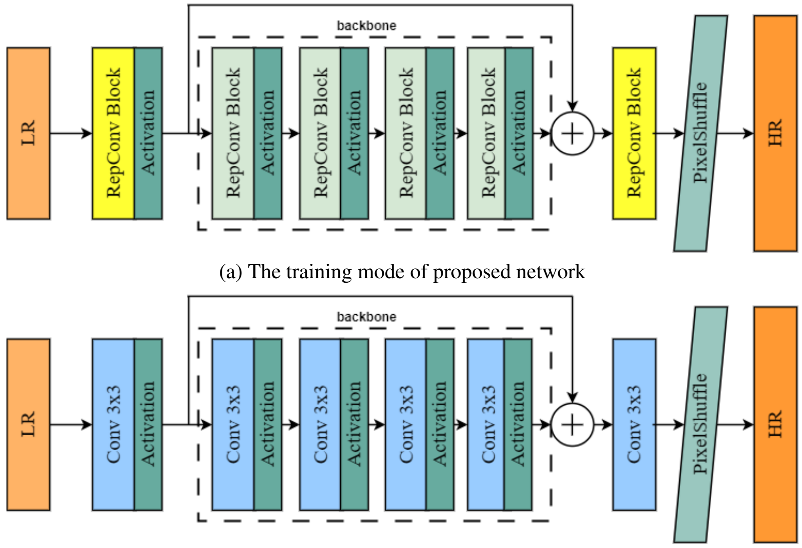
Significant improvements in developing deep learning methods have led to notable advances in super-resolution techniques in recent years. Deep learning-based single-image super-resolution methods studied to achieve state-of-the-art performance, such as the Super-Resolution Convolutional Neural Network (SRCNN) [12]. Various novel ideas and techniques were introduced, including different types of deep learning network architectures [13] [24], loss functions [7], training strategies and techniques [36].

However, numerous SISR methods have primarily focused on the reconstruction quality of HR images from LR images, demanding high-performance GPUs. Several techniques and networks were proposed to improve the reconstruction quality, developing SISR models with numerous parameters and high computational complexity [38]. However, the complex network structure, various types of deep learning techniques, and multiple parameters were challenges in deploying traditional SISR methods for 4K resolution images in real-time. Due to the limited computing resources of GPUs, a lightweight network and hardware-friendly deep learning techniques are required for the SISR model.

We proposed a lightweight real-time image super-resolution network (LRSRN) capable of reconstructing low-resolution (LR) images to high-resolution (HR) images with considerable accuracy and real-time inference speed, specifically for 4K images. The contribution of our proposed work was two-fold. Firstly, we introduced the LRSRN network structure that simultaneously achieves high accuracy and real-time speed, thereby overcoming the computational complexity and accuracy of traditional SISR methods. Secondly, we employed a reparameterized convolution (RepConv) layer, which enhances image quality while maintaining model size and inference speed.

The remainder of the paper is organized as follows. In Section 2, we discuss the related works of super-resolution. The proposed method is described in Section 3. The effectiveness of our LRSRN model was validated in Section 4. Finally, the conclusions of this study are summarized in Section 5.

*:Indicate equal contribution



(a) The training mode of proposed network

(b) The inference mode of proposed network

Figure 1. Network structure of a lightweight real-time image super-resolution network.

2. Related Work

2.1. Single Image Super-Resolution Methods

Single Image Super-Resolution (SISR) is a technology that generates high-resolution images from low-resolution images, and it is classified into traditional and deep learning-based methods. Traditional SISR methods include interpolation-based [42] and representation-based [39] methods. Interpolation-based methods, such as bilinear, bicubic, and nearest neighbor interpolation, estimate new pixel values by considering neighboring pixel values. These methods have relatively low complexity and computation. However, interpolation-based methods generate images uniformly without considering detailed image information, the results may appear unnatural, and the detailed information of the original image may be lost. Furthermore, implementing complex nonlinear models with these methods is limited due to their computational complexity. On the other hand, reconstruction-based methods consider detailed image information, resulting in more natural results than traditional interpolation-based methods. However, the computational cost of these methods increases due to the consideration of detailed image information.

Recently, with the advancement of deep learning, the result of deep learning-based SISR has significantly improved. SRCNN [12], the first deep learning-based SISR network proposed by Dong, improved the quality of images enlarged by the conventional bicubic interpolation using a network with only three fully connected layers. Afterward, a lightweight model, FSRCNN [13], was proposed to reduce the computation and number of parameters while improving the performance. Unlike the previous approach of improving the quality of images generated by bicubic interpolation, FSRCNN [13] added deconvolution layers within the network to generate SR images through training. However, existing networks have three major issues. First, they employ a limited number of convolution layers, which results in the utilize information from a small receptive field. Second, when a high learning rate is used, gradient vanishing and exploding issues occur. Third, existing networks can handle only a single scale factor per model. To address these issues, several networks have been proposed that use deep convolution layers to utilize information from the large receptive fields and employ residual learning to resolve problems such as gradient vanishing and exploding [22] [23]. In addition, a network that improves performance by removing unnecessary elements (e.g., Batch Normalization) from the network and increasing its depth is proposed [27].

Although the performance of deep learning-based SISR has significantly improved, the size of models and computational cost has also increased. Furthermore, there is a demand for real-time processing of SISR. Various studies are being conducted to achieve this goal. Some studies have proposed models for real-time SISR by reducing the computational complexity and model size of SISR networks [5, 18, 43], and also for SR on mobile devices [2, 3, 14, 16, 20, 28, 29]. Other methods such as transformer-based SISR have been researched [10] [26], showing significant performance improvements compared to other existing super-resolution techniques.

The Real-Time Image Super-Resolution task of NTIRE 2023 [11] aims to perform super-resolution of JPEG-compressed 4K images in real-time. Therefore, based on this objective, we propose a model, training methodology, and dataset designed to operate in real-time and enable the super-resolution of JPEG-compressed 4K images.

3. The proposed method

This section presents detailed information about the proposed method. Initially, we illustrate the architecture of LRSRN. Subsequently, we describe the parameterization (RepConv) block, an over-parameterized strategy to enhance the network's performance. Thus our base RepConv block is based on SCSRN [16], and we improve RepConv block efficiency. Finally, we explain our training strategies.

3.1. The architecture of LRSRN

The overall structure of the LRSRN is illustrated in Fig 1, which is an SCSRN-inspired structure comprising four components. The first component is a feature extraction layer that extracts the features from an LR image. The second component is a backbone comprising four RepConv blocks to learn deeper features. The third component includes a transition layer for the residual learning effect after directly adding the backbone's feature maps and input features. The final component (PixelShuffle) involves pixel re-arrangement for restoring the HR image.

For more clear understanding, let LR and HR denote the input and output of the network. We get the features extraction result of Fet_0 as follows:

$$Fet_0 = F_{fe}(LR), \tag{1}$$

where $F_{fe}(\cdot)$ denotes that features extraction from an LR image. Subsequently, we obtained the Fet_1 by

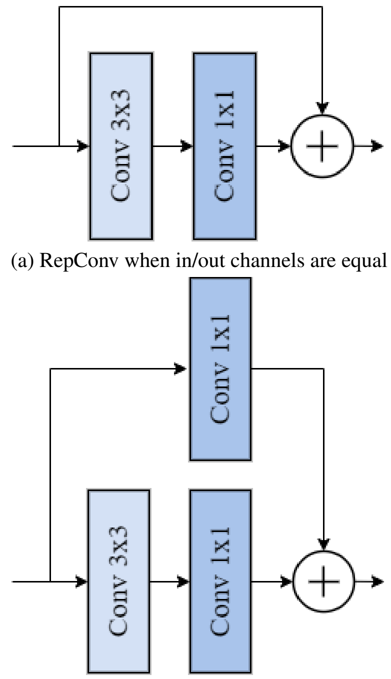
$$Fet_1 = F_{bb}(Fet_0), \tag{2}$$

where F_{bb} represents the function that contains high frequency and texture information extraction. Therefore, we add the Fet_0 and Fet_1, expecting a residual effect. After that, we pass the result through the transition layers for obtaining HR as follows:

$$HR = F_{ps}(F_{tr}(Fet_0 + Fet_1), \tag{3}$$

where F_{TR} denotes the transition layer. Using the pixel shuffle function F_{ps} to shuffle pixels to high-resolution images.

Unlike the SCSRN [16], the first feature of LR and the backbone feature were added. Also, we applied the RepConv block to all convolution layers. Moreover, the SCSRN uses the concatenation operation that helps to reduce the quantization error; however, this research did not need to concern with quantization error.



(a) RepConv when in/out channels are equal

(b) RepConv when in/out channels are not equal

Figure 2. RepConv Block of training mode.

3.2. The Reparameterization block

We applied the reparameterization (RepConv) method in the training stage to improve the reconstructed image quality. According to [6] and as depicted in Fig. 2, the RepConv can be reconstructed if it maintains the linearity property, even if the convolution layer overlaps in various manners. Thus, [6] [16] are applied RepConv only convolution layer's input and output channels are equal. However, we applied RepConv to each convolution layer, which is a more efficient application method, as illustrated in Fig. 1. When input and output channels are equal, we applied the original RepConv block in Fig. 2a. Input and output channels are not equal, we applied an advanced version of the RepConv block. This block used the Conv 1×1 block instead of the skip connection. Therefore, our RepConv block learns high-level information in the training step and can be simplified to the Conv 3×3 layer in the inference step.

3.3. Training strategy

We trained our model in two stages, including the scratch training stage and fine-tuning stage, with a different loss function.

Backbone	Channels	Patch Sizes	RepConv	Fine-Tune	NTIRE2023 val PSNR	NTIRE2023 val SSIM	Inference Time (ms)	Scores
5	64	96	X	X	31.897	0.9291	26.62	4.47
4	64	96	X	X	31.920	0.9296	22.19	4.98
4	32	96	X	X	31.909	0.9295	9.77	7.44
4	32	192	X	X	31.900	0.9294	9.77	7.40
4	32	192	O	X	32.784	0.9382	9.77	13.66
4	32	192	O	DIV2K [1]	32.812	0.9386	9.77	13.92
4	32	192	O	Proposed work	32.831	0.9388	9.77	14.11

Table 1. Ablation study results on DIV2K [1] val dataset.

3.3.1 Scratch train stage:

In the first stage, we train our model from scratch. The LR patches cropped from LR images with 192 × 192 size (128 x 128 for scale 3)and eight mini-batch sizes. The Adam optimizer uses a 0.0005 learning rate during scratch training. The cosine warm-up scheduler set a 0.1 percentage warm-up ratio. The total number of epochs is set to 800. We use $l1$ loss expressed in Eq. (4).

$$L^{l1}(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left|f(LR^i) - HR^i\right|, \tag{4}$$

where θ represents the trainable parameters of the proposed network, and n denotes the number of training patched images. LR^i and HR^i represent the patch image of LR and corresponding HR images. $f(\cdot)$ denotes the function of the proposed work.

3.3.2 Fine-tuning stage:

In the second stage, the model we initialize with the weights trained in the first stage. To improve the accuracy, we applied $l2$ loss as expressed in Eq. (5). Fine-tuning with $l2$ loss improves the peak signal-to-noise ratio (PSNR) value by 0.01 $\sim$ 0.02 dB. In this step, the initial learning rate is set as 0.0001, and the Adam optimizer is used. The cosine warm-up scheduler is set with a 0.1 percentage warm-up ratio. The total epoch is set to 200 epochs.

$$L^{l2}(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(f(LR^i) - HR^i\right)^2 \tag{5}$$

4. Experiment

This section describes our implementation details for the training of our proposed network. Then, we conducted an ablation study to compare the optimization performance of our proposed model. Finally, we compare the quantitative and qualitative results of our proposed model LRSRN with the recent state-of-the-art model.

4.1. Training Details

For training, we implemented Pytorch 1.13 version for all training steps. And we used DIV2K [1] dataset for the scratch training stage. A new SISR dataset proposed to validate and test the NTIRE2023 Real-Time Super-Resolution Challenge [11]. This dataset includes high-resolution, high-quality filtered content like generative models, digital art, video games, and photographs. Sample images illustrated in Fig. 3. To enhance performance on these datasets, we customized dataset includes the DIV2K [1] train set (800 images), Flicker2K [37] train set (2650 images), GTA [32] (train seq $00-19$ seq) sample 361, and LSDIR [25] (first 1000 images). And we used our customized datasets for the fine-tuning stage. The training process executed using NVIDIA RTX 3090Ti GPUs. It takes 16 hours for scratch training and 24 hours for fine-tuning, based on X2 SISR.

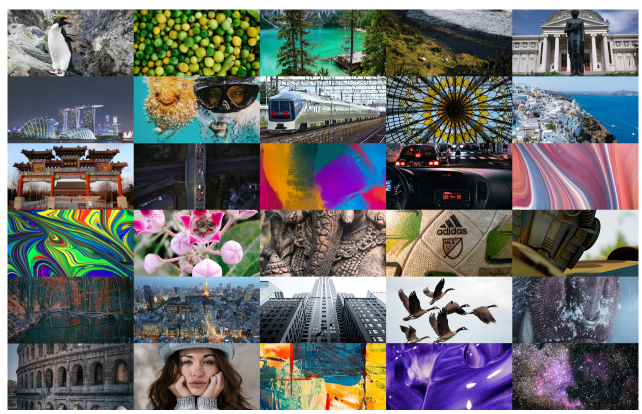


Figure 3. NTIRE2023 Real-Time Super-Resolution Challenge [11] validation and test dataset.

4.2. Ablation study

We compared the performance improvement of each contribution, such as model structure, reparameterization block, and training strategy. The ablation study was conducted for the X2 task. The PSNR, SSIM, and score changes for each contribution can be shown in Table 1. The score is calculated by NTIRE2023 evaluation script [15]. First, we adopted the RTSRN [40] model structure provided by NTIRE2023 as the baseline. To further improve inference time, we reduced the backbone of the model and decreased the channel size. As a result, PSNR and SSIM results are degraded, while we achieve a faster inference time.

Scale	Network	Set5			Set14			B100			Urban100			DIV2K		
		PSNR	SSIM	PSNR (Y)	PSNR	SSIM	PSNR (Y)	PSNR	SSIM	PSNR (Y)	PSNR	SSIM	PSNR (Y)	PSNR	SSIM	PSNR (Y)
X2	Bicubic	29.96	0.8676	31.86	27.38	0.8121	28.58	27.67	0.8169	28.05	24.98	0.8069	25.53	29.85	0.8662	30.79
	FSRCNN [13]	31.36	0.8892	33.41	28.38	0.8335	28.81	28.57	0.8379	29.01	26.44	0.8440	27.16	30.39	0.8849	31.95
	SESR [5]	31.71	0.8944	33.98	28.78	0.8423	30.29	28.94	0.8461	29.37	27.38	0.8641	28.20	31.40	0.8923	32.47
	IMDN [20]	32.24	0.9022	34.32	29.34	0.8525	30.66	29.26	0.8521	29.65	28.38	0.8821	29.18	31.98	0.9006	32.90
	RTSRN [40]	30.33	0.8713	32.31	27.72	0.8196	29.06	28.02	0.8171	28.44	25.41	0.8171	26.05	30.20	0.8719	31.23
	SCSRN [16]	31.72	0.8952	33.99	28.78	0.8426	30.29	28.92	0.8466	29.35	27.34	0.8640	28.15	31.41	0.8925	32.46
	Proposed work	31.84	0.8969	33.92	28.84	0.8436	30.26	28.93	0.8470	29.34	27.35	0.8641	28.11	31.44	0.8933	32.41
X3	Bicubic	27.28	0.7962	28.84	25.16	0.7190	26.11	25.43	0.7098	25.76	22.71	0.7040	23.13	27.42	0.7918	28.16
	FSRCNN [13]	28.41	0.8284	30.03	25.96	0.7436	27.06	26.03	0.7314	26.39	23.63	0.7430	24.12	28.17	0.8127	28.96
	SESR [5]	29.05	0.8429	30.84	26.39	0.7590	27.51	26.40	0.7432	26.73	24.37	0.7713	24.91	28.62	0.8240	29.42
	IMDN [20]	29.68	0.8559	31.32	26.86	0.7714	27.84	26.67	0.7514	26.96	25.13	0.7961	25.63	29.13	0.8357	29.79
	RTSRN [40]	27.55	0.7997	29.16	25.4	0.7263	26.43	25.62	0.7196	25.98	22.97	0.7137	23.44	27.63	0.7974	28.43
	SCSRN [16]	29.06	0.8431	30.84	26.41	0.7597	27.53	26.40	0.7449	26.74	24.41	0.7728	24.94	28.65	0.8253	29.44
	Proposed work	29.13	0.8459	30.75	26.47	0.7606	27.51	26.40	0.7443	26.72	24.40	0.7713	24.88	28.69	0.8255	29.41

Table 2. Quantitative results comparison on benchmark datasets.(Red indicates best PSNR/SSIM values within each dataset and Blue indicates second best.)

Scale	Models	Params (K)	Inference Time (ms)	NTIRE2023 val (PSNR db) [11]	Scores
X2	FSRCNN [13]	25.35	33.05	29.60	5.65
	SESR [5]	23.64	12.52	29.95	10.99
	IMDN [20]	873.20	143.34	30.44	4.75
	RTSRN [40]	193.40	23.37	29.14	4.77
	SCSRN [16]	46.80	10.24	29.95	12.15
	Proposed work	41.40	9.77	30.15	14.11
X3	FSRCNN [13]	25.35	14.68	29.60	7.91
	SESR [5]	23.64	6.11	29.95	15.63
	IMDN [20]	881.88	63.97	30.44	6.78
	RTSRN [40]	202.00	10.67	29.14	6.74
	SCSRN [16]	53.01	5.05	29.95	17.19
	Proposed work	45.70	4.75	30.15	20.36

Table 3. Development phase results of NTIRE2023 Real-Time Super-Resolution Track1 (X2) and Track2 (X3) with benchmark.

Thus, the final score also showed an improvement of 2.97 compared to the baseline model.

We also used reparameterization blocks and fine-tuning to improve PSNR and SSIM without increasing inference time. When using the reparameterization block, the inference time remained the same, while PSNR and SSIM were improved by 0.875 dB and 0.0087, respectively. As a result, the final score increased by 6.22. In addition, when we trained the fine-tuning stage of Section 3 using only DIV2K [1], the PSNR and SSIM improved by 0.028 dB and 0.0004, respectively, and the final score increased by 0.28. Finally, to improve the performance of the NTIRE2023 validation dataset, which is composed of various datasets, we used a customized dataset for fine-tuning. We improved PSNR by 0.19 dB, SSIM by 0.0002, and the final score by 0.19.

4.3. Comparison with the State-of-the-Arts

To compare our proposed network, we selected several state-of-the-art networks, including traditional network FSRCNN [13], SESR [5], IMDN [20], the baseline model RTSRN [40] provided by NTIRE2023 real-time super-resolution challenge, and the best network in Mobile AI & AIM 2022 Real-Time Single-Image Super-Resolution Challenge SCSRN [16]. To ensure a fair comparison, we trained the existing networks using the JPEG-compressed DIV2K [1] dataset. To compare the results of the trained networks, we used six benchmark datasets: Set5 [4], Set14 [41], B100 [30], Urban100 [19], and DIV2K [1] dataset, and the validation dataset provided in NTIRE 2023 [11] [40].

The primary goal of the challenge is to perform real-time SISR on JPEG-compressed datasets. Therefore, we compressed the benchmark images using JPEG format with a compression ratio of $q = 90$, and we used the compressed datasets for comparison. And we compared PSNR and SSIM in the RGB domain and measured the inference time of each model. Lastly, we reached the final scores used in NTIRE2023. Eq. 6 is the score function used for the X2 and X3 tasks.

$$scores = \frac{2^{PSNR-thr}}{0.1 \times \sqrt{InferenceTime}} \tag{6}$$

, where thr is equal to 31.69 at X2 tasks and 29.00 at X3 tasks. The inference time is measured at a resolution of 3840x2160 (4K).

Table 2 shows the PSNR, SSIM and PSNR (Y) results for the LRSRN and SOTA network. Our proposed network

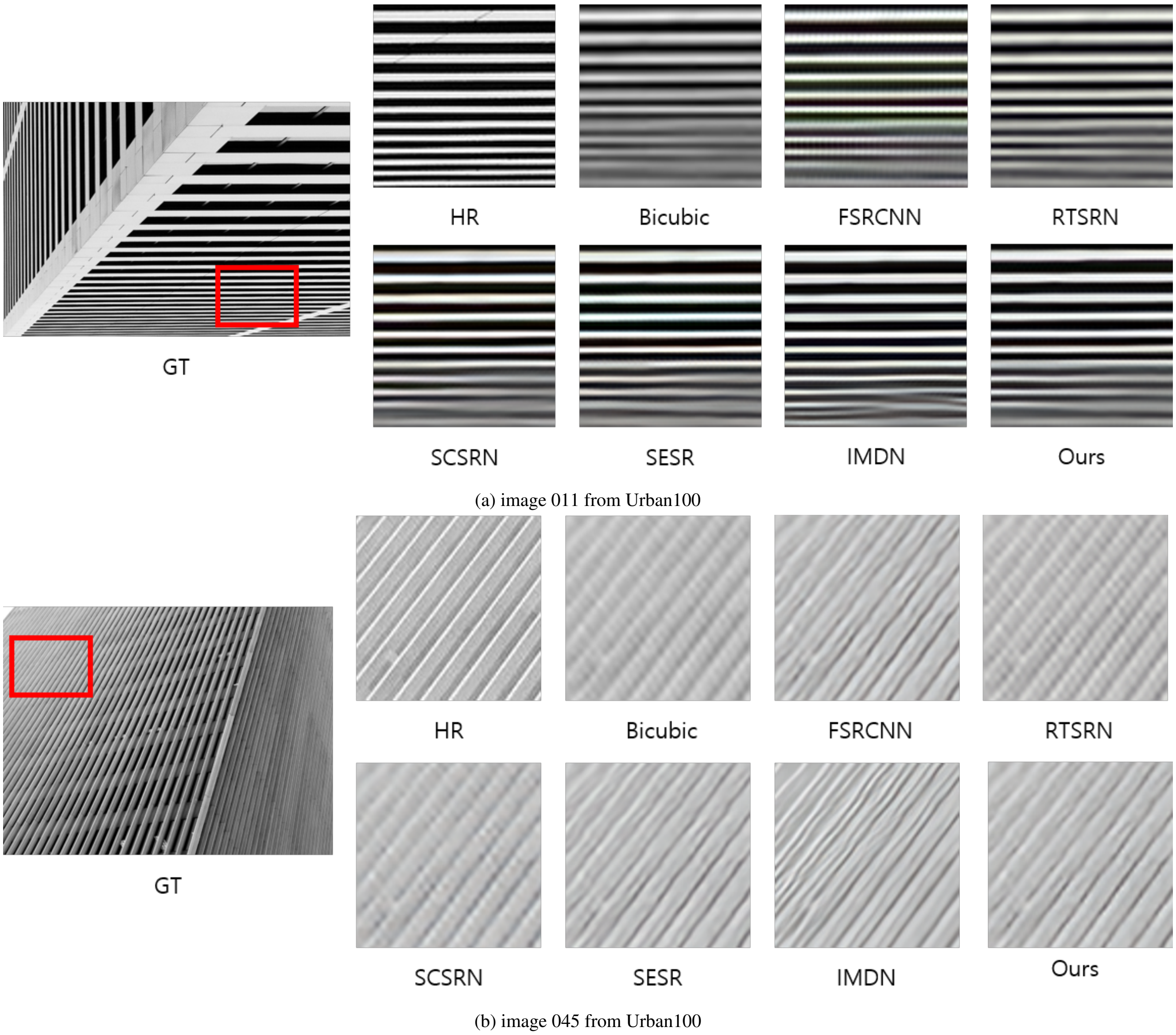


Figure 4. Qualitative comparison of proposed work with previous networks on scale X3.

shows the second-best performance regarding PSNR and SSIM for the X2 task. Furthermore, it also indicates the second-best performance in most benchmarks for the X3 task. However, in Table 3, which indicates the number of parameters, inference time, and scores of each model, it can be seen that although IMDN [20] shows the best PSNR result, it included a large number of parameters and is inefficient in terms of inference time. On the other hand, our proposed network shows approximately 16 times faster in the X2 task and approximately 15 times faster in the X3 task compared to IMDN [20], although its PSNR is lower. Therefore, considering the overall performance in inference time and PSNR, our proposed network exhibits the highest score among the compared methods.

We compared the subjective image quality of the SOTA network and our proposed network. Fig. 4 present the qualitative results of the X3 SISR. The results for Urban100 demonstrate that the existing networks generate relative blurred or distorted images. On the other hand, we can see that our proposed network generates relatively sharp images. Moreover, we also present the qualitative results of the X2 SISR in Fig. 5. In these results, we can see that our proposed network successfully reconstructs the edges and textures.

Finally, Table 4 compares our network and the top-ranked teams participating in the NTRE2023 Real-Time

(a) image 886 from DIV2K

(b) image 896 from DIV2K

Figure 5. Qualitative comparison of proposed work with previous networks on scale X2.

Super-Resolution Challenge. The competition provided PSNR, SSIM, PSNR (Y), and inference time. Moreover, inference time was measured using RTX 3060 or RTX 3090. Although the inference time of the proposed network is slower than other teams, it works in real-time and achieves the best PSNR performance in X2 SISR track and the second best in X3 SISR track.

5. Conclusions

The proposed work proposes a lightweight approach to address the problem of real-time image super-resolution for 4K images using a lightweight neural network architecture. The proposed network employs a reparameterization method that enhances the quality of super-resolved images without affecting the inference time performance. The experimental results demonstrate that the proposed network achieves a PSNR of 30.15 dB and an inference time of 4.75 ms on an RTX 3090Ti device, as evaluated on the NTIRE 2023 Real-Time Super-Resolution validation scale X3 dataset.

Acknowledgment

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2021-0-00800, Development of Driving Environment Data Trans-

Scale	Network	NTIRE2023 test [11]			
		PSNR	SSIM	PSNR (Y)	Inference Time
X2	Bicubuc	33.92	0.8829	36.66	0.45
	Noah_TerminalVision	35.02	0.8957	37.74	3.19
	ALONG	34.68	0.8906	37.38	1.91
	RTVSR	34.71	0.8910	37.50	2.24
	Team OV	34.62	0.8899	37.45	2.91
	Proposed work	35.02	0.8948	37.76	11.19
X3	Bicubuc	31.30	0.8246	33.82	0.5
	Aselsan Research	32.06	0.8344	34.56	1.17
	Team OV	32.17	0.8376	34.72	1.51
	ALONG	32.18	0.8367	34.66	1.66
	RTVSR	32.22	0.8372	34.77	1.96
	Proposed work	32.59	0.8446	35.05	5.47

Table 4. Test phase results of NTIRE2023 Real-Time Super-Resolution Track 1 and Track 2

formation and Data Verification Technology for the Mutual Utilization of Self-driving Learning Data for Different Vehicles)

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 126–135, 2017. 4, 5

[2] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *Proceedings of the European conference on computer vision (ECCV)*, pages 252–268, 2018. 2

[3] Mustafa Ayazoglu. Extremely lightweight quantization robust real-time single-image super resolution for mobile devices. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop*, pages 2472–2479, 2021. 2

[4] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012. 5

[5] Kartikeya Bhardwaj, Milos Milosavljevic, Liam O'Neil, Dibakar Gope, Ramon Matas, Alex Chalfin, Naveen Suda, Lingchuan Meng, and Danny Loh. Collapsible linear blocks for super-efficient super resolution. *Proceedings of Machine Learning and Systems*, 4:529–547, 2022. 2, 5

[6] Kartikeya Bhardwaj, Milos Milosavljevic, Liam O'Neil, Dibakar Gope, Ramon Matas, Alex Chalfin, Naveen Suda, Lingchuan Meng, and Danny Loh. Collapsible linear blocks for super-efficient super resolution. *Proceedings of Machine Learning and Systems*, 4:529–547, 2022. 3

[7] Adrian Bulat and Georgios Tzimiropoulos. Super-fan: Integrated facial landmark localization and super-resolution of real-world low resolution faces in arbitrary poses with gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 109–117, 2018. 1

[8] Mun Byoung-Il, Ganzorig Gankhuyag, Son Haeng-Seon, Sung Gi-Ho, and Park Jin-Man. Point-cloud perspective transform via super-resolution model. In *2022 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia)*, pages 1–4, 2022. 1

[9] Yuzhang Chen, Kangli Niu, Zhangfan Zeng, and Yongcai Pan. A wavelet based deep learning method for underwater image super resolution reconstruction. *IEEE Access*, 8:117759–117769, 2020. 1

[10] Marcos V Conde, Ui-Jin Choi, Maxime Burchi, and Radu Timofte. Swin2sr: Swinv2 transformer for compressed image super-resolution and restoration. In *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pages 669–687. Springer, 2023. 2

[11] Marcos V Conde, Eduard Zamfir, Radu Timofte, et al. Efficient deep models for real-time 4k image super-resolution. ntire 2023 benchmark and report. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 2, 4, 5, 8

[12] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2015. 1, 2

[13] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *European conference on computer vision*, pages 391–407. Springer, 2016. 1, 2, 5

[14] Zongcai Du, Jie Liu, Jie Tang, and Gangshan Wu. Anchor-based plain net for mobile image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2494–2502, 2021. 2

[15] NTIRE2023 evaluation script. https://github.com/eduardzamfir/NTIRE23-RTSR. 4

[16] Ganzorig Gankhuyag, Jingang Huh, Myeongkyun Kim, Kihwan Yoon, HyeonCheol Moon, Seungho Lee, Jinwoo Jeong, Sungjei Kim, and Yoonsik Choe. Skip-concatenated image super-resolution network for mobile devices. *IEEE Access*, 2022. 2, 3, 5

[17] Khizar Hayat. Multimedia super-resolution via deep learning: A survey. *Digital Signal Processing*, 81:198–217, 2018. 1

[18] Zejiang Hou and Sun-Yuan Kung. Efficient image super resolution via channel discriminative deep neural network pruning. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3647–3651. IEEE, 2020. 2

[19] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5197–5206, 2015. 5

[20] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *Proceedings of the 27th acm international conference on multimedia*, pages 2024–2032, 2019. 2, 5, 6

[21] Andrey Ignatov, Radu Timofte, Maurizio Denna, Abdel Younes, Ganzorig Gankhuyag, Jingang Huh, Myeong Kyun Kim, Kihwan Yoon, Hyeon-Cheol Moon, Seungho Lee, Yoonsik Choe, Jinwoo Jeong, Sungjei Kim, Maciej Smyl,

Tomasz Latkowski, Pawel Kubik, Michal Sokolski, Yujie Ma, Jiahao Chao, Zhou Zhou, Hongfan Gao, Zhengfeng Yang, Zhenbing Zeng, Zhengyang Zhuge, Chenghua Li, Dan Zhu, Mengdi Sun, Ran Duan, Yan Gao, Lingshun Kong, Long Sun, Xiang Li, Xingdong Zhang, Jiawei Zhang, Yaqi Wu, Jinshan Pan, Gaocheng Yu, Jin Zhang, Feng Zhang, Zhe Ma, Hongbin Wang, Hojin Cho, Steve Kim, Huaen Li, Yanbo Ma, Ziwei Luo, Youwei Li, Lei Yu, Zhihong Wen, Qi Wu, Haoqiang Fan, Shuaicheng Liu, Lize Zhang, Zhikai Zong, Jeremy Kwon, Junxi Zhang, Mengyuan Li, Nianxiang Fu, Guanchen Ding, Han Zhu, Zhenzhong Chen, Gen Li, Yuanfan Zhang, Lei Sun, Dafeng Zhang, Neo Yang, Fitz Liu, Jerry Zhao, Mustafa Ayazoglu, Bahri Batuhan Bilecen, Shota Hirose, Kasidis Arunruangsirilert, Luo Ao, Ho Chun Leung, Andrew Wei, Jie Liu, Qiang Liu, Dahai Yu, Ao Li, Lei Luo, Ce Zhu, Seongmin Hong, Dongwon Park, Joonhee Lee, Byeong Hyun Lee, Seunggyu Lee, Se Young Chun, Ruiyuan He, Xuhao Jiang, Haihang Ruan, Xinjian Zhang, Jing Liu, Garas Gendy, Nabil Sabor, Jingchao Hou, and Guanghui He. Efficient and accurate quantized image super-resolution on mobile npus, mobile ai & aim 2022 challenge: Report. In Leonid Karlinsky, Tomer Michaeli, and Ko Nishino, editors, *Computer Vision – ECCV 2022 Workshops*, pages 92–129, Cham, 2023. Springer Nature Switzerland. 1

[22] 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. 2

[23] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1637–1645, 2016. 2

[24] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 624–632, 2017. 1

[25] Yawei Li, Kai Zhang, Jingyun Liang, Jiezhang Cao, Ce Liu, Rui Gong, Yulun Zhang, Hao Tang, Yun Liu, Denis Demandolx, Rakesh Ranjan, Radu Timofte, and Luc Van Gool. Lsdir: A large scale dataset for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 4

[26] 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. 2

[27] 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. 2

[28] Jie Liu, Jie Tang, and Gangshan Wu. Residual feature distillation network for lightweight image super-resolution. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 41–55. Springer, 2020. 2

[29] Ziwei Luo, Youwei Li, Lei Yu, Qi Wu, Zhihong Wen, Haoqiang Fan, and Shuaicheng Liu. Fast nearest convolution for real-time efficient image super-resolution, 2022. 2

[30] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423. IEEE, 2001. 5

[31] Defu Qiu, Lixin Zheng, Jianqing Zhu, and Detian Huang. Multiple improved residual networks for medical image super-resolution. *Future Generation Computer Systems*, 116:200–208, 2021. 1

[32] Stephan R. Richter, Zeeshan Hayder, and Vladlen Koltun. Playing for benchmarks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2232–2241, 2017. 4

[33] Zhenfeng Shao, Lei Wang, Zhongyuan Wang, and Juan Deng. Remote sensing image super-resolution using sparse representation and coupled sparse autoencoder. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(8):2663–2674, 2019. 1

[34] Liangtian Wan, Yuchen Sun, Lu Sun, Zhaolong Ning, and Joel JPC Rodrigues. Deep learning based autonomous vehicle super resolution doa estimation for safety driving. *IEEE Transactions on Intelligent Transportation Systems*, 22(7):4301–4315, 2020. 1

[35] Peijuan Wang and Elif Sertel. Channel–spatial attention-based pan-sharpening of very high-resolution satellite images. *Knowledge-Based Systems*, 229:107324, 2021. 1

[36] Yifan Wang, Federico Perazzi, Brian McWilliams, Alexander Sorkine-Hornung, Olga Sorkine-Hornung, and Christopher Schroers. A fully progressive approach to single-image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 864–873, 2018. 1

[37] Yingqian Wang, Longguang Wang, Jungang Yang, Wei An, and Yulan Guo. Flickr1024: A large-scale dataset for stereo image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 4

[38] Zhihao Wang, Jian Chen, and Steven CH Hoi. Deep learning for image super-resolution: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3365–3387, 2020. 1

[39] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE transactions on image processing*, 19(11):2861–2873, 2010. 2

[40] Eduard Zamfir, Marcos V Conde, and Radu Timofte. Towards real-time 4k image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 4, 5

[41] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *Curves and Surfaces: 7th International Conference, Avignon, France, June 24-30, 2010, Revised Selected Papers 7*, pages 711–730. Springer, 2012. 5

[42] Lei Zhang and Xiaolin Wu. An edge-guided image interpolation algorithm via directional filtering and data fusion. *IEEE transactions on Image Processing*, 15(8):2226–2238, 2006. 1, 2

[43] Yulun Zhang, Huan Wang, Can Qin, and Yun Fu. Aligned structured sparsity learning for efficient image super-resolution. *Advances in Neural Information Processing Systems*, 34:2695–2706, 2021. 2