

DIPNet: Efficiency Distillation and Iterative Pruning for Image Super-Resolution

Lei Yu^{1*} Xinpeng Li^{1*} Youwei Li² Ting Jiang¹ Qi Wu¹ Haoqiang Fan¹ Shuaicheng Liu^{3,1†}

¹Megvii Technology ²Microbt

³University of Electronic Science and Technology of China

{yulei02, lixinpeng}@megvii.com, liyouwei.wellee@gmail.com,
{jianting, wuqi02, fhq}@megvii.com, liushuaicheng@uestc.edu.cn

Abstract

Efficient deep learning-based approaches have achieved remarkable performance in single image super-resolution. However, recent studies on efficient super-resolution have mainly focused on reducing the number of parameters and floating-point operations through various network designs. Although these methods can decrease the number of parameters and floating-point operations, they may not necessarily reduce actual running time. To address this issue, we propose a novel multi-stage lightweight network boosting method, which can enable lightweight networks to achieve outstanding performance. Specifically, we leverage enhanced high-resolution output as additional supervision to improve the learning ability of lightweight student networks. Upon convergence of the student network, we further simplify our network structure to a more lightweight level using reparameterization techniques and iterative network pruning. Meanwhile, we adopt an effective lightweight network training strategy that combines multi-anchor distillation and progressive learning, enabling the lightweight network to achieve outstanding performance. Ultimately, our proposed method achieves the fastest inference time among all participants in the NTIRE 2023 efficient super-resolution challenge while maintaining competitive super-resolution performance. Additionally, extensive experiments are conducted to demonstrate the effectiveness of the proposed components. The results show that our approach achieves comparable performance in representative dataset DIV2K, both qualitatively and quantitatively, with faster inference and fewer number of network parameters.

1. Introduction

Single Image Super-Resolution (SISR) aims to reconstruct a high-resolution (HR) image from a low-

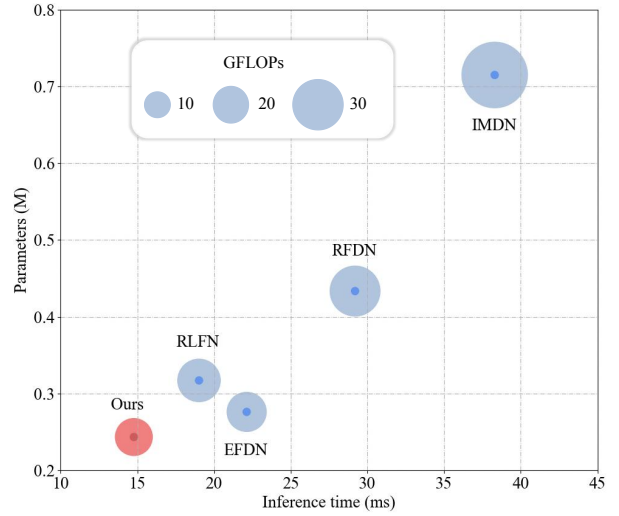


Figure 1. Comparison with recent efficient SR methods. The figure shows the GFLOPs of these methods when the input is 256×256 , the number of parameters of these models and their average inference time using NVIDIA 2080ti under the DIV2K validation set.

resolution (LR) input and has become an essential task in low-level computer vision for enhancing image resolution [1, 2]. Recent SISR approaches [1, 3–10] based on deep learning have achieved great success by significantly improving the quality of reconstructed images. However, these methods frequently require large amounts of computational resources, making it challenging to deploy them on resource-constrained devices for real-world applications.

To address this issue, there is a growing need to develop efficient SISR models with higher inference speed while maintaining good trade-offs between image quality and computation cost. Prior research has attempted to reduce model parameters or floating-point operations (FLOPs) to improve efficiency. Recursive networks with weight-sharing strategies are often used to decrease the number of parameters, but they may not necessarily reduce

*Equal contribution

†Corresponding author

the number of operations and inference time due to their complex graph topology. Similarly, commonly used techniques to reduce FLOPs, such as depth-wise convolutions, feature splitting, and shuffling [4, 8, 11, 12], may not always improve computational efficiency.

Therefore, we consider the problem of achieving efficient super-resolution from another perspective, that is, how to obtain an efficient super-resolution model only through better training strategies without too much additional network design. As shown in Fig. 1, the strategies we proposed make the SR model faster and smaller. It is difficult to train a small network directly, and it is easier to train a large teacher network first and then guide the small student network to learn through knowledge distillation. However, due to the large gap between the learning ability of the teacher network and the student network, it is difficult for the student network to learn enough high-frequency information if it is directly distilled. Inspired by HGGT [13], the HR images we used are not the original HR images in the data set, but the enhanced HR images. The enhanced images can provide richer high-frequency information, which can help student network with limited learning ability to learn more easily. Inspired by repVGG [14], we designed the network structure including series branches, parallel branches and residuals in the process of designing the student network. These additional branches can increase the learning ability of the student network. When the student network training converges, it can be reparameterized and simplified into a lightweight structure. This operation enhances the learning ability of the model without introducing additional model complexity. In order to make full use of the guiding ability of the teacher network, we use a multi-level distillation strategy, that is, set anchor points at different nodes of the network, and use the features of different levels at the anchor points to perform distillation.

Usually, for the convenience of training in image tasks, we will use relatively small patches for training. However, recent studies [15] pointed out that using this method will cause the input distribution to be different during training and testing so that it cannot be effective enough when the network performs some global operations during testing. Therefore, in order to make the model perform better in the testing phase, the input size during training and testing should be as close as possible. However, directly using large-sized patches for training will make the training very time-consuming, and the inability to use larger batches will further affect the stability of training. At the same time, it is not conducive to using large patches as input for networks with limited expressive capabilities to mine global information. So we adopted a progressive learning method, that is, gradually increasing the input patch during training, and achieved good results. The trained student network still has some unimportant redundant parameters, so we further it-

eratively pruned the model to further compress the model size.

Our contributions can be summarized as follows:

- For the first time, we propose the use of enhanced HR images to improve the learning ability of lightweight networks.
- We propose a novel multi-stage lightweight training strategy combining distillation, progressive learning, and pruning.
- We conducted extensive experiments to demonstrate the effectiveness of our method, and our method outperformed all other competitors in the NTIRE 2023 efficient super-resolution challenge in terms of time consumption and model size.

2. Related Work

2.1. Single Image Super-Resolution

In the past few years, deep neural networks (DNN) have shown remarkable capability on improving SISR performance. The pioneering work is SRCNN [16] which applies the bicubic downsampling on HR images to construct data pairs and employs a simple convolution neural networks (CNN) to learn the end-to-end mapping from LR to HR images. Then plenty of CNN-based methods have been proposed to achieve better performance [6, 7, 17–24]. For example, Kim *et al.* [17] proposed a 20-layer network with residual learning, which inspired the development of deeper and wider networks for SISR. EDSR [18] followed the idea of residual learning and modified residual blocks by removing the batch normalization layer to build a very deep and wide network. RCAN [22] have introduced channel attention and second-order channel attention respectively, which exploit feature correlations for improved performance. Moreover, recent works [25–29] have been proposed to improve the perceptual visual quality of real-world images. Meanwhile, some studies based on blind image super-resolution were proposed [30–32], addressing the problem of degenerate kernels present in real-world super-resolution. In addition, some works employ some advanced losses such as the VGG loss [16], perceptual loss [33], and GAN loss [34] to learn realistic image details. Recently, transformer-based super-resolution methods [35, 36] have gained popularity, which achieve high performance. However, most of these methods require a large amount of computational resources and have a high number of parameters, FLOPs and inference time, and do not facilitate practical deployment and application in edge devices.

2.2. Efficient Image Super-Resolution

Efficient Image Super-Resolution aims to reduce the computational effort and the number of parameters of the

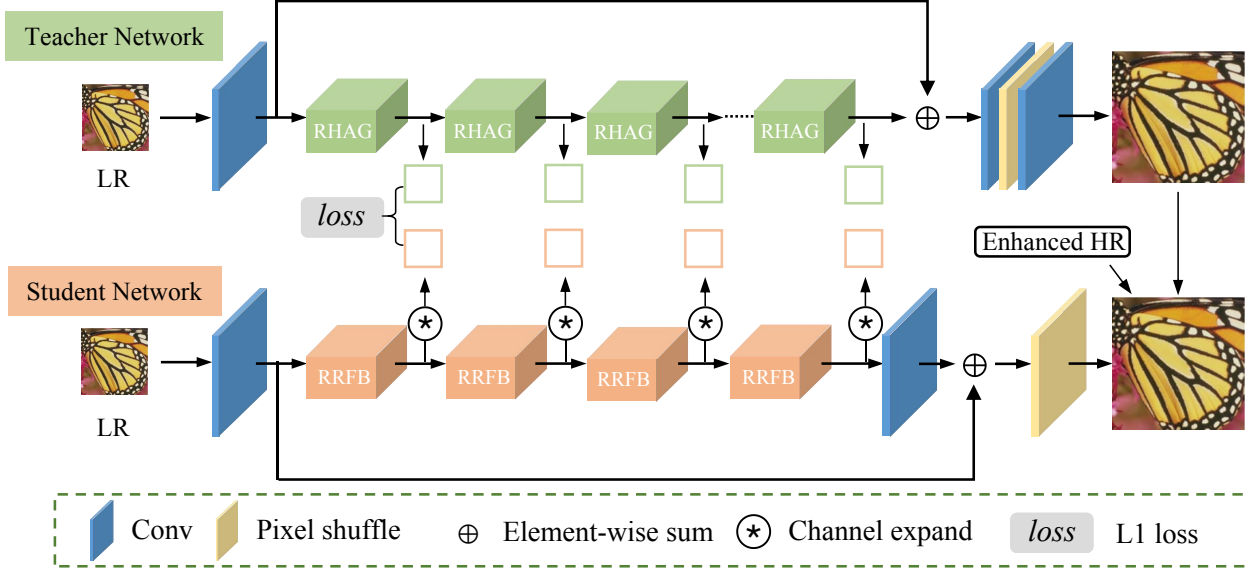


Figure 2. The multi-stage feature distillation pipeline of our method.

SR network while achieving faster inference times and maintaining high performance. In real-world SR model deployments, the computing power of the deployed devices is often limited, such as edge devices, etc. In this case, the efficiency of the SR network becomes an important aspect. To fit the increasing demands for deploying SR models with limited computing resources, numerous works have refocused their attention to efficient image SR techniques [4, 8, 11, 37–39]. At the same time, a number of competitions, just like NTIRE and AIM, have launched efficient image SR entries to promote the development of relevant research [40, 41]. In recent related studies, CARN [11] presented local and global cascading mechanisms to achieve a lightweight SR network. IMDN [4] designs an information multi-distillation network by constructing the cascaded information multi-distillation blocks to extract hierarchical features. The following work RFDN [8] further improves the network by introducing feature distillation blocks that employ 1×1 convolution layers to implement dimensional change. Based on RFDN, RLFN [37] investigates its speed bottleneck and enhances its speed by removing the hierarchical distillation connections. Furthermore, RLFN proposes a feature extractor to extract more information of edges and textures. With these advancements, they achieved first place in the NTIRE 2022 Efficient Super-Resolution Challenge [40].

3. Method

We propose an efficiency distillation and iterative pruning SR network named DIPNet which consists of four main components. In Sec. 3.1, we revisit the RLFB and propose a

reparameterization residual feature block (RRFB), and our network structure is mainly constructed by stacking multiple RRFBs, as shown in Fig. 2. In Sec. 3.2, we introduce the method of model-guided ground-truth enhancement strategy to improve the quality of original HR. Then we discuss the multi-anchor feature distillation in Sec. 3.3, which can effectively improve the performance of the network. Finally, we propose an iterative pruning strategy in Sec. 3.4 to further reduce the number of model parameters.

3.1. Reparameterization Residual Feature Block

Following RFDN [8] and RLFN [37], we also use an information distillation network to reconstruct high-quality SR images. Based on the block RLFB of RLFN, we introduce the re-parameterizable topology to the block. The original block of RLFB in RLFN is shown in Fig. 3(a), we expand the RLFB in RLFN to the structure reparameterization residual feature block (RRFB) shown in Fig. 3(b) in the training phase. The structure of RRB which is shown in Fig. 3(c) excavates the potential ability of complex structure during optimization, while maintaining computational efficiency, as it is computationally equivalent to a single 3×3 convolution during inference.

3.2. Model Guided Ground-truth Enhancement

According to our understanding, almost all existing SR methods directly use the original HR images in the training phase. However, the perceptual quality of the original HR images may not be high enough as mentioned by HGGT [13]. Inspired by HGGT, we proposed a model guided ground-truth (GT) enhancement strategy to enhance

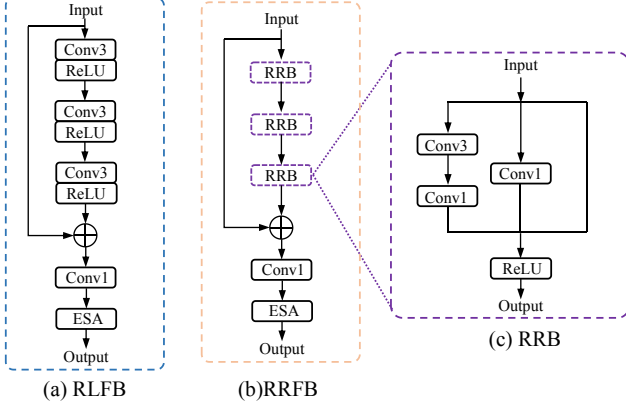


Figure 3. (a) Structure of RLFB. (b) The structure of RRFB.

the quality of HR. We first train a large network with Hybrid Attention Transformer [36] backbone for 1x super-resolution. The HR image I_{HR} is then utilized as input for 1x super-resolution, yielding an enhanced HR output I_{enh} . Then the low-resolution image I_{LR} and the enhanced HR I_{enh} is used for 4x super-resolution training. Different from HGGT, we do not conduct manual patch selection and retain the patch in the flat area since that manual selection tends to generate results that were more visually pleasing, but may not improve objective metrics. As shown in Fig. 4, the quality of the enhanced ground-truth in some patches is significantly better than the original ground-truth.

3.3. Multi-anchor Feature Distillation

In order to further enhance the performance of our lightweight model, we proposed a multi-anchor feature distillation method. As illustrated in Fig. 2, our multi-anchor feature distillation consists of two stages. In the first stage, we also train a large teacher network HAT [36], denoted as \mathcal{T} . It is worth noting that we use the enhance high-resolution image I_{enh} as discussed in Sec. 3.2 for 4x super-resolution training through minimizing the following loss:

$$L_{\mathcal{T}} = \|\mathcal{T}(I_{LR}) - I_{enh}\|_1. \quad (1)$$

After training the teacher network, we perform a multi-level distillation on the proposed student network, denoted as \mathcal{S} . Once the student network training converges, it can be restored to the RLFB structure by the reparameterization technique. During distillation, we use the feature maps extracted from four different depths of \mathcal{T} to supervise the learning of each of the four blocks in \mathcal{S} . Specifically, we minimize the following losses:

$$L_{feat} = \lambda_i \sum_{i=1}^4 \|F_i^{\mathcal{T}} - \psi(F_i^{\mathcal{S}})\|_1, \quad (2)$$

where $F_i^{\mathcal{S}}$ represents the feature map of the output of the i -th block of \mathcal{S} , while $F_i^{\mathcal{T}}$ represents the feature of the out-

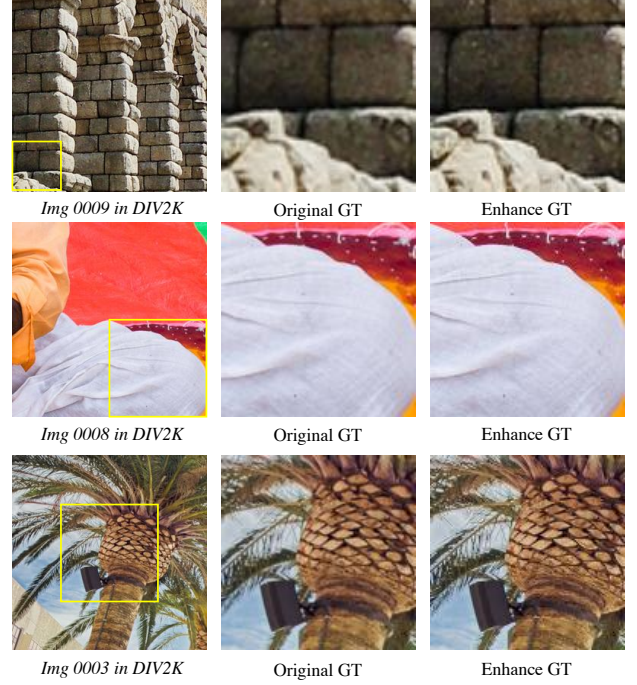


Figure 4. Visual comparison of the original GT and the enhance GT.

put of some residual hybrid attention groups (RHAGs) of \mathcal{T} , ψ represents the operation of using a 1×1 convolution to expand the feature channels of \mathcal{S} to the number of feature channels of \mathcal{T} , and λ_i is a weight for controlling the importance of the supervision from each depth level.

We also use the outputs of \mathcal{T} as pseudo ground-truth and the enhanced ground-truth to further supervise the learning of \mathcal{S} . Specifically, we compute the following losses:

$$L_{out} = \|\mathcal{T}(I_{LR}) - \mathcal{S}(I_{LR})\|_1 + \|\mathcal{S}(I_{LR}) - I_{enh}\|_1, \quad (3)$$

During distillation, the final loss is a combination of L_{feat} and L_{out} :

$$L_{dis} = L_{feat} + L_{out}, \quad (4)$$

After distillation, we employ a progressive learning strategy to finetune \mathcal{S} . We gradually increase the size of the input patch while using L_2 loss for supervised training until the model fully converges:

$$L_{pl} = \|\mathcal{S}(I_{LR}) - I_{enh}\|_2^2, \quad (5)$$

3.4. Iterative Pruning Strategy

Finally, we iteratively pruned the reparameterized student network \mathcal{S} :

$$S_p^i = \varphi(\Phi(S_p^{i-1}; r)), \quad (6)$$



Figure 5. Visual comparison of the results of ours and other methods on the validation set of DIV2K.

where Φ is the pruning operation, r is the pruning rate, φ means the finetuning operation, S_p^i means the network after the i -th pruning. Inspired by AGP [42], L_2 filter pruning is used in our iterative pruning method for model training. We stop pruning until the network cannot make effective predictions, and use the network obtained from the last effective pruning as the final network.

4. Experiments

4.1. Settings and Details

Training and Test Datasets. We adopt widely used high-quality (2K resolution) DIV2K [44] dataset which includes 800 training samples for training, following some of the prior research [4, 8, 11, 37]. We test the performance of our method on five benchmark dataset: Set5, Set14, BSD100, Urban100 and Manga109. We also test the data for DIV2K and LDSIR [45], which are provided by NTIRE 2023 Challenge on Efficient Super Resolution.

Evaluation and metrics. We use two common metrics called peak signal-to-noise ratio (PSNR) and structure similarity index (SSIM) to evaluate our method and comparison methods on the RGB space, following the evaluation settings of the NTIRE 2023 challenge on efficient super-resolution. In addition, to verify the efficiency of our meth-

ods, we statistics the number of parameters, GFLOPs, and inference time of each method network structure in a standard way as validation metrics, which are obtained statistically in the same computing environment.

Comparison methods. We select representative open-sourced methods, which include CARN [11], IMDN [4], RFDN [8], RLFN [37] and so on. The results of each method are generated by the implementations from the original authors with default settings for a fair comparison.

Implementation details. All training experiments are done on NVIDIA 2080ti. During the training phase, we use random flip and rotation augmentation and choose Adam as the optimizer. When training the teacher net and distilling the student network, we set an initial learning rate of $1e-4$, halved the learning rate every 100,000 iterations, and then used L_1 loss for supervision. When using the progressive learning strategy to finetune the distilled network, an initial learning rate of $2e-5$ is used, which is halved every 20,000 iterations. In this process, the training patch size is progressively increased to improve the performance, which is selected from [64, 128, 256, 384]. In the iterative pruning process, the ratio of each pruning is 0.05, and it is repeated three times in total. After each pruning, I_{LR} and I_{enh} are used for finetune, and 384×384 patches are used as input

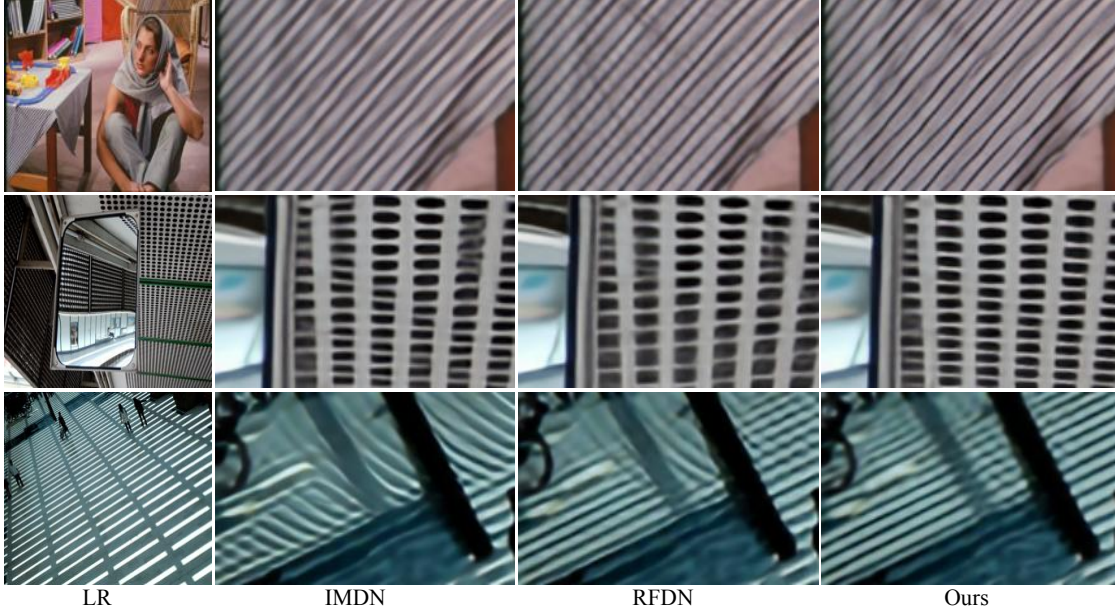


Figure 6. More visual comparison on the other benchmarks.

Method	PSNR	Time (ms)	Params (M)	GFLOPS	Activations	Memory (M)
CARN [11]	-	72.41	1.59	130.04	232.98	2192.74
EDSR [18]	-	72.36	1.52	130.05	232.98	2192.74
IMDN [4]	29.13	38.29	0.72	46.59	122.68	922.15
RFDN [8]	29.04	29.19	0.43	27.10	112.03	813.06
EFDN [43]	29.01	22.12	0.27	16.73	111.11	687.44
RLFN [37]	29.00	19.02	0.32	19.70	80.04	494.80
DIPNet (Ours)	29.00	14.77	0.24	14.90	72.97	521.02

Table 1. Comparison of our method and some recent efficient super-resolution methods. Times represent the average inference time measured on the DIV2K dataset with an NVIDIA 2080ti in milliseconds (ms). GFLOPS and memory is measured when the input is 256×256 . PSNR is the result of testing on DIV2K. The best and second-best results are marked in red and blue colors, respectively.

during finetune.

4.2. Model Complexity

In Fig. 1, we provide an overview of the deployment performance of our DIPNet. We can find that our DIPNet obtains the best inference time. To evaluate the method complexity of our model precisely, we compare several representative open source networks in Table 1. The table shows that our DIPNet consumes the least resource while maintaining 29.00 PSNR. Specifically, in terms of running time, we compare our approach with RFDN on an NVIDIA 2080ti. The iterative pruning strategy that we employ enables significant speed improvements with minimal cost, our speed is significantly faster than RFDN.

4.3. Qualitative Comparison

As shown in Fig. 5, we compare our method with some recent efficient super-resolution methods. As can be seen

from the figure, although our model is smaller, we can still obtain a good super-resolution effect. Compared with other larger methods, there is no obvious difference in the super-resolution effect. Even in some scenarios, the super-resolution effect of our method is more obvious, such as in the first and second rows in the figure, our method gets clearer lines. This is due to our use of enhanced ground-truth (GT), which makes our method more inclined to learn clearer objects during training.

4.4. Quantitative Comparison

As shown in Table 2, we compare our method with some other state-of-the-art efficient super-resolution models on four benchmark datasets Set5, Set14, and UrBan100. Experiments show that our method still achieves good results on these datasets. It is worth noting that here we do not directly use the final model used in the NTIRE competition in order to achieve better results, but used a larger model with

Scale	Model	Params (M)	Set5	Set14	BSD100	UrBan100	Manga109
			PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
×2	SRCNN [16]	0.024	36.66 / 0.9542	32.42 / 0.9063	31.36 / 0.8879	29.50 / 0.8946	35.60 / 0.9663
	FSRCNN [46]	0.012	36.98 / 0.9556	32.62 / 0.9087	31.50 / 0.8904	29.85 / 0.9009	36.67 / 0.9710
	VDSR [17]	0.666	37.53 / 0.9587	33.05 / 0.9127	31.90 / 0.8960	30.77 / 0.9141	37.22 / 0.9750
	DRCN [47]	1.774	37.63 / 0.9588	33.04 / 0.9118	31.85 / 0.8942	30.75 / 0.9133	37.55 / 0.9732
	LapSRN [48]	0.251	37.52 / 0.9591	32.99 / 0.9124	31.80 / 0.8952	30.41 / 0.9103	37.27 / 0.9740
	IDN [12]	0.579	37.83 / 0.9600	33.30 / 0.9148	32.08 / 0.8985	31.27 / 0.9196	38.01 / 0.9749
	EDSR [18]	1.370	37.91 / 0.9602	33.53 / 0.9172	32.15 / 0.8995	31.99 / 0.9270	38.40 / 0.9766
	CARN [11]	1.592	37.76 / 0.9590	33.52 / 0.9166	32.09 / 0.8978	31.92 / 0.9256	38.36 / 0.9765
	ECBSR [10]	0.596	37.90 / 0.9615	33.34 / 0.9178	32.10 / 0.9018	31.71 / 0.9250	- / -
	IMDN [4]	0.694	38.00 / 0.9605	33.63 / 0.9177	32.19 / 0.8996	32.17 / 0.9283	38.88 / 0.9774
	RFDN [8]	0.534	38.05 / 0.9606	33.68 / 0.9184	32.16 / 0.8994	32.12 / 0.9278	38.88 / 0.9773
	Ours	0.527	37.98 / 0.9605	33.66 / 0.9192	32.20 / 0.9002	32.31 / 0.9302	38.62 / 0.9770
×4	SRCNN [16]	0.057	30.48 / 0.8628	27.49 / 0.7503	26.90 / 0.7101	24.52 / 0.7221	27.58 / 0.8555
	FSRCNN [46]	0.013	30.72 / 0.8660	27.61 / 0.7550	26.98 / 0.7150	24.62 / 0.7280	27.90 / 0.8610
	VDSR [17]	0.666	31.35 / 0.8838	28.01 / 0.7674	27.29 / 0.7251	25.18 / 0.7524	28.83 / 0.8870
	DRCN [47]	1.774	31.53 / 0.8854	28.02 / 0.7670	27.23 / 0.7233	25.14 / 0.7510	28.93 / 0.8854
	LapSRN [48]	0.502	31.54 / 0.8852	28.09 / 0.7700	27.32 / 0.7275	25.21 / 0.7562	29.09 / 0.8900
	IDN [12]	0.600	31.93 / 0.8923	28.45 / 0.7781	27.48 / 0.7326	25.81 / 0.7766	30.04 / 0.9026
	EDSR [18]	1.518	31.98 / 0.8927	28.55 / 0.7805	27.54 / 0.7348	25.90 / 0.7809	30.24 / 0.9053
	CARN [11]	1.592	32.13 / 0.8937	28.60 / 0.7806	27.58 / 0.7349	26.07 / 0.7837	30.47 / 0.9084
	ECBSR [10]	0.603	31.92 / 0.8946	28.34 / 0.7817	27.48 / 0.7393	25.81 / 0.7773	- / -
	IMDN [4]	0.715	32.21 / 0.8948	28.58 / 0.7811	27.56 / 0.7353	26.04 / 0.7838	30.45 / 0.9075
	RFDN [8]	0.550	32.24 / 0.8952	28.61 / 0.7819	27.57 / 0.7360	26.11 / 0.7858	30.58 / 0.9089
	Ours	0.543	32.20 / 0.8950	28.58 / 0.7811	27.59 / 0.7364	26.16 / 0.7879	30.53 / 0.9087

Table 2. Quantitative results of the state-of-the-art efficient super-resolution models on four benchmark datasets. The best and second-best results are marked in **red** and **blue** colors, respectively.

GT Type	\mathcal{T}	\mathcal{S}
Ori.	31.27	28.94
Enh.	31.20	29.00

Table 3. PSNR of the teacher network and the student network on the DIV2K validation set when using the original GT and the enhanced GT, respectively. Ori. means training with original GT, and Enh. means training with enhanced GT.

Shallow Features KD	Deep Feature	PSNR
		28.95
✓		28.96
	✓	28.98
✓	✓	29.00

Table 4. Effect of using different degrees of distillation on PSNR. KD means knowledge distillation, Deep Feature means the features of the last block, and Shallow Features means the features of the first three blocks.

the similar structure, but even so our model is still relatively small.

In Fig. 6 we show some examples of our method and other methods on these datasets. It can be found that our method is much better than other methods in some densely textured areas. This is due to the fact that we use enhanced GT to make our method can better distinguish these repeated regular contents.

4.5. Ablation Studies

Original GT vs. Enhanced GT. We compared our enhanced ground-truth with the original ground-truth in Fig. 4, and we can see that the enhanced ground-truth has

a clearer texture. And this enhanced ground-truth will help our student network to better learn weaker textures. We further make a quantitative analysis of the effect of enhanced ground-truth in Table 3. From the table, we find that the use of enhanced ground-truth for the teacher network makes the performance of the network worse. This is because the large network has a strong learning ability. It already has the ability to learn most kinds of details, and the use of enhanced ground-truth makes it learn more noise, which leads to a decrease in its PSNR. The student network has a limited learning capability that restricts its ability to ex-

Method	PSNR	Time (ms)	Params (M)	GFLOPS	Activations	Memory (M)
KaiBai_Group	28.95	20.49	0.272	16.76	65.10	296.45
Young	28.97	22.09	0.543	33.38	61.87	293.05
NoahTerminalCV_TeamB	28.96	27.83	0.209	13.34	118.71	188.21
Sissie_Lab	29.00	30.34	0.461	28.85	107.07	628.94
Antins_cv	29.00	20.92	0.315	20.07	70.82	488.61
CMVG	29.01	24.42	0.307	18.98	81.55	454.51
DFCDN	29.00	18.71	0.245	15.49	82.76	376.99
Zapdos	28.96	18.59	0.352	21.97	63.01	420.50
DIPNet (Ours)	29.04	18.30	0.243	14.90	72.97	495.91

Table 5. Quantitative comparison of our results with those of other NTIRE 2023 Challenge on Efficient Super Resolution participating teams. The best and second-best results are marked in red and blue colors, respectively.

Pruning times	0	1	2	3
PSNR	29.042	29.034	29.018	29.001

Table 6. In the iterative pruning process, the PSNR of the model on the DIV2K validation set after each pruning. The 0th represents the PSNR of the model before pruning.

Pruning Type	One Stage Pruning	Iterative pruning
L1	28.883	28.946
L2	28.894	29.001

Table 7. The PSNR when the model is cut to the same size using different pruning strategies

plore all possible directions during the training process. As a result, the network may only focus on certain directions and fail to learn other important features necessary for high-quality image generation. However, by incorporating an enhanced ground-truth, which contains more information than the original ground-truth, the student network can overcome its limited capacity to learn details and achieve better performance in terms of PSNR. The enhanced ground-truth provides additional guidance to the network, allowing it to learn a wider range of features and produce more accurate and detailed images.

Multi-stage feature distillation. The results in Table 4 show the advantages of our multi-level feature distillation. Compared with direct end-to-end training of small models, our method can significantly improve the model accuracy by about 0.05dB, which benefits from the strong representation of the teacher model capacity. At the same time, our student model does not add additional overhead. Meanwhile, we found that only using deep features is better than using only shallow features, and the effect is best when the two are used at the same time.

Iterative Pruning Strategy. We show the changes of PSNR after each pruning in the iterative pruning process in Table 6. It can be seen that the PSNR decline in the previous pruning is relatively small, and there will be a relatively ob-

vious decline in PSNR when pruning later. But overall, the reduction in PSNR brought by our iterative pruning method is very small. Table 7 shows the impact of using different pruning methods on the results. The experimental results show that using iterative pruning is better than one-time pruning. At the same time, L2 pruning is better than L1 pruning for our task.

4.6. NTIRE 2023 Challenge on Efficient SR

In this competition, we design a lightweight network to maintain the PSNR of 29.00dB on DIV2K validation set by reparameterization and multi-stage feature distillation. It is worth mentioning that the baseline of the competition is RFDN. We follow the official evaluation setting and report the number of parameters, FLOPs, runtime, peak memory consumption, activations, and number of convolutions in Table 5. Compared with AIM 2020 winner solution E-RFDN, our model can decrease 43.9% parameters, 45.0% FLOPs, 48.5% runtime, 37.1% peak memory consumption and 34.9% activations. Compared with other participants in NTIRE 2023 challenge [49] on efficient super-resolution, our model achieves the best inference speed.

5. Conclusion

In this paper, we propose a novel approach to efficient single image super-resolution by improving training strategies instead of solely depending on network design. Specifically, we leverage enhanced ground-truth images as additional supervision and employ a multi-stage lightweight training strategy that combines distillation, progressive learning, and pruning. Our experiments demonstrate the effectiveness of our method, achieving state-of-the-art performance in terms of time consumption and model size on the NTIRE 2023 efficient super-resolution challenge. Our contributions include introducing the use of enhanced GT images to improve the learning ability of lightweight networks and proposing a novel multi-stage lightweight training strategy.

References

- [1] Yu Chen, Ying Tai, Xiaoming Liu, Chunhua Shen, and Jian Yang. Fsrnet: End-to-end learning face super-resolution with facial priors. In *CVPR*, pages 2492–2501, 2018. 1
- [2] Ying Fu, Tao Zhang, Yinqiang Zheng, Debing Zhang, and Hua Huang. Hyperspectral image super-resolution with optimized rgb guidance. In *CVPR*, pages 11661–11670, 2019. 1
- [3] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *ECCV*, pages 184–199, 2014. 1
- [4] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *ACM MM*, pages 2024–2032, 2019. 1, 2, 3, 5, 6, 7
- [5] Andrey Ignatov, Radu Timofte, Maurizio Denna, and Abdel Younes. Real-time quantized image super-resolution on mobile npus, mobile ai 2021 challenge: Report. In *CVPR*, pages 2525–2534, 2021. 1
- [6] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *CVPR*, pages 3147–3155, 2017. 1, 2
- [7] 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 *CVPR*, pages 1874–1883, 2016. 1, 2
- [8] Jie Liu, Jie Tang, and Gangshan Wu. Residual feature distillation network for lightweight image super-resolution. In *ECCV*, pages 41–55, 2020. 1, 2, 3, 5, 6, 7
- [9] Pengxu Wei, Hannan Lu, Radu Timofte, Liang Lin, Wangmeng Zuo, Zhihong Pan, Baopu Li, Teng Xi, Yanwen Fan, Gang Zhang, et al. Aim 2020 challenge on real image super-resolution: Methods and results. pages 392–422, 2020. 1
- [10] Xindong Zhang, Hui Zeng, and Lei Zhang. Edge-oriented convolution block for real-time super resolution on mobile devices. In *ACM MM*, 2021. 1, 7
- [11] Namhyuk Ahn, Byungkoo Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *ECCV*, 2018. 2, 3, 5, 6, 7
- [12] Zheng Hui, Xiumei Wang, and Xinbo Gao. Fast and accurate single image super-resolution via information distillation network. In *CVPR*, pages 723–731, 2018. 2, 7
- [13] Du Chen, Jie Liang, Xindong Zhang, Ming Liu, Hui Zeng, and Lei Zhang. Human guided ground-truth generation for realistic image super-resolution. In *CVPR*, 2023. 2, 3
- [14] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *CVPR*, pages 13733–13742, 2021. 2
- [15] Xiaojie Chu, Liangyu Chen, Chengpeng Chen, and Xin Lu. Improving image restoration by revisiting global information aggregation. In *ECCV*, pages 53–71, 2022. 2
- [16] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2, 7
- [17] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *CVPR*, pages 1646–1654, 2016. 2, 7
- [18] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *CVPRW*, pages 136–144, 2017. 2, 6, 7
- [19] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *CVPR*, pages 2472–2481, 2018. 2
- [20] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *CVPR*, pages 1664–1673, 2018. 2
- [21] Ziwei Luo, Lei Yu, Xuan Mo, Youwei Li, Lanpeng Jia, Haoqiang Fan, Jian Sun, and Shuaicheng Liu. Ebsr: Feature enhanced burst super-resolution with deformable alignment. In *CVPR*, pages 471–478, 2021. 2
- [22] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *ECCV*, 2018. 2
- [23] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image restoration. *IEEE TPAMI*, 43(7):2480–2495, 2020. 2
- [24] Ziwei Luo, Youwei Li, Shen Cheng, Lei Yu, Qi Wu, Zhihong Wen, Haoqiang Fan, Jian Sun, and Shuaicheng Liu. Bsrt: Improving burst super-resolution with swin transformer and flow-guided deformable alignment. In *CVPR*, pages 998–1008, 2022. 2
- [25] 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 *CVPR*, pages 4681–4690, 2017. 2
- [26] Mehdi SM Sajjadi, Bernhard Scholkopf, and Michael Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. In *ICCV*, pages 4491–4500, 2017. 2
- [27] 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 *ECCV*, pages 0–0, 2018. 2
- [28] Xiaozhong Ji, Yun Cao, Ying Tai, Chengjie Wang, Jilin Li, and Feiyue Huang. Real-world super-resolution via kernel estimation and noise injection. In *CVPRW*, 2020. 2
- [29] Youwei Li, Haibin Huang, Lanpeng Jia, Haoqiang Fan, and Shuaicheng Liu. D2c-sr: A divergence to convergence approach for real-world image super-resolution. In *ECCV*, pages 379–394, 2022. 2
- [30] Jinjin Gu, Hannan Lu, Wangmeng Zuo, and Chao Dong. Blind super-resolution with iterative kernel correction. In *CVPR*, pages 1604–1613, 2019. 2

- [31] Sefi Bell-Kligler, Assaf Shocher, and Michal Irani. Blind super-resolution kernel estimation using an internal-gan. *NeurIPS*, 32, 2019. 2
- [32] Ziwei Luo, Haibin Huang, Lei Yu, Youwei Li, Haoqiang Fan, and Shuaicheng Liu. Deep constrained least squares for blind image super-resolution. In *CVPR*, pages 17642–17652, 2022. 2
- [33] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, pages 694–711, 2016. 2
- [34] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014. 2
- [35] Jingyun Liang, Jiezhong Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *ICCV*, pages 1833–1844, 2021. 2
- [36] Xiangyu Chen, Xintao Wang, Jiantao Zhou, and Chao Dong. Activating more pixels in image super-resolution transformer. *arXiv preprint arXiv:2205.04437*, 2022. 2, 4
- [37] Fangyuan Kong, Mingxi Li, Songwei Liu, Ding Liu, Jingwen He, Yang Bai, Fangmin Chen, and Lean Fu. Residual local feature network for efficient super-resolution. In *CVPR*, pages 766–776, 2022. 3, 5, 6
- [38] Zongcai Du, Ding Liu, Jie Liu, Jie Tang, Gangshan Wu, and Lean Fu. Fast and memory-efficient network towards efficient image super-resolution. In *CVPR*, pages 853–862, 2022. 3
- [39] 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. pages 561–572, 2023. 3
- [40] Yawei Li, Kai Zhang, Radu Timofte, Luc Van Gool, Fangyuan Kong, Mingxi Li, Songwei Liu, Zongcai Du, Ding Liu, Chenhui Zhou, et al. Ntire 2022 challenge on efficient super-resolution: Methods and results. In *CVPR*, pages 1062–1102, 2022. 3
- [41] Andrey Ignatov, Radu Timofte, Maurizio Denna, Abdel Younes, Ganzorig Gankhuyag, Jingang Huh, Myeong Kyun Kim, Kihwan Yoon, Hyeon-Cheol Moon, Seungho Lee, et al. Efficient and accurate quantized image super-resolution on mobile npus, mobile ai & aim 2022 challenge: report. pages 92–129, 2023. 3
- [42] Michael Zhu and Suyog Gupta. To prune, or not to prune: exploring the efficacy of pruning for model compression. *arXiv preprint arXiv:1710.01878*, 2017. 5
- [43] Yan Wang. Edge-enhanced feature distillation network for efficient super-resolution. In *CVPR*, pages 777–785, 2022. 6
- [44] Eirikur Agustsson and Radu Timofte. NTIRE 2017 challenge on single image super-resolution: Dataset and study. In *CVPRW*, pages 126–135, 2017. 5
- [45] Yawei Li, Kai Zhang, Jingyun Liang, Jiezhong 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 *CVPRW*, 2023. 5
- [46] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *ECCV*, pages 391–407, 2016. 7
- [47] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *CVPR*, pages 1637–1645, 2016. 7
- [48] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *CVPR*, pages 624–632, 2017. 7
- [49] Yawei Li, Yulun Zhang, Luc Van Gool, Radu Timofte, et al. Ntire 2023 challenge on efficient super-resolution: Methods and results. In *CVPRW*, 2023. 8