DyConvSR: Lightweight Image Super-Resolution with Dynamic Convolutions

Tien-Thong Doan, and Hoang-Giap Nguyen

Falculity of Electrical Engineering, Ho Chi Minh city University of Technology
thong.doanbk19@hcmut.edu.vn, nhgiap@hcmut.edu.vn

Abstract. In the past few years, high-resolution videos and images have become popular for display on edge devices such as mobile phones. While recent studies of super-resolution problems require a huge resource for inference, this brings a great challenge for super-resolution networks to achieve better inference time on commercial GPUs. This paper presents a fast and lightweight super-resolution network that can leverage the power of Dynamic Convolution. We further adopt many efficient network designs such as pixel-unshuffle, high frequent block, repeat upscaling, and methods to increase the model's capacity like spatial attention module. On average, our model is more than **10 times smaller** than most current models in efficient super-resolution problem. Overall, our technique achieves a comparable performance with contemporary state-of-the-art methods with much fewer parameters. The code will be available at https://github.com/doantienthongbku/DyConvSR-PytorchLightning.

Keywords: Super-Resolution · Lightweight · Dynamic Convolution.

1 Introduction

Image super-resolution (SR) is a computer vision task, which refers to the process of recovering high-resolution (HR) images from low-resolution (LR) images. SR is a highly challenge problem, and many studies with different approaches to yield reasonable and high-quality HR images. Along with building model files focusing on visual quality, many SR networks focus on optimizing runtime performance while still maintaining good visual effects. In recent years, super-resolution has been widely used to improve the visual quality of images and videos, especially in edge devices such as mobile phones. Because of the limited resources of edge devices, optimizing runtime performance is an extremely important issue. This paper will focus on the problem of lightweight image SR which is needed in optimized timing applications.

Facing the aforementioned challenges, we started to design a simple network that includes only parts that are needed for building an SR network, shallow feature extraction, computation on deep features and a final up-scaling module. We explore and evaluate the performance by replacing the deep features extraction part with different efficient network structures such as information multi-distillation block (IMDB) [9], Residual Feature Distillation Block (RFDB) [15]

and deep feature extraction of RT4KSR [23]. These structures can significantly increase the performance of the network, but they also increase the number of parameters. Considering the efficiency of the SR network, we expect a simple topology but strong for deep feature extraction, and Dynamic Convolution is the perfect candidate. Dynamic Convolution[4] represents a significant advancement in convolutional neural networks (CNN), which offers a powerful method that increases the capacity and flexibility of convolutional neural networks (CNN) without adding redundant computation cost, and can be applied in various SR architectures.

Besides the deep feature extraction, to achieve a more efficient network, we explore and apply other efficient architectures such as pixel shuffle and pixel unshuffle operations respectively at the beginning and the end of the network. We discard all the local skip connections to reduce the computation and only retain the global skip connection with repeat upscaling method from [7] to keep the stability and global information of the network. We also added the high frequent block (HFB) to extract complementary high-frequency details to the result. Moreover, spatial attention (SA) [22] is incorporated to increase the network's adaptability.

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

1. We successfully adapt Dynamic Convolution in the network and demonstrate its power through experiments on the super-resolution problem. Our design is a straightforward network design that has ten times fewer model parameters while maintaining comparable performance.
2. We conduct the investigation about balancing between quantitative and runtime performance with various methods: Dynamic Convolution, High Frequent Block (HFB), Spatial Attention (SA), and skip connection.

2 Related Work

2.1 Deep learning-based Single Image Super-Resolution

Single image super-resolution tasks are actively researched in the field of computer vision. Since the pioneering work of Super-Resolution Convolutional Neural Network (SRCNN [5]), more complicated network architectures have been designed. FSRCNN [6] is a faster version of SRCNN, by replacing the ReLU with PReLU activation, reforming the architecture to be deeper and more efficient. VDSR [10], EDSR [14], RDN [27], and RCAN [26] are the approaches by expanding the number of parameters to leverage the power of deeper networks and many architectures to reconstruct better quantitative performance. While SRGAN [12], ESRGAN [20], and CinGAN [28] use GAN-based mechanisms focus on the visual effect of the output images. Especially, SwinIR [13] proposes the Swin Transformer-based image restoration method and achieves impressive performance.

2.2 Dynamic Convolution

Dynamic Convolution [4] represents a significant advancement in the field of computer vision and image processing, which offers a powerful method that increases the capacity and flexibility of convolutional neural networks (CNN) without adding redundant computation cost, and can be readily applied in various CNN architectures include SR network. This method has shown to be successful in fundamental computer vision tasks including classification, object identification, and segmentation are utilized in DyNet [25] to replace traditional convolution. In this paper, we will demonstrate the power of Dynamic Convolution in low-level tasks such as denoising and super-resolution.

3 Method

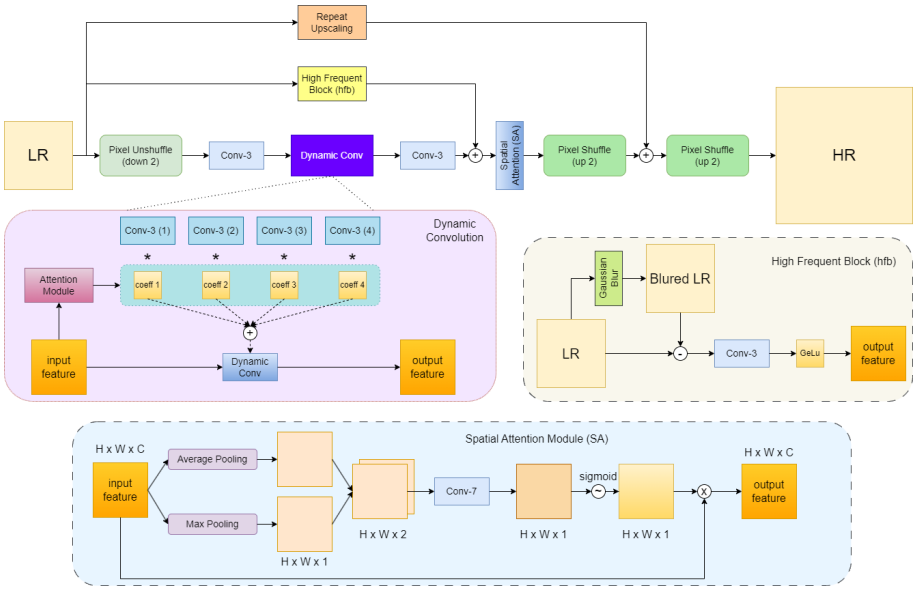

Fig. 1: Architecture of Lightweight Image Super-Resolution with Dynamic Convolutions (DyConvSR)

Before going deeper into the network architectures, we explain in brief about the floating point operations per second (FLOPs). The equation of FLOPs of traditional convolution is described as follows:

$$FLOPs = C_{in} \times K^2 \times H \times W \times C_{out} \tag{1}$$

Where C_{in}, H, W are the dimensions of the input feature map, K is the kernel size and C_{out} is the number of output channels.

3.1 Pixel-Shuffle and Pixel-Unshuffle

Pixel-Shuffle is a periodic shuffling operator used in super-resolution models, which transfers a $(H \times W \times C * N^2)$ tensor to a tensor with shape $(H * N \times W * N \times C)$. While Pixel-Unshuffle is the invert of Pixel-Shuffle, which transfers a $(H * N \times W * N \times C)$ tensor to a tensor with shape $(H \times W \times C * N^2)$.

We use Pixel-Unshuffle to downscale and increase the number of channels at the beginning of the network to reduce the computational cost of the second and following convolution layers. According to Equ. 1, the FLOPs decrease by the factor N^2 since does not change the capacity of information in the input image. At the end of the network, we add two Pixel-Shuffle layers to reconstruct the HR images. Although this design does not change the amount of information, the overall performance usually slightly degrades if we simply downscale the input by only applying pixel-unshuffle, but this is a deserved trade-off to improve runtime performance.

(a) Input Image (b) Blurred Image (c) HF Information

Fig. 2: Extracting high frequency (HF) information.

3.2 High Frequent Block (HFB) and Skip Connection

One of the main weaknesses of the efficient SR network with simple architecture is the lack of high-frequency detail in the output images. It is essential to use a module to increase the high-frequency detail in the network design. Inspired by [21], we integrate a simplified version of the high-frequency branch into our network. We use the Gaussian blur operation (a low-pass filter) on the input image to generate the blurred images, this blurred version contains only low-frequency characteristics. Then subtract from the input LR image to obtain the HF components, the visual result is shown in Fig. 2. The high-frequency then are extracted by a 3×3 convolution and GeLU activation to have better information before injecting back to the main branch.

Skip connection also accounts for a large number of computational resources besides the convolution layers. The skip connection in the SR network is divided

into 2 types: local skip connection is used inside the sub-block network architecture, global skip connection is the addition of input images upscaled to the end of the network before generating the output HR images. The global skip connection is used to add the base information from the LR image, this can help reduce the burden on the model and stabilize the training process. According to [7], we use the repeat upscaling block instead of traditional interpolation for the global skip connection to reduce the computational cost.

To balance between the performance and the runtime, we only use repeat upscaling for global skip connection and the high-frequency branch are the external branch and remove all local skip connections.

3.3 Spatial Attention

Throughout the investigation of DIV2K and Flickr2K datasets, we found that different image regions have different restoration difficulties and should be concentrated to different degrees in the model, for example, the flat area (sky, land, ...) is naturally easier to process than textures (hair, feathers, ...). This indicates that we need a module to help the model focus more on the more difficult region instead of considering all areas equally.

A Spatial Attention Module is a module for spatial attention in convolutional neural networks that was first introduced in [22], it generates a spatial attention map by utilizing the inter-spatial relationship of features. Because multiplying a whole tensor with spatial attention takes up a lot of computing space, we add only one spatial attention after the last Convolution layer before the first Pixel Shuffle layer. This position is in the decision stage and needs an attention module to decide what is important to process next. We use standard attention in our network with kerner_size = 7, which is the best case that the author recommends.

3.4 Network Architecture

The whole architecture is described in Fig. 1. Our architecture mainly consists of three branches: the main branch is essentially in charge of feature extraction and reconstruction, the high-frequency branch (HFB) is responsible for learning high-frequency features, and the repeat upscaling branch helps keep the base information and stabilize the training process.

The first pixel-unshuffle layer takes the input LR picture, converts its resolution to the channel dimension, and then passes it through a traditional convolution to extract shallow features and then fed into the dynamic convolution. The dynamic convolution is configured with 4 different convolution kernels, these kernels will adaptively to different inputs throughout the attention modules. After the dynamic, a 3×3 traditional convolution layer is used to convert the number of channels to 48 so that the feature map can be restored to HR resolution. Then the tensor map is added with the high-frequent feature from the HFB branch and adjusted by the spatial attention (SA) layer, and the end of the network is two pixel-shuffle layers to restore the tensor to the target HR image.

4 Experiment

4.1 Experimental setup

Dataset and Metrics. We use DIV2K [1] and Flick2K [14] datasets for training and DIV2K validation set for validation. For testing, we use the common dataset to validate SR benchmark, namely Set5 [3], Set14 [24], Urban100 [8], and BSD100 [18]. We evaluate PSNR and SSIM metrics on Y channel of the transformed YCbCr space. In the training dataset, to increase the number of images, we crop the original images on the dataset to the smaller ones with sizes 300×300, and the step is 150, similar to sliding window cropping. Thanks to this strategy, from the original dataset with $3,450$ images, we have a new training dataset with nearly $323,296$ images.

Table 1: Quantitative comparison between our model (DyConvSR) with recent **state-of-the-art** methods. We compare PSNR and SSIM (Y) for $\times 2$ on standard benchmarks.

Methods	#Params (K)	**Set5 [3]**	**Set14 [24]**	**BSD100 [18]**	**Urban100 [8]**
Bicubic	-	34.01/.9309	31.27/.8652	29.99/.8539	28.65/.8498
LapSRN [11]	251 (+418%)	37.52/.9591	32.99/.9124	31.80/.8952	30.41/.9103
CARN [2]	1592 (+2653%)	37.76/.9590	33.52/.9166	32.09/.8978	31.92/.9256
IMDN [9]	694 (+1157%)	38.00/.9605	33.63/.9177	32.19/.8996	32.17/.9283
LatticeNet [17]	756 (+1260%)	38.15/.9610	33.78/.9193	32.25/.9005	32.43/.9302
SwinIR [13]	878 (+1463%)	38.14/.9611	33.86/.9206	32.31/.9012	32.76/.9340
DyConvSR	60	35.98/.9497	31.57/.9017	30.95/.8813	28.51/.8798

Training Details. We extract random crops of size 256×256 from the RGB training set and further augment the crops by **random rotation** (0°, 90°, 180°, 270°), **random horizontal** and **random vertical** flipping. LR images are generated online using bicubic downsampling of the original HR images. We use **ADAMW** [16] optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$ to minimize the $L1$ loss between the SR output and HR target for 100 epochs with the batch size set to 64 and an initial learning rate of $1e-3$, along with a step scheduler with step size 20 epochs and decay factor 0.5. Our model is trained on GPU NVIDIA GeForce RTX 4090 and the training process takes along 12 hours for each experiment.

4.2 Result

Table.1 shows the comparison between our model and recent state-of-the-art methods. On average, our model is **1390% smaller** than the approaches we compare them to, even though these approaches are considered "lightweight".

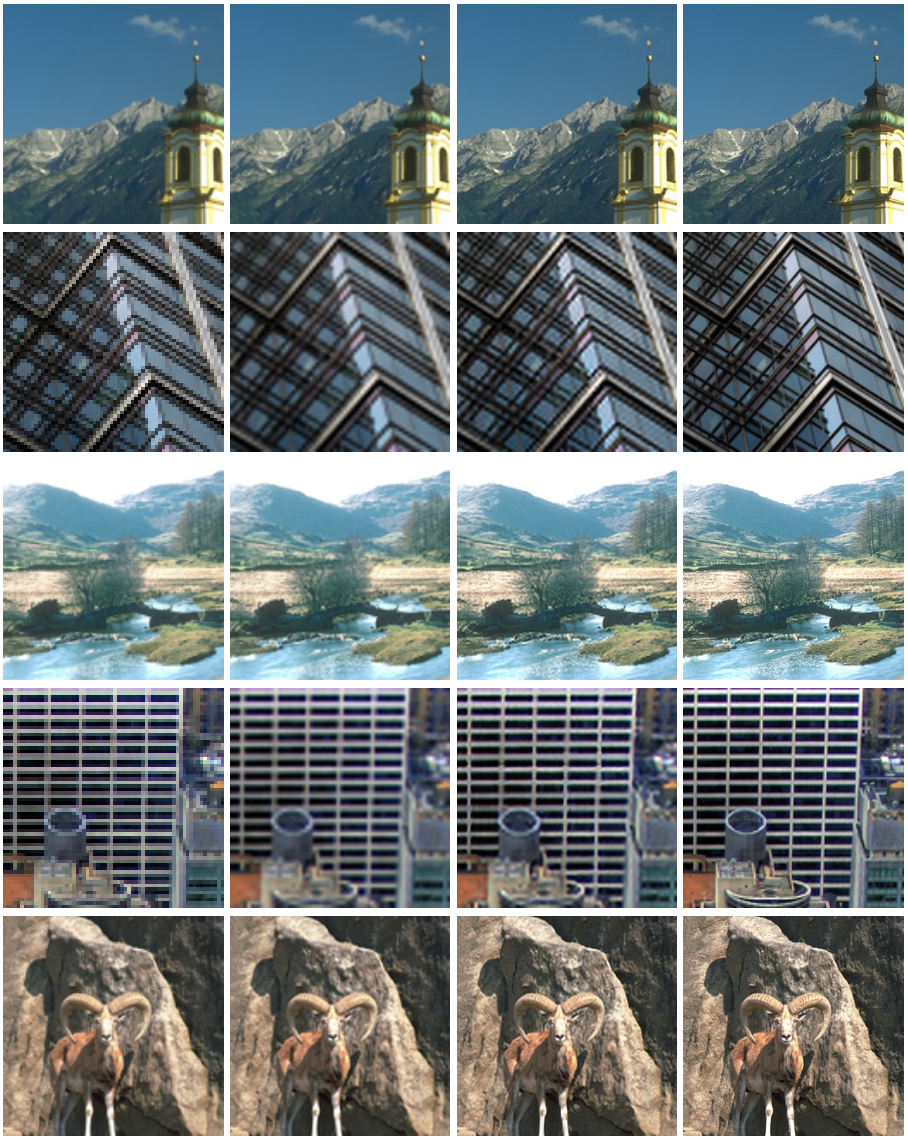

Fig. 3: Rendered samples from BSD100 [18] and Urban100 [8] benchmarks.

While there is still a measurable gap between our model and the other, our performance is still impressive compared to the number of parameters.

In Fig. 3 shows extracted crops from BSD100 [18] and Urban100 [8] benchmarks datasets. Our model shows an impressive performance in reconstructing the high-frequency features from the LR image. Moreover, our model demonstrates a superior output quality in both structural and natural input images.

5 Conclusion

In this paper, we conduct a comprehensive analysis of various designs to efficiently super-resolution problems such as dynamic convolution, high-frequency block, spatial attention, and skip connection. By applying these designs, we propose DyConvSR - a lightweight super-resolution network which reduces the number of parameters significantly while still maintaining an impressive performance.

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.
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.
3. Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012.
4. Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic convolution: Attention over convolution kernels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11030–11039, 2020.
5. 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.
6. Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 391–407. Springer, 2016.
7. 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.
8. 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.
9. 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.

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. 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.
12. 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.
13. 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.
14. 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.
15. 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.
16. Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
17. Xiaotong Luo, Yuan Xie, Yulun Zhang, Yanyun Qu, Cuihua Li, and Yun Fu. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, pages 272–289. Springer, 2020.
18. 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.
19. Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 114–125, 2017.
20. 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.
21. Xiaotian Weng, Yi Chen, Zhichao Zheng, Yanhui Gu, Junsheng Zhou, and Yudong Zhang. A high-frequency focused network for lightweight single image super-resolution. *arXiv preprint arXiv:2303.11701*, 2023.
22. Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.
23. 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*, pages 1522–1532, 2023.

24. 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.
25. Yikang Zhang, Jian Zhang, Qiang Wang, and Zhao Zhong. Dynet: Dynamic convolution for accelerating convolutional neural networks. *arXiv preprint arXiv:2004.10694*, 2020.
26. 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.
27. Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2472–2481, 2018.
28. Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.