Hybrid Pixel-Unshuffled Network for Lightweight Image Super-resolution

Bin Sun1,3, Yulun Zhang2, Songyao Jiang1, Yun Fu1,3

1 Northeastern University, Boston, MA, USA
2 ETH Zürich, Zürich, Switzerland
3 AInnovation Labs Inc., Boston, MA, USA
sun.bi@northeastern.edu, {yulun100, jiangsongyao}@gmail.com, yunfu@ece.neu.edu

Abstract

Convolutional neural network (CNN) has achieved great success on image super-resolution (SR). However, most deep CNN-based SR models take massive computations to obtain high performance. Downsampling features for multi-resolution fusion is an efficient and effective way to improve the performance of visual recognition. Still, it is counter-intuitive in the SR task, which needs to project a low-resolution input to high-resolution. In this paper, we propose a novel Hybrid Pixel-Unshuffled Network (HPUN) by introducing an efficient and effective downsampling module into the SR task. The network contains pixel-unshuffled downsampling and Self-Residual Depthwise Separable Convolutions. Specifically, we utilize pixel-unshuffle operation to downsample the input features and use grouped convolution to reduce the channels. Besides, we enhance the depthwise convolution’s performance by adding the input feature to its output. The comparison findings demonstrate that, with fewer parameters and computational costs, our HPUN achieves and surpasses the state-of-the-art performance on SISR. All results are provided in the github https://github.com/Sun1992/HPUN.

Introduction

Single Image Super-Resolution (SISR) is a fundamental vision task which involves accurately reconstructing a high-resolution (HR) image from a low-resolution (LR) image. SISR has been utilized on various high-level tasks such as face synthesis (Yin et al. 2020, 2021), medical imaging (Shi et al. 2013), surveillance imaging (Zou and Yuen 2012), and image generation (Karras et al. 2017). Dong et al. (Dong et al. 2014) first introduced CNN into SISR and achieved impressive performance in 2014. Afterward, more deep CNN methods are proposed for the super-resolution tasks (Schulter, Leistner, and Bischof 2015; Huang, Singh, and Ahuja 2015; Kim, Kwon Lee, and Mu Lee 2016a,b; Lim et al. 2017; Tong et al. 2017; Tai et al. 2017; Zhang, Zuo, and Zhang 2018; Zhang et al. 2018b). Among these, one of the most fundamental architectures is EDSR (Lim et al. 2017). However, these networks need expensive computation resources, which is the main bottleneck for their deployment on mobile devices for real-time purposes.

In recent years, a lot of manually designed lightweight structures have been proposed (Sifre and Mallat 2014; Howard et al. 2017; Chollet 2017; Iandola et al. 2017; Sandler et al. 2018; Zhang et al. 2018a; Ma et al. 2018; Howard et al. 2019; JTan and Le 2019; Han et al. 2020). Among these structures, the most fundamental one is the depthwise convolution layer (Sifre and Mallat 2014), which processes the spatial information with a single convolution on each input feature. A $1 \times 1$ convolution layer named pointwise layer is usually deployed around the depthwise convolution layer for the communication among channels (Howard et al. 2017; Sandler et al. 2018; Howard et al. 2019; Zhang et al. 2018a; Ma et al. 2018). However, such structures are not popular in the SISR due to their significant performance loss. CARN (Ahn, Kang, and Sohn 2018) tried to use a similar structure as MobileNet (Howard et al. 2017) on SISR in 2018. They utilized the group convolution to reduce the parameters, but they had to introduce a complicated recurrent scheme to improve the performance. As shown in Figure 1, the computation costs and parameters of CARN are not satisfied. Therefore, it is still a main challenge to effectively implement depthwise convolution based lightweight structures to the image super-resolution task.

Besides using lightweight operations, the computation
costs can be alleviated by reducing the size of feature maps (Tan and Le 2019; Howard et al. 2017; Chollet 2017; Iandola et al. 2017; Sandler et al. 2018; Zhang et al. 2018a; Ma et al. 2018; Howard et al. 2019; JTan and Le 2019; Han et al. 2020). Meanwhile, size-reduced features can also improve high-level representations by merging with higher-resolution features in many tasks (Sun et al. 2019; Wang et al. 2020). However, it is counter-intuitive to apply downsampling modules in SISR since SISR is an upsampling task that restores information of a low-resolution image. In contrast, the downsampling operation usually causes significant information loss. To direct the reconstruction, Haris et al. (Haris, Shakhnarovich, and Ukita 2018) suggested an iterative error-correcting feedback method that computes both up- and down-projection errors. Furthermore, Li et al. (Li et al. 2019) also proposed a framework that introduced the downsampling module into SISR to generate high-level representations. Their success shows the possibility of getting pleasing high-resolution images through downsampling operations. However, they still had to utilize a recurrent scheme to resist the performance drop, which heavily increased the parameters and computation costs.

This paper explores an effective way to design a lightweight network with depthwise convolutions and downsampling operations. Specifically, we develop a simple yet effective module named Self-Residual Depthwise Separable Convolution to overcome the drawback in Depthwise Separable Convolution (DSC) (Howard et al. 2017) without any additional parameters. Inspired by the previous explorations on downsampling features (Sun et al. 2019; Haris, Shakhnarovich, and Ukita 2018; Li et al. 2019; Gu et al. 2019), we propose a pixel-unshuffled downsampler constructed with the pixel-unshuffle operation, max-pooling, and group convolution to further enhance the performance of DSC with similar computation costs as depthwise convolution. Specifically, the pixel-unshuffle operation is the reverse operation of pixel-shuffle (Shi et al. 2016), which can help to avoid information loss shown by Gu et al. (Gu et al. 2019) The scheme is shown in Figure 2. Moreover, we propose a practical, lightweight module named Pixel-Unshuffled Block (PUB) constructed with the pixel-unshuffled downsampling and the Self-Residual DSC. Last, we replace one Self-Residual DSC in the PUB with a standard convolution layer and construct a Hybrid Pixel-Unshuffled Network (HPUN) to achieve the state-of-the-art (SOTA) performance and slightly increase the number of the HPUN to beyond the SOTA performance. The overall comparison is shown in Figure 1. The main contributions are summarized as:

- We propose the Self-Residual DSC to overcome the defects of the depthwise convolution in the SISR task with a simple and effective operation, which barely needs computation and additional parameters.
- We propose a novel downsampling module with the pixel-unshuffle operation, which is our key module to enhance the performance.
- We propose a lightweight module named PUB with our Self-Residual DSC and the pixel-unshuffled downsampler, which can provide reliable performance with a few parameters and computation costs.
- We propose the Hybrid Pixel-Unshuffled Block (HPUB) by integrating the standard convolution into the PUB, and construct the effective and efficient HPUN to achieve a new SOTA performance with a few parameters and Multi-Adds comparing the baselines.
- We discover the relationship between PSNR and the Normalized Mean Error (NME) among the shallow features and deep features based on our ablation study, which may be valuable in designing the network for SISR. We will discuss the details in the experiment section.

**Related Work**

**Deep Super Resolution.** SRCNN developed the initial end-to-end system that converts the interpolated LR images to their HR counter-parts (Dong et al. 2014). The SRCNN was further improved by its successors with advanced network architectures (Kim, Kwon Lee, and Mu Lee 2016a; Zhang et al. 2017b). As studied in (Dong, Loy, and Tang 2016), computational costs are quadratically increased by this upsampling operation in data preprocessing. To solve the problem, an efficient sub-pixel convolution layer that upsampled the last LR feature maps to HR was introduced in ESPCN (Shi et al. 2016). It was also adopted in residual-learning networks SRResNet (Ledig et al. 2017) and EDSR (Lim et al. 2017). The performance of the SISR was then further improved by stacking more blocks with dense residuals (Zhang et al. 2017a; Zhang and Patel 2018; Zhang et al. 2018c, 2019).

**Lightweight Super Resolution.** LapSRN (Lai et al. 2017) reduced the computation complexity by removing the bicubic interpolation before prediction. Inspired by LapSRN, a lot of works started to reconstruct the HR image from...
the origin LR input. Recursive learning was first introduced by DRCN (Kim, Kwon Lee, and Mu Lee 2016b). Then it was widely used to reduce the parameters with weight sharing strategy (Tai, Yang, and Liu 2017; Tai et al. 2017; Haris, Shakhnarovich, and Ukita 2018; Ahn, Kang, and Sohn 2018; Li et al. 2019). Besides the recurrent scheme, IDN (Hui, Wang, and Gao 2018) and CARN (Ahn, Kang, and Sohn 2018) introduced the group convolution for the lightweight purpose. As the success of the residual operation in SISR, many works (Hui et al. 2019; Luo et al. 2020) adopted the residual into their lightweight design to keep the performance. A recent work named SMSR (Wang et al. 2021) reduced the parameters and computation costs with pruning. Different with SMSR, we manually design the lightweight network which can be further improved by pruning.

Deep Lightweight Structure. In recent years, as the deep-learning models become deeper and larger, many researchers have been working on the lightweight networks. An activation function named ReLU was proposed for speed-up purpose (Glorot, Bordes, and Bengio 2011). There was a flattened CNN architecture that sped up the forward feeding presented in (Jin, Dundar, and Culurciello 2014). The concept of depthwise separable convolution was initially proposed by Sifre et al. (Sifre and Mallat 2014) and was widely adopted in Inception (Ioffe and Szegedy 2015), Xception (Chollet 2017), ShuffleNets (Zhang et al. 2018a; Ma et al. 2018), MobileNets (Howard et al. 2017; Sandler et al. 2018), CondenseNet (Huang et al. 2018), and LRPR-Net (Sun et al. 2021). Besides, researchers suggested using Neural Architecture Search (NAS) in addition to manually created lightweight architectures to locate the ideal lightweight network (Liu, Simonyan, and Yang 2019; Zoph et al. 2018; Cai, Zhu, and Han 2019; Wu et al. 2019; Howard et al. 2019; JTan and Le 2019). All these networks are constructed based on the depthwise convolution as well. Thus, it is necessary to explore an effective way to implement the depthwise convolution on SISR. In this work, we provide a module that can greatly improve the depthwise convolution’s performance for super resolution tasks.

Proposed Method
We propose a lightweight structure called Hybrid Pixel-Unshuffled Block to replace the traditional Residual Convolution Block, which is shown in Figure 3(a). Our proposed method has three parts: a standard convolution layer, the proposed pixel-unshuffled downsampling, and the proposed PUB. Specifically, the PUB is an integration of the pixel-unshuffled downsampling and the Self-Residual DSC. Therefore, we organize this section as follows: first, we will introduce the details of the Self-Residual DSC; the pixel-unshuffled downsampling will be introduced in the second sub-section; at last, we will present the details of the HPUN.

Self-Residual DSC
DSC. Depthwise separable convolution (DSC) is composed by a depthwise layer and a pointwise layer as shown in Fig. 3(b). The depthwise layer uses single kernel for each input feature map. DSC is a popular lightweight module to reduce the redundant operations in the standard convolution. The conversion from the standard convolution to the DSC can be described as:

\[
F_{\text{out}} = C(F_{\text{in}}) \approx P(D(F_{\text{in}})),
\]

(1)

where \(F_{\text{out}}\) means the output features, \(C\) represents the standard convolution, \(F_{\text{in}}\) means the input features, \(D\) stands for the depthwise convolution, while \(P\) stands for the pointwise convolution. Depthwise convolution is the major part to process the spatial information of the input features, which needs far fewer parameters and computation costs than standard convolution with the same kernel settings.

Self-Residual DSC. It may have a significant side effect on the performance of SISR since SISR needs to enrich the information. The side effect is shown in the experiment section. To overcome the defects brought by the depthwise layer and keep its ability to process the spatial information, We find a balanced trade-off design by simply adding the input before the depthwise layer to the output of the depthwise layer as shown in Figure 3(c). The whole structure is described as:

\[
F_{\text{out}} = P(D(F_{\text{in}}) + F_{\text{in}}).
\]

(2)

Comparing Equation (1) and Equation (2), we can easily figure out that the outputs of the Self-Residual DSC have more similarity to the inputs than the outputs of the DSC. We will analyze the importance of the similarity in the experiment section. The self-residual does not introduce any additional parameters, and its computation costs can be ignored.

Pixel-Unshuffled Downsampling
Now we present details about the pixel-unshuffled downsampling, which is shown in Figure 4(a). As discussed in previous sections, low-frequency features can enhance the high-level representations (Sun et al. 2019; Wang et al. 2020). The work (Wang et al. 2020) investigates that repeating multi-resolution fusions can improve high-resolution representations for image segmentation tasks by leveraging low-resolution representations. However, previous SR

The conversion from the standard convolution to the DSC can be described as:

\[
F_{\text{out}} = C(F_{\text{in}}) \approx P(D(F_{\text{in}})),
\]

(1)

where \(F_{\text{out}}\) means the output features, \(C\) represents the standard convolution, \(F_{\text{in}}\) means the input features, \(D\) stands for the depthwise convolution, while \(P\) stands for the pointwise convolution. Depthwise convolution is the major part to process the spatial information of the input features, which needs far fewer parameters and computation costs than standard convolution with the same kernel settings.

Self-Residual DSC. It may have a significant side effect on the performance of SISR since SISR needs to enrich the information. The side effect is shown in the experiment section. To overcome the defects brought by the depthwise layer and keep its ability to process the spatial information, We find a balanced trade-off design by simply adding the input before the depthwise layer to the output of the depthwise layer as shown in Figure 3(c). The whole structure is described as:

\[
F_{\text{out}} = P(D(F_{\text{in}}) + F_{\text{in}}).
\]

(2)

Comparing Equation (1) and Equation (2), we can easily figure out that the outputs of the Self-Residual DSC have more similarity to the inputs than the outputs of the DSC. We will analyze the importance of the similarity in the experiment section. The self-residual does not introduce any additional parameters, and its computation costs can be ignored.
Figure 4: (a) Pixel-Unshuffled Downsampling (PUD). (b) Pixel-Unshuffled Block (PUB). (c) Hybrid Pixel-Unshuffled Block (HPUB). Note that the group number of the depthwise convolution in the PUD equals to the number of its inputs.

Figure 5: Our proposed HPUN. The architecture is based on EDSR, and the tail is from IMDN. The total number of HPUBs in HPUN is no less than 8.

To enhance the feature, we use an upsampler to project the low-frequency features to high dimension, and add them to the original input features. After that, a pointwise convolution is utilized for the communication among the channels. The process can be described as:

$$F^{\text{out}} = P(U(F^{\text{in}})) + L,$$

where $U$ stands for the upsampling function, $F^{\text{in}}$ means the input channels to the upsampler, and $L$ means the original input features. We use bi-linear upsample.

Hybrid Pixel-Unshuffled Network

Pixel-Unshuffled Block. After the exploration of the Self-Residual DSC and the pixel-unshuffled downsampling, we introduce the lightweight Pixel-Unshuffled Block (PUB). The PUB is composed of the Self-Residual DSC and the pixel-unshuffled downsampling. The detail of PUB is shown in Figure 4(b). The block can be represented as:

$$F^{\text{out}} = D(\sigma(PUD(F^{\text{in}}))) + \sigma(PUD(F^{\text{in}})) + F^{\text{in}},$$

where the PUB denotes the whole procedure of the pixel-unshuffled downsampling, and $\sigma$ represents the ReLU (Glorot, Bordes, and Bengio 2011).

Hybrid Pixel-Unshuffled Block. To further improve the performance on reconstruction, we integrate the standard convolution into our proposed PUB, which is named Hybrid Pixel-Unshuffled-Block (HPUB). The details of the HPUB are shown in Figure 4(c). The standard convolution layer’s kernel size is set to 3 to strike a balance between efficiency and performance. The kernel settings for the rest modules are the same as the PUB.

Hybrid Pixel-Unshuffled Network. We use HPUB to construct our Hybrid Pixel-Unshuffled Network (HPUN). The network is similar to EDSR. Since one HPUB has two residual blocks, we construct the body parts with 8 HPUB to align the settings in EDSR. To further reduce the parameters, we use the tail of IMDN. The details of the architecture are shown in Figure 5. We control the total number of HPUB for different sizes of the model. HPUN-M has 8 HPUN blocks in total, and HPUN-L has 12 HPUN blocks. The total number of HPUN-S is 8, but we replace one HPUB with two PUBs. The upsampler for the final high-resolution output is the pixel-shuffle module (Shi et al. 2016).

Experimental Results

Datasets and Metrics. As training data, we use the DIV2K dataset (Timofte et al. 2017) following the pop-
Table 1: Experiment results of different components generated with step schedule. The kernel size in PUD is set to 3 for fair comparison. PUB is the combination of PUD and Self-Residual DSC.

| Methods          | PSNR(Set5) | Params | Multi-Adds |
|------------------|------------|--------|------------|
| P-Conv           | 37.36      | 181K   | 40.7G      |
| DSC              | 37.65      | 207K   | 44.9G      |
| Self-Residual DSC| 37.73      | 207K   | 44.9G      |
| PUD              | 35.68      | 267K   | 44.9G      |
| PUB              | 37.76      | 226K   | 44.9G      |

Figure 6: Results of different combinations of up- and down-scale operations conducted with step scheduler.

Table 2: The experiment results of different combination of pooling layers and upsamplers. The results are generated with cosine scheduler on Set5 with $\times 2$ scale.

| Operation          | Nearest | Bi-linear | Max-pooling | Avg-pooling | PSNR  |
|--------------------|---------|-----------|-------------|-------------|-------|
| stride-2 Dconv     | $\times$ | $\checkmark$ | $\times$ | $\checkmark$ | 37.83 |
| stride-2 Maxpool   | $\times$ | $\times$ | $\checkmark$ | $\checkmark$ | 37.82 |
| stride-2 Avgpool   | $\times$ | $\times$ | $\times$ | $\times$ | 37.87 |

Effectiveness of the pixel-unshuffled downsampling. We have run six experiments to target the best combination of the pooling layer and the upsample. The results are presented in Table 2. From the table, we can observe that the model with max-pooling layer and bi-linear upsampling can achieve the best performance among all combinations.

We also compare the performance of the pixel-unshuffled downsampling with other kinds of downsampling operations. The results are shown in Figure 6. We compare the PUD module, depthwise convolution with 2 strides, max pooling, and average pooling with both bi-linear and nearest upsampling. As shown in the figure, the pixel-unshuffled downsampling is far beyond others.

Besides, we construct a network with only pixel-unshuffled downsampling and compare its performance with the networks constructed by the baseline DSC, the Self-Residual DSC, and the PUB. The results are shown in Table 1. The network constructed with only pixel-unshuffled downsampling performs far worse than the baseline DSC. However, the network of PUB can perform better than the Self-Residual DSC. Specifically, the network constructed by PUB achieves 0.03dB higher than the Self-Residual DSC only. The results show that the pixel-unshuffled downsampling can enhance the performance, but it only works together with other spatial operations.

Intuition. For the further exploration, we generate the heatmap features using the Normalized Mean Error (NME) among their shallow features and deep features. The NME can be described as $\text{NME} = \frac{1}{N} \left\| F^S - F^D \right\|_F$, where $N$ means the total number of the elements in the features, $F^S$ means the output features from the head block, $F^D$ means the output features from the body block, and $\left\| \cdot \right\|_F$ denotes...
Table 3: Benchmark results with BI degradation. We show average PSNR & SSIM values for scaling factor \( \times 4 \). We highlight the top-2 least params, Multi-Adds, and performance on each datasets with bold text (best) and underlined text (second), respectively. + denotes the results are generated with self-ensemble.

![Figure 7: Charts of the relationship between PSNR and the NME. The chart is generated on the whole Set5 dataset with \( \times 2 \) scale. Abbrevs.: PUB=Pixel-Unshuffled Downsampling, SRDSC=Self-Residual DSC.](image)

Specifically, the NME of the pointwise structured network is much lower than the DSC baseline network, which means the main change of the features is brought by the depthwise convolution. The PSNR results of the pointwise structured network and pixel-unshuffled downsampled structured net-
work show that limited changes will decrease the performance. However, the DSC network has large NME but limited PSNR, which means the larger NME does not represent the better performance. Although the chart generated from the bird indicate that there exist some variance, the conclusion still stands. We also visualize the NME result among the head features and body features of these networks, respectively. The visualization results are shown in Figure 8.

Thus, we can conclude that the PUD can significantly reduce the NME among the shallow features and deep features. Adjusting the number of the modules will improve the performance of the architecture. Besides, we find out that the NME among the head and body features gets smaller when we integrate the standard convolution into the PUB comparing the heatmap features of HPUN-S and HPUN-M. Considering the NME of the pointwise structured network, it is natural to think about the communication among the features can also help to learn the similarity among the features.

From the scatter figure of mean results, we notice that the performance increases explosively with the increase of the NME at first. Then the performance starts to drop after the NME surpasses the value around 0.007. Besides, the NME gets smaller by increasing the number of pixel-unshuffled downsamplers. Therefore, we can conclude that there may exist an optimal NME value, and increasing pixel-unshuffled downsamplers or adding residuals will reduce the NME of the network towards the optimal. The intuition can help us to design the network structure or apply the pruning strategy on SISR. However, we notice that the optimal NME is variable with the inputs. We still need more experiments to validate the conclusion for different tail structures and datasets.

Comparison Results

To simulate LR images in SR settings, the BI degradation model is commonly used. For the BI degradation model, we compare our HPUN network to 13 cutting-edge SR methods: SRCNN (Dong et al. 2016), VDSR (Kim, Kwon Lee, and Mu Lee 2016a), DRCN (Kim, Kwon Lee, and Mu Lee 2016b), DRRN (Tai, Yang, and Liu 2017), LapSRN (Lai et al. 2017), MemNet (Tai et al. 2017), CARN (Ahn, Kang, and Sohn 2018), IDN (Hui, Wang, and Gao 2018), SRFBN-S (Li et al. 2019), IMDN (Hui et al. 2019), LatticeNet (Luo et al. 2020), SMSR (Wang et al. 2021), and LatticeNet-CL (Luo et al. 2022). All of them are popular lightweight SR methods. Note that based on the work of LatticeNet-CL (Luo et al. 2022), the result in LatticeNet (Luo et al. 2020) is generated with self-ensemble. Thus in Table 3 we report the results cited from LatticeNet-CL (Luo et al. 2022) for fair comparison.

Quantitative Results. Quantitative results are shown in Table 3 for ×4 SR. Among all methods, our HPUN-L achieves the new SOTA performance. Its performance on Set14, B100, and Urban100 is even better than the LatticeNet-CL+, which is generated with self-ensemble. With the self-ensemble technique, our HPUN-L achieves around 0.11dB in average better than the LatticeNet-CL+. Our HPUN-M can achieve top-3 performance on Set5, Set14, B100 and Manga109 with self-ensemble results excluded. Besides, compared with other competitive methods such as IMDN, LatticeNet, and SMSR, it only has two thirds or even fewer parameters and Multi-Adders. Furthermore, we can significantly improve the HPUN-M network using the self-ensemble technique. Our HPUN-S achieves comparable performance with the second least parameters among all the methods, and takes the minor computation costs.

Visualization Results. Visualization results are shown in Figure 9. The results are generated with ×4 scale on Urban100 and Manga109. Compared with other methods, our HPUN-L network generates better reconstruction results, especially on Manga109. As shown in the figure, the results of our HPUN-L network have fewer artifacts.

Conclusions

In this paper, we proposed a lightweight network named Hybrid Pixel-Unshuffled Network (HPUN) for image SR. Specifically, we design the Self-Residual Depthwise Separable Convolution to overcome the defects of the depthwise convolution, and the pixel-unshuffled downsampling to enhance the performance with low-frequency representations. Both proposed modules take limited computation costs and parameters. With the two proposed modules, we design a lightweight block named Hybrid Pixel-Unshuffled Block with the standard convolution layer and the proposed Pixel-Unshuffled Block. The HPUN can achieve new SOTA performance with limited parameters and Multi-Adders. We also discover and discuss the relationship between the PSNR and the NME among the shallow features and deep features. We believe that the phenomenon should be general, and we can take advantage of it for network design.
Sun, K.; Xiao, B.; Liu, D.; and Wang, J. 2019. Deep high-resolution representation learning for human pose estimation. In CVPR.

Tai, Y.; Yang, J.; and Liu, X. 2017. Image Super-Resolution via Deep Recursive Residual Network. In CVPR.

Tai, Y.; Yang, J.; Liu, X.; and Xu, C. 2017. MemNet: A Persistent Memory Network for Image Restoration. In ICCV.

Tan, M.; and Le, Q. 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. In ICML.

Timofte, R.; Agustsson, E.; Van Gool, L.; Yang, M.-H.; Zhang, L.; Lim, B.; Son, S.; Kim, H.; Nah, S.; Lee, K. M.; et al. 2017. Ntire 2017 challenge on single image super-resolution: Methods and results. In CVPR.

Tong, T.; Li, G.; Liu, X.; and Gao, Q. 2017. Image Super-Resolution Using Dense Skip Connections. In ICCV.

Wang, J.; Sun, K.; Cheng, T.; Jiang, B.; Deng, C.; Zhao, Y.; Liu, D.; Mu, Y.; Tan, M.; Wang, X.; et al. 2020. Deep high-resolution representation learning for visual recognition. IEEE transactions on pattern analysis and machine intelligence, 43(10): 3349–3364.

Wang, L.; Dong, X.; Wang, Y.; Ying, X.; Lin, Z.; An, W.; and Guo, Y. 2021. Exploring Sparsity in Image Super-Resolution for Efficient Inference. In CVPR.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. TIP.

Wu, B.; Dai, X.; Zhang, P.; Wang, Y.; Sun, F.; Wu, Y.; Tian, Y.; Vajda, P.; Jia, Y.; and Keutzer, K. 2019. Fbnet: Hardware-aware efficient convnet design via differentiable neural architecture search. In CVPR.

Yin, Y.; Robinson, J. P.; Jiang, S.; Bai, Y.; Qin, C.; and Fu, Y. 2021. SuperFront: From Low-resolution to High-resolution Frontal Face Synthesis. In ACMMM.

Yin, Y.; Robinson, J. P.; Zhang, Y.; and Fu, Y. 2020. Joint super-resolution and alignment of tiny faces. In AAAI.

Zeyde, R.; Elad, M.; and Protter, M. 2010. On single image scale-up using sparse-representations. In Proc. 7th Int. Conf. Curves Surf.

Zhang, H.; and Patel, V. M. 2018. Densely Connected Pyramid Dehazing Network. In CVPR.

Zhang, X.; Zhou, X.; Lin, M.; and Sun, J. 2018a. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. In CVPR.

Zhang, Y.; Li, K.; Li, K.; Wang, L.; Zhong, B.; and Fu, Y. 2018b. Image super-resolution using very deep residual channel attention networks. In ECCV.

Zhang, Y.; Li, K.; Li, K.; Zhong, B.; and Fu, Y. 2019. Residual Non-local Attention Networks for Image Restoration. In ICLR.

Zoph, B.; Vasudevan, V.; Shlens, J.; and Le, Q. V. 2018. Learning transferable architectures for scalable image recognition. In CVPR.

Zou, W. W.; and Yuen, P. C. 2012. Very low resolution face recognition problem. TIP.