Towards Real-Time 4K Image Super-Resolution

Eduard Zamfir*, Marcos V. Conde*, Radu Timofte
Computer Vision Lab, CAIDAS, IFI, University of Würzburg, Germany
{eduard-sebastian.zamfir, marcos.conde, radu.timofte}@uni-wuerzburg.de

Figure 1. We introduce a new baseline termed *RT4KSR* for upscaling images from 720p and 1080p input to 4K in real-time at $>$ 60FPS.

Abstract

Over the past few years, high-definition videos and images in 720p (HD), 1080p (FHD), and 4K (UHD) resolution have become standard. While higher resolutions offer improved visual quality for users, they pose a significant challenge for super-resolution networks to achieve real-time performance on commercial GPUs. This paper presents a comprehensive analysis of super-resolution model designs and techniques aimed at efficiently upscaling images from 720p and 1080p resolutions to 4K. We begin with a simple, effective baseline architecture and gradually modify its design by focusing on extracting important high-frequency details efficiently. This allows us to subsequently downscale the resolution of deep feature maps, reducing the overall computational footprint, while maintaining high reconstruction fidelity. We enhance our method by incorporating pixel-unshuffling, a simplified and speed-up reinterpretation of the basic block proposed by NAFNet, along with structural re-parameterization. We assess the performance of the fastest version of our method in the new NTIRE 2023 Real-Time 4K Super-Resolution challenge and demonstrate its potential in comparison with state-of-the-art efficient super-resolution models when scaled up. Our method was tested successfully on high-quality content from photography, digital art, and gaming content.

*Corresponding authors. Code and models are open-sourced at: https://github.com/eduardzamfir/RT4KSR

1. Introduction

In image super-resolution (SR), one deals with the ill-posed problem of recovering the high-resolution (HR) counterpart of a previously down-sampled and possibly further degraded low-resolution (LR) source image. Previous research has introduced several classical methods for single image SR, as documented in [5, 6, 17, 42, 46–48]. Nonetheless, with the emergence of Deep Learning, research on single image SR rapidly shifted towards deep learning-based approaches [7, 13, 30, 35, 36, 51, 58, 62]. While much progress in restoration performance has been achieved through larger and deeper networks, these improvements have incurred a higher demand for time and computational resources, necessitating more efficient and lightweight solutions. As media streaming platforms have achieved overwhelming success, and the amount of image and video content created and shared online has become practically inexhaustible, the need for stable and high-bandwidth internet connections has significantly increased. However, the media industry has adopted the practice of compressing its content before transmission and reconstructing it to its full resolution at the consumer's end. Efficient and lightweight super-resolution methods have become increasingly important as a result.

To further the development of efficient and fast SR methods, in conjunction with the *NTIRE 2023 Real-Time 4K SR* challenge [10] we investigate previous SR concepts in the context of upscaling diverse types of content, including digital art and photography, to ultra-high resolution. Following

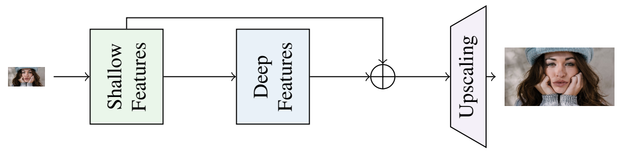


Figure 2. *Blueprint of modern SR methods.* Most CNN-based SR methods have three-part structure in common, consisting of shallow feature extraction, computation on deep features and a final up-scaling module [1, 10, 13, 31, 35, 64].

this study, we propose a fast and lightweight model tailored for 4K image SR from large input resolutions (720p, 1080p $\rightarrow$ 4K). Moreover, our method showcases its scalability and achieves comparable results on established benchmarks while surpassing them in terms of runtime and efficiency. Additionally, we explore the significance of training data when presented with the challenge of enhancing computer-generated visuals along with photorealistic content.

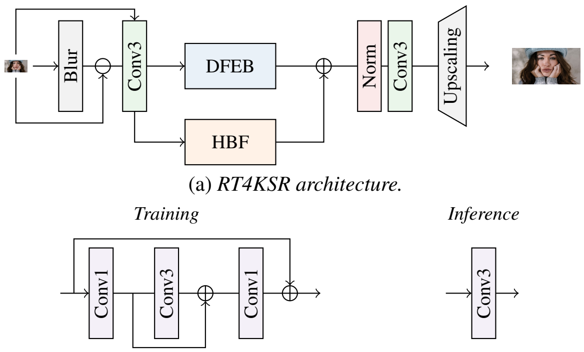
Starting with the basic blueprint shown in Fig. 2 for learning-based super-resolution approaches, we gradually modify a simple and shallow network architecture to improve its performance in terms of runtime and reconstruction fidelity. Drawing concepts from the image compression research [53], the key aspect of our approach is downsizing the deep features to accelerate the computation, while simultaneously retaining valuable high-frequency (HF) information from the LR input. Therefore, we efficiently extract high-frequency details first before downsampling the feature. To ensure effective computation of deep features, we utilize the potent NAFNet [7] block. Additionally, we simplify the design of its basic components and apply structural re-parameterization [12] at inference time to further reduce the total runtime of our method. After extracting high-frequency details, we refine them through a dedicated parallel branch before reintroducing them to the deep features. This compensates for the previously performed downsampling operation. We provide exhaustive ablation studies using the novel 4K RTSR [10] benchmark as a reference.

2. Related Work

Efficient Architectures. In recent years, achieving near real-time SR on resource-constrained platforms has gained popularity [24, 33, 34, 57]. As a result, researchers have proposed optimized neural architectures [23], network compression methods, and training strategies to address the need for efficient solutions [2, 14, 29, 44].

IMDN [22] introduces a lightweight information multi-distillation network that employs cascaded blocks to extract hierarchical features using an information distillation mechanism (IDM). RFDN [37] refines the architecture of IMDN [22] by proposing the residual feature distillation network, which replaces IDM with feature distillation connections. ECBSR [61] introduces an edge-oriented convolutional block that utilizes structural re-parameterization [12] to enhance the learning capability of the model without impacting the inference time. While accessing preceding network layers can be compute-intensive, sequential operations can minimize memory consumption and runtime overhead. RLFN [28] leverages this idea to achieve high reconstruction accuracy through the use of simple 3×3 convolutions instead of concatenation and feature distillation layers, as well as a multi-stage training strategy. Similarly, FMEN [16] employs a lightweight backbone by stacking multiple optimized convolutions and reducing compute through re-parameterization at inference time. ESRT [38] combines a lightweight CNN to dynamically adjust the feature map size, allowing for the extraction of deep features with low computational cost, with a lightweight Transformer [15, 49] to capture long-term dependencies between similar patches. VapSR [65] introduces large receptive field design with depth-wise convolutions into the attention mechanism and presents a novel pixel normalization approach for improved training stability. Furthermore, the Mobile AI workshop in 2022 [24] highlighted the challenge of achieving efficient and accurate quantization for image super-resolution on edge devices. To address this issue, most of the methods proposed at the workshop utilized a shallow CNN architecture and re-parameterization techniques to reduce inference time while maintaining competitive restoration performance. NAFNet [7] presents a highly efficient approach for image restoration by simplifying commonly used architectural components, *i.e.* removing nonlinearities, outperforming previous techniques across a wide range of image restoration problems.

Upscaling to Ultra-High Definition. The field of super-resolving images or videos to achieve ultra-high resolutions, such as 4K and 8K, remains relatively unexplored in the research community. While modern display technologies can handle ultra-high definition (UHD) content, effective broadcasting and streaming require significant bandwidth. As a result, the industry standard involves downscaling prior to data transfer and upscaling back to full resolution on the consumer's end. This process demands highly efficient super-resolution (SR) approaches [8, 11, 26]. Moreover, cloud-based gaming experiences a large gain in popularity, where upscaling digital content presents additional challenges, *e.g.* aliasing, consequently requiring tailored approaches [52, 54]. The upscaling of images to 4K resolution in real-time remains a relatively unexplored topic within the broad research community of SR. The NTIRE 2023 4K RTSR challenge [10] addresses this open question by demanding lightweight yet effective SR solutions from its participants. It also provides them with a competitive benchmark for 4K image SR. Moreover, to the best of our



(a) *RT4KSR architecture.*

(b) *Reparameterizable Residual Block [16].*

Figure 3. *Overview of proposed RT4KSR architecture.* (a) Initially, structural high-frequencies (HF) are extracted from the LR input, followed by a shared convolution that extracts both HF and LR features. These features are then separately processed by the DFEB and HFB modules. The HFB module is responsible for enhancing the HF components before enriching the output of the DFEB module, preceding the upscaling stage. (b) During training, channels C are first expanded by $2C$ and refined before squeezed back to C. During inference, RRB is a standard 3×3 convolution with C channels thanks to reparameterization [16].

knowledge, Zhang et al. [59] offer a comprehensive dataset for evaluating recent SR techniques on upscaling to 4K and 8K resolution. However, this dataset does not consider the increased model runtime and has limited content.

3. Method

In this section, we first revisit the blueprint approach for SR based on deep models and introduce our proposed architecture in Sec. 3.1. Next, in Sec. 3.2 we describe the training schedule to enhance the performance of our method in the NTIRE 2023 4K RTSR challenge [10].

3.1. Model Architecture

Over the past few years, the SR research community has developed a common structure for CNN-based neural architectures. The majority of methods follow a three-part blueprint shown in Fig. 2 that includes an initial extraction of shallow features, a computationally intensive refinement in deep feature space, and a final upscaling stage to achieve the target resolution. Moreover, previous works [20, 36, 62, 63] have shown the tremendous benefit of adding local and global residual connections. The methods proposed in [24] effectively utilize this blueprint to achieve strong reconstruction capabilities while maintaining a low computational footprint. In our work we adopt this aforementioned structure and prioritize real-time inference with a focus on a shallow and lightweight design. We begin with a simple stack of 3×3 convolutions, each followed by a GeLU [19] non-linearity visualized in Fig. 5a. Contrary to prior work [24, 31, 56], we address SR from large-scale inputs, which poses an additional layer of complexity for real-time processing. In Deep Learning, a common approach is to reduce the spatial resolution of feature maps to keep the computational burden low. However, it has been demonstrated that decreasing spatial resolution within the network can negatively impact the reconstruction performance of SR methods, since high-frequency (HF) details are already scarce in the LR input image. Yet, the atypical large size of LR inputs in our use-case allows us to effectively process the HF information differently.

An overview of our final architecture design is presented in Fig. 3. First, we efficiently extract the HF components from the LR input. Subsequently, the LR input and extracted HF maps undergo processing via a shared convolution for shallow feature extraction. Next, we enhance the HF features in a dedicated high frequency branch (HFB) while simultaneously compressing the features in the deep feature extraction branch (DFEB) of the network. Lastly, we inject the enhanced HF components back into the deep features. We add a LayerNorm [3] and another convolution before upscaling to the desired output resolution using PixelShuffle [44]. In particular LayerNorm provides consistent improvement and stable training, becoming a standard in image restoration [7, 9]. Next, we gradually modify the basic structure to enhance the network capabilities while aiming at keeping computational costs low.

Enhancing High Frequencies. Inspired by [38, 40], we aim at efficiently extracting and enhancing the remaining HF details in the LR input. To achieve this goal, we explore two straightforward approaches that are both rapid and do not introduce additional complexity that could impede the processing speed of our method. *(i)* We reduce the size of the LR input through average pooling, immediately followed by upscaling it back to the original resolution using Nearest Neighbor interpolation. *(ii)* We use an inexpensive Gaussian blur operation on the input to obtain its blurred version. Both approaches yield an image that represents the signal's uniformity, which we then subtract from the initial LR input to obtain the HF components.

In Figure 4, we conduct a visual comparison between discussed approaches where the Down-and-Up operation falls short in extracting fine-grained details, while utilizing the Gaussian Blur aids in extracting circular contours. Tab. 2a quantifies the impact of both approaches on our method. The HF details are then further refined by a shallow parallel branch using a 3×3 convolution and GeLU activation before being injected back into the deep features. Tab. 1 presents a direct comparison between the baseline and its variant with the HFB. Although both modifications depicted in Figs. 5a and 5b exhibit improved PSNR and SSIM, they also entail longer runtime.

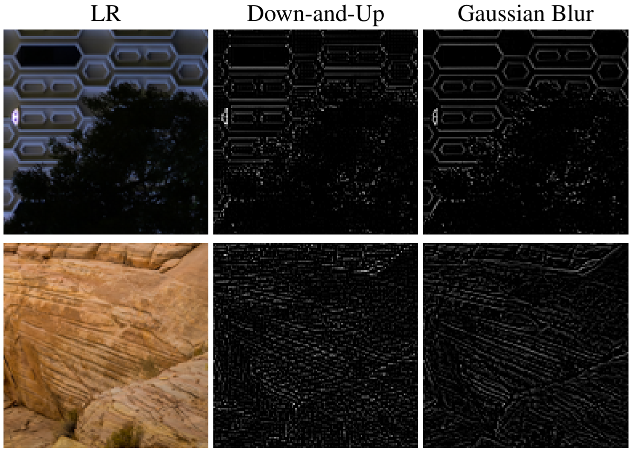
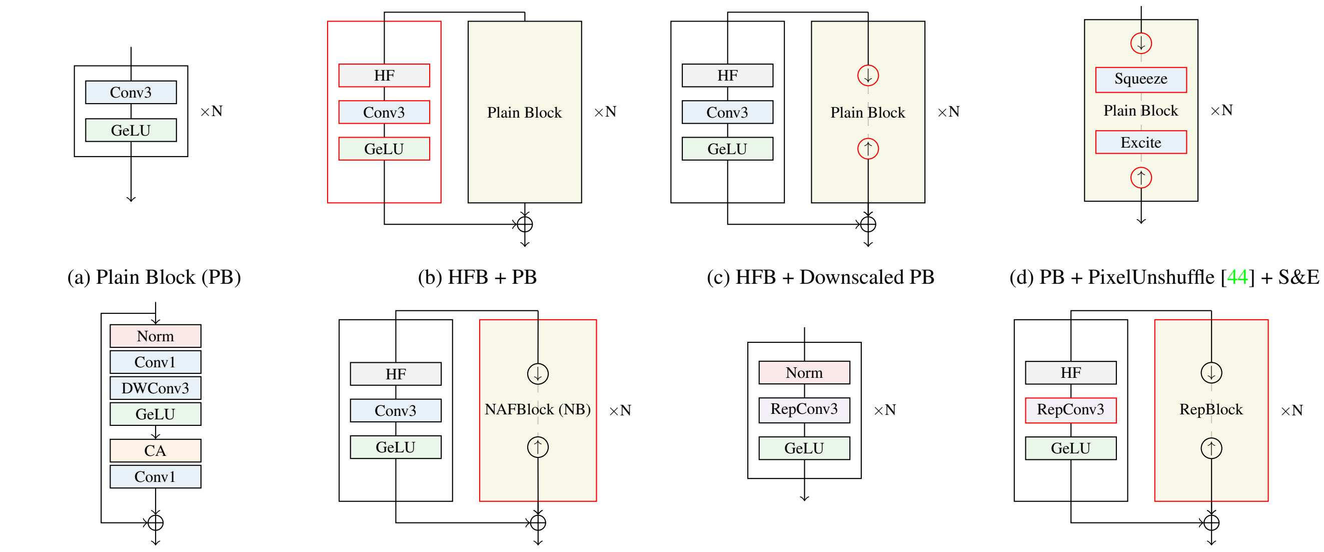


Figure 4. *Extracting high frequency (HF) information.* Applying the Down-and-Up generates speckled information with higher intensities, whereas employing Gaussian Blur yields more intricate details, particularly in circular regions.

Compressing Deep Features. In practical applications, the runtime of SR methods can be decreased by reducing the network's depth or width. Nonetheless, this often results in inferior restoration performance. In addition to achieving the optimal balance between depth and width, downsizing the spatial dimensions provides another means of lowering the runtime. However, this usually results in the loss of valuable details, which is, in fact, crucial for SR tasks that aim to recover previously lost information. Earlier studies [37, 38] employed strided convolutions or pooling operations to attain the required spatial resolution. When we apply the Down-and-Up scheme to the DFEB of our method, as shown in Tab. 1, we observe a significant loss of reconstruction fidelity due to pooling. Intriguingly, the improvements obtained by incorporating the HFB into our method are not as prominent when the deep features are downscaled. To address this issue, we explore the use of the PixelUnshuffle [44] operation for feature downsampling, in addition to the initial extraction of HF components. PixelUnshuffle [44] reduces the spatial dimensions by a factor of s, while increasing the channel dimension by a factor of s^2. Although incorporating PixelUnshuffle naively into our architecture significantly reduces its efficiency, it does improve the reconstruction accuracy, as shown in Tab. 1 and the visualization in Fig. 5c. An architecture that employs the unshuffling may benefit from a larger channel dimension. However, to compensate for the loss of inference efficiency, we investigate the possibility of squeezing the channel dimensions after the unshuffling process and then mapping them back after the output of the DFEB. While this approach can reduce the runtime, the model is unable to recover the loss of information resulting from channel reduction, see Tab. 1. To address this issue, we restructure the 3×3 convolution used for extracting shallow features so that it occurs after the LR input has been unshuffled, see Fig. 5d. Both HF and LR features undergo additional processing through the DFEB and HFB modules. Prior to the upscaling stage, we merge the refined high-frequency components with the deep features, and then increase the output resolution by $\times 4$ (for $\times 2$ SR). Next, we detail the enhancements made to both the DFEB and HFB modules in order to improve their modeling capacity leading to the final design of our model.

Increasing Block Complexity. The architecture design at this point has limitations in terms of the expressiveness of its features as the basic component of the DFEB is a standard 3×3 convolution and GeLU activation repeated N times, see Fig. 5a. Recently, NAFNet [7] has shown strong reconstruction capabilities with a more complex block design. We enhance our plain block by incorporating components from the basic block of NAFNet visualized in Fig. 5e. Specifically, we replace the 3×3 convolution with an inverted depthwise separable convolution, which is followed by GeLU non-linearity to increase the feature dimensions from C to $2C$. The basic block also includes Channel Attention [7, 50, 62], LayerNorm [3], and a local skip connection. To expedite the performance of this block design, we substitute the standard Channel Attention with its efficient version [50]. A final 1×1 convolution maps the feature dimensions back to C. As anticipated, incorporating the NAFNet-inspired block in Fig. 5f enhances the modeling capacity of our method, see Tab. 1. However, this also results in a substantial increase in runtime. Although we can address this issue by downsampling the deep features, we encounter challenges in maintaining the accuracy gains achieved by the more intricate block design, see Tab. 1. Unfortunately, this renders the modified architecture impractical for our use case. In the following section, we will outline our approach to streamline the network design while preserving both high inference speeds and reconstruction fidelity.

Model reparameterization. First introduced in [12], structural reparameterization has rapidly gained traction within the research community as a means of reducing model runtime during inference. Many participants of the NTIRE and AIM efficiency challenges [24, 33] have adopted various forms of reparameterization for their architectures. We closely follow [16] and replace the depthwise separable convolutions within the DFEB with a reparameterizable residual block (RRB), *cf.* Fig. 3b. The RRB expands the channel dimension C by a factor of $f_{Exp} = 2$ with a 1×1 convolution. Next, a 3×3 convolution enhances the learned features in a higher dimensional space, followed by a final 1×1 convolution that compresses the



(a) Plain Block (PB) (b) HFB + PB (c) HFB + Downscaled PB (d) PB + PixelUnshuffle [44] + S&E

(e) NAFNet [7] Basic Block (NB) (f) HFB + NB (g) Reparameterized Block (h) HFB + Rep. NB

Figure 5. *Network configurations.* We start from a shallow CNN (Fig. 5a) and gradually increase its complexity. By downscaling or unshuffling the feature maps (Figs. 5c and 5d) and inserting reparameterizable (Fig. 5g) convolutions we reduce the inference time. *CA*, *Norm* and *HF* denote channel attention [50], LayerNorm [3] and HF extraction, respectively. Modifications are highlighted in red.

channels back to C, retaining only the most discriminative features. Short and long residual connections facilitate feature propagation. During inference, we can summarize the RRB using a single 3×3 convolution. This reduces processing time while still preserving the expressive power of higher feature channels. Additionally, we eliminate the efficient Channel Attention module, final 1×1 convolution, and local residual, retaining only the normalization operation preceding the RRB. Our new reparameterizable basic block is visualized in Fig. 5g. Furthermore, we enhance the model capacity of the HFB by substituting the 3×3 convolution with the RRB, see Fig. 5h.

3.2. Towards Learning the High Frequency Details

Besides traditional pixel-wise reconstruction loss functions, the Computer Vision community proposed several *perceptual* losses [18, 25, 40, 60] to improve SR models and impose meaningful priors during model training. Explicitly modeling the high frequencies is a key concept of our model. Therefore, we extract high frequency information, *e.g.* edges and contours, from the SR output and HR target image using the same Gaussian blur operation as within our model. As an auxiliary optimization task, we minimize the L1 distance between obtained HF maps. The loss is formulated as follows:

$$\mathcal{L}_{HF} = \|(y - (y * b)) - (\hat{y} - (\hat{y} * b))\|_1 \tag{1}$$

We incorporate this auxiliary loss solely to enhance the performance of our model in the NTIRE 4K RTSR challenge [10]. Typically, participants develop increasingly complex training strategies to improve the performance of their methods in this challenge.

4. Experiments

4.1. Experimental Setup

Datasets and Metrics. Our training dataset is a combination of 800 images from DIV2K [1], 2650 images from Flickr2K, and 1000 images from LSDIR [32]. Following standard practice, we report PSNR and SSIM metrics on RGB. To explore the real-time performance of our method on large-scale inputs, we conduct most of our experiments on the new benchmark proposed in the NTIRE 2023 4K RTSR challenge [10]. Additionally, we evaluate our approach on canonical SR benchmarks, namely Set5 [4], Set14 [55], Urban100 [21] and BSD100 [41], when comparing our results to previously published work.

Training Details. We extract random crops of size 128×128 from the RGB training set and further augment the crops by random rotation, horizontal and vertical flipping. LR images are generated online using bicubic downsampling of the original HR images. We use ADAM [27] optimizer to minimize the $\mathcal{L}1$ 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 and decay factor 0.5.

Runtime evaluation. Unlike other studies [33], we assess the runtime of our proposed architectures by repetitively

Table 1. *Results on the NTIRE 2023 RTSR4K Benchmark.* The runtimes are computed using Nvidia RTX 3090. For better comparison we color-code the runtime using < 24 FPS, $30 > x > 24$ FPS, $60 > x > 30$ FPS, $120 > x > 60$ FPS and > 120 FPS, respectively.

Scale	Method	# Params	FLOPs (G)	PSNR (dB,↑) *RGB*	SSIM (↑) *RGB*	Runtime (ms, ↓) RTX 3090
	Bicubic	-	-	33.92	0.8829	0.46
	(a) Baseline	35.2K	429.98	34.11	0.8834	11.70
	(b) Baseline + High Frequency Branch (HFB)	34.6K	515.98	34.13	0.8836	14.44
	(-) Baseline + Down-and-Up	35.2K	200.66	34.01	0.8830	07.79
	(c) Baseline + HFB + Down-and-Up	35.2K	286.65	34.00	0.8827	10.48
×2	(-) Baseline + PixelUnshuffle	346.6K	1347.28	34.15	0.8835	16.46
	(d) Baseline + PixelUnshuffle + Squeeze-and-Excite Channels	40.0K	217.65	34.01	0.8829	8.42
	(e) NAFNet Basic Block	30.0K	353.54	34.16	0.8844	40.71
	(-) NAFNet Basic Block + HFB	30.6K	439.54	34.17	0.8846	43.40
	(-) NAFNet Basic Block + Down-and-Up	30.0K	181.55	34.00	0.8829	14.92
	(f) NAFNet Basic Block + HFB + Down-and-Up	30.6K	267.54	34.01	0.8828	17.61
	(h) Rep. Basic Block + HFB + PixelUnshuffle (final)	44.5K	**171.99**	**34.20**	**0.8848**	**7.09**
	Bicubic	-	-	31.31	0.8251	0.44
×3	(a) Baseline	38.5K	477.76	31.43	0.8252	5.57
	(h) Rep. Basic Block + HFB + PixelUnshuffle (final)	57.5K	**219.77**	**31.72**	**0.8297**	**3.74**

passing a randomly initialized tensor through the network for $n = 244$ iterations. This approach enables us to avoid incorporating the costly data loading process to GPU memory, which could impair the actual inference speed of the evaluated model architectures. As proposed in [10], we measure the runtime using mixed-precision.

4.2. Ablation Studies

In this section, we analyze the correlation between fidelity improvement and runtime consumption for different network aspects. Our studies are conducted for ×2 SR on the 4K RTSR [10] benchmark.

Non-learnable HF extraction. As mentioned in Sec. 3.1, we explore two widely-used and efficient approaches to extract HF details from images. While learning-based techniques have shown significant advancements over traditional hand-crafted methods, the overall efficiency of the model is crucial for our use case. This ablation study aims to determine which approach is the most effective in extracting valuable HF information to compensate for the feature downscaling inside the DFEB. The findings are showcased in Tab. 2a, indicating that the application of Gaussian blur not only results in more visually meaningful information but also improves the quantitative performance. As a result, we incorporate the Gaussian blur approach for extracting HF components into our final method.

Simplifying NAFNet's basic block. In Sec. 3.1, we explained our decision to use the basic block proposed by NAFNet [7] as a starting point and detailed the modifications we made to arrive at our final version. In this experiment, we aim to investigate the impact of each modification on both the total runtime and the reconstruction performance of our approach. Our findings, presented in Tab. 2b using the 4K RTSR [10] benchmark, reveal that including normalization results in a significant increase in

Table 2. *Model architecture.* We present the PSNR and SSIM results of the S-variant of our method on the full RGB test samples of the 4K RTSR benchmark [10].

(a) *Comparison of HF extraction.*

Scale	Method	PSNR (dB,↑)	SSIM (↑)
×2	RT4KSR-S + Down-and-Up	34.17	0.8844
	RT4KSR-S + Gaussian Blur	**34.20**	**0.8848**
×3	RT4KSR-S + Down-and-Up	31.70	0.8295
	RT4KSR-S + Gaussian Blur	**31.72**	**0.8297**

(b) *Ablation on the basic block.*

Scale	Method	Runtime (ms,↓)	PSNR (dB,↑)	SSIM (↑)
	Plain	**05.19**	34.16	0.8842
×2	Residual	05.54	34.15	0.8841
	LayerNorm [3]	07.09	**34.20**	**0.8848**
	Plain	**02.83**	31.66	0.8285
×3	Residual	02.98	31.65	0.8283
	LayerNorm [3]	03.74	**31.72**	**0.8297**

(c) ×2 *and* ×3 *results in the NTIRE 2023 4K RTSR challenge [10].*

Method	Runtime (ms,↓)		Score (↑)		PSNR (dB,↑)		SSIM (↑)	
S + aux. Loss	7.09	3.74	9.27	14.01	34.22	31.74	.8854	.8299

runtime, but yields the best reconstruction accuracy in terms of PSNR and SSIM. This result holds consistently across both upscaling scenarios from 720p ($\times 3$) and 1080p ($\times 2$) to 4K resolution. Our approach offers a practical trade-off between runtime and accuracy, besides adjusting the network's depth or width. For our final contribution, we decide to incorporate the LayerNorm [3] operation for improved reconstruction fidelity.

Training on diverse contents. The novel 4K RTSR [10] features HR high-quality images from a variety of sources. Consequently, we enhance our training data with various sources of content, such as GTA5 [43] and LSDIR [32], in addition to the conventional DIV2K [1] and Flickr2K [45] datasets. To keep the dataset size reasonable, we include not more than 2500 images from GTA5 [43] and 1000 images from LSDIR [32]. In Tab. 3a, we present the performance results of our S-variant model trained on various dataset configurations. We find that by including a subset of LSDIR [32] in our training data, we observe a marginal improvement in performance compared to training solely on DIF2K [1, 45]. We experiment the same behaviour with the inclusion of a random subset of 1000 images of gaming content from the GTA5 [43]. We attribute this to (i) the fact that unlike photorealistic datasets, GTA5 [43] does not offer high-resolution images exceeding [1914×1052], (ii) the constrained model complexity -which acts as self-regularization- does not allow to exploit the variety and abundance of data [32] during training.

Increasing the model complexity. This ablation study aims to explore the scalability of our method by increasing the model complexity, with the trade-off of longer runtime for improved model performance. We enhance our model by increasing the number of blocks B and channels C. During inference, we report the runtime and number of channels of the reparameterized model. However, at training time, the number of channels within the RRB doubles. As illustrated in Tab. 3b, our findings reveal that by considering reduced runtime, we can significantly enhance the reconstruction performance of our method in terms of PSNR and SSIM, with the *XL*-variant delivering the best results. Nonetheless, our current shallow architecture demonstrates limitations in simply increasing its size, indicating that more sophisticated approaches must be employed to effectively benefit from a larger model complexity.

NTIRE 2023 4K RTSR challenge. In Tab. 2c, we present the results of our *S*-sized variant for both tracks of the challenge [10]. In addition to the standard SR metrics and the runtime per image, the participating teams are evaluated and ranked by the *score* function described in [10]. Unlike other proposed solutions, we do not employ multiple training stages and extensive hyperparameter search. Our primary objective in this study is to provide a detailed account of how to develop a competitive baseline for 4K real-time SR while examining various architectural design choices.

Table 3. *Training data and model complexity.* We evaluate all method variants on 4K RTSR [10] benchmark using PSNR and SSIM (RGB). The S-variant is used for data ablation.

(a) *Ablation on the training data.*

Scale	Dataset	PSNR (dB,↑)	SSIM (↑)
$\times 2$	DIV2K + Flickr2K (DIF2K)	34.18	0.8844
	DIF2K + LSDIR	34.20	0.8848
	DIF2K + GTA5	34.20	0.8850
	DIF2K + LSDIR + GTA5	**34.21**	**0.8850**
$\times 3$	DIV2K + Flickr2K (DIF2K)	31.71	0.8293
	DIF2K + LSDIR	31.74	0.8297
	DIF2K + GTA5	31.73	0.8297
	DIF2K + LSDIR + GTA5	**31.75**	**0.8300**

(b) *Ablation on the network complexity for* $\times 2$. *# B and # C indicate the number of blocks and channels respetively.*

Method	# B	# C	Runtime (ms,↓)	PSNR (dB,↑)	SSIM (↑)
RT4KSR-XXS	2	24	**05.18**	34.13	0.8837
RT4KSR-XS		34	06.21	34.17	0.8842
RT4KSR-S	4	24	07.09	34.20	0.8848
RT4KSR-M		34	08.71	34.20	0.8849
RT4KSR-L	6	24	09.01	34.21	0.8851
RT4KSR-XL		32	11.22	**34.26**	**0.8857**
RT4KSR-XXL	8	24	10.92	34.23	0.8857
RT4KSR-XXXL		32	13.71	34.25	0.8857

Visual comparison. In Fig. 7, we show extracted crops from the 4K RTSR benchmark [10]. Also in Fig. 6 we provide SR results on a real 60MP image. Our model shows strong performance in reconstructing missing HF components from the LR input. Although our results still have room for improvement in dealing with shiny areas primarily found in computer-generated content, they produce sharper and visually more appealing outputs despite the presence of checkerboard artifacts.

4.3. Comparision to State of the Art

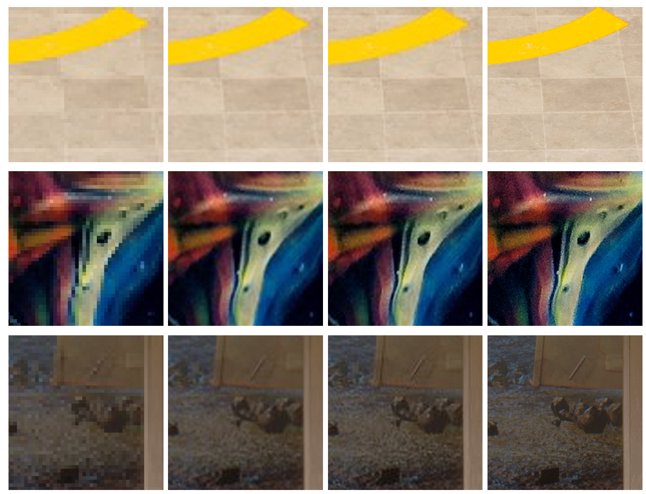
To ensure a fair comparison with published work, we exclusively train the *XL* and *XXXL* variants of our method for $\times 2$ and $\times 4$ SR on the DIV2K and Flickr2K datasets. Additionally, for $\times 4$ SR, we trained 64×64 crops, following the widely accepted training schedule in the SR literature. Attending to Tab. 4 our models are, on average, **755% smaller** than the approaches we compare them to, even those considered "lightweight," our performance is still impressive. While there is still a measurable gap, we are able to close it significantly in cases such as $\times 4$ SR on Set14 [55].

Table 4. *Quantitative comparison with state-of-the-art.* We compare RT4KSR-XL and RT4KSR-XXXL to published lightweight image SR methods and report SSIM and PSNR (Y) for $\times 2$ and $\times 4$ on standard benchmarks. Model sizes are compared w.r.t RT4KSR-XXXL.

Method	#Params	Set5 [4]				Set14 [55]				BSD100 [41]				Urban100 [21]			
		PSNR (dB,↑)		SSIM (↑)		PSNR (dB,↑)		SSIM (↑)		PSNR (dB,↑)		SSIM (↑)		PSNR (dB,↑)		SSIM (↑)	
Bicubic	-	34.01	28.76	.9309	.8113	31.27	26.72	.8652	.7351	29.99	26.19	.8539	.6803	28.65	24.85	.8498	.6719
LapSRN [29]	251K(+228%)	37.52	31.54	.9591	.8850	32.99	29.19	.9124	.7720	31.80	27.32	.8952	.7280	30.41	25.21	.9103	.7560
CARN [2]	1,592K(+1442%)	37.76	32.13	.9590	.8937	33.52	28.60	.9166	.7806	32.09	27.58	.8978	.7349	31.92	26.07	.9256	.7837
IMDN [22]	694K (+629%)	38.00	32.21	.9605	.8948	33.63	28.58	.9177	.7811	32.19	27.56	.8996	.7353	32.17	26.04	.9283	.7838
LatticeNet [39]	756K (+685%)	38.15	32.30	.9610	.8962	33.78	28.68	.9193	.7830	32.25	27.62	.9005	.7367	32.43	26.25	.9302	.7873
SwinIR [35]	878K (+795%)	38.14	32.44	.9611	.8976	33.86	28.77	.9206	.7858	32.31	27.69	.9012	.7406	32.76	26.47	.9340	.7980
RT4KSR-XL	91.8K (-17%)	36.83	30.43	.9545	.8600	33.46	28.02	.9197	.7806	31.76	27.09	.8935	.7213	30.75	25.83	.8955	.7208
RT4KSR-XXXL	110.4K	36.92	30.45	.9550	.8610	33.51	28.04	.9202	.7814	31.82	27.11	.8943	.7222	30.85	25.86	.8971	.7221

Bicubic Ours GT

Figure 6. *Qualitative samples.* Super-Resolution results on a real 60MP photography. Our method can recover structural elements and textures while being extremely efficient.



LR Bicubic Ours GT

Figure 7. *Rendered samples from 4K RTSR [10] benchmark.*

5. Conclusion

In this paper, we provide a comprehensive analysis of super-resolution techniques for efficiently upscaling images to 4K resolution from 720p and 1080p. To address this, we started with a simple, yet effective baseline architecture and derived a competitive design by focusing on extracting important high-frequency details and downsizing feature maps for efficiency. Over-parameterization during training allowed us to learn more expressive features and transfer knowledge into inexpensive 3×3 convolutions at inference time using structural re-parameterization. Our proposed method reduces significantly the overall computational footprint in comparison to previous approaches and achieves high reconstruction fidelity on the new 4K RTSR benchmark and other standard SR test sets.

Acknowledgements. This work was supported by the Humboldt Foundation and Sony Interactive Entertainment.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017. 2, 5, 7

[2] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *European Conference on Computer Vision*, pages 252–268, 2018. 2, 8

[3] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 3, 4, 5, 6, 7

[4] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie line Alberi Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *British Machine Vision Conference*, pages 135.1–135.10, 2012. 5, 8

[5] David Capel and Andrew Zisserman. Computer vision applied to super resolution. *IEEE Signal Processing Magazine*, 20(3):75–86, 2003. 1

[6] Hong Chang, Dit-Yan Yeung, and Yimin Xiong. Super-resolution through neighbor embedding. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 1, pages I–I, 2004. 1

[7] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. *arXiv preprint arXiv:2204.04676*, 2022. 1, 2, 3, 4, 5, 6

[8] Xiangyu Chen, Zhengwen Zhang, Jimmy S Ren, Lynhoo Tian, Yu Qiao, and Chao Dong. A new journey from sdrtv to hdrtv. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4500–4509, 2021. 2

[9] Marcos V Conde, Florin Vasluianu, Javier Vazquez-Corral, and Radu Timofte. Perceptual image enhancement for smartphone real-time applications. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1848–1858, 2023. 3

[10] 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 Workshops*, 2023. 1, 2, 3, 5, 6, 7, 8

[11] Senyou Deng, Wenqi Ren, Yanyang Yan, Tao Wang, Fenglong Song, and Xiaochun Cao. Multi-scale separable network for ultra-high-definition video deblurring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14030–14039, 2021. 2

[12] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13733–13742, 2021. 2, 4

[13] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *European Conference on Computer Vision*, pages 184–199, Cham, 2014. Springer International Publishing. 1, 2

[14] 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. 2

[15] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[16] Zongcai Du, Ding Liu, Jie Liu, Jie Tang, Gangshan Wu, and Lean Fu. Fast and memory-efficient network towards efficient image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 853–862, 2022. 2, 3, 4

[17] William T Freeman, Thouis R Jones, and Egon C Pasztor. Example-based super-resolution. *IEEE Computer graphics and Applications*, 22(2):56–65, 2002. 1

[18] Jinjin Gu, Haoming Cai, Chao Dong, Jimmy S Ren, Radu Timofte, Yuan Gong, Shanshan Lao, Shuwei Shi, Jiahao Wang, Sidi Yang, et al. Ntire 2022 challenge on perceptual image quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 951–967, 2022. 5

[19] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 3

[20] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4700–4708, 2017. 3

[21] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2015. 5, 8

[22] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *ACM International Conference on Multimedia*, pages 2024–2032, 2019. 2, 8

[23] Andrey Ignatov, Radu Timofte, Maurizio Denna, and Abdel Younes. Real-time quantized image super-resolution on mobile npus, mobile ai 2021 challenge: Report. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2525–2534, 2021. 2

[24] 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. In *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 92–129. Springer, 2023. 2, 3, 4

[25] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer, 2016. 5

[26] Soo Ye Kim, Jihyong Oh, and Munchurl Kim. Deep sr-itm: Joint learning of super-resolution and inverse tone-mapping for 4k uhd hdr applications. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3116–3125, 2019. 2

[27] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[28] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 766–776, 2022. 2

[29] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 624–632, 2017. 2, 8

[30] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4681–4690, 2017. 1

[31] Yawei Li, Kai Zhang, Luc Van Gool, Radu Timofte, et al. NTIRE 2022 challenge on efficient super-resolution: Methods and results. In *CVPR Workshops*, 2022. 2, 3

[32] 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. 5, 7

[33] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1062–1102, 2022. 2, 4, 5

[34] Yawei Li, Yulun Zhang, Luc Van Gool, Radu Timofte, et al. NTIRE 2023 challenge on efficient super-resolution: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 2

[35] 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. 1, 2, 8

[36] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 136–144, 2017. 1, 3

[37] 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, 4

[38] Zhisheng Lu, Juncheng Li, Hong Liu, Chaoyan Huang, Linlin Zhang, and Tieyong Zeng. Transformer for single image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 457–466, 2022. 2, 3, 4

[39] Xiaotong Luo, Yuan Xie, Yulun Zhang, Yanyun Qu, Cuihua Li, and Yun Fu. Latticenet: Towards lightweight image super-resolution with lattice block. In *European Conference on Computer Vision*, pages 272–289, 2020. 8

[40] Cheng Ma, Yongming Rao, Yean Cheng, Ce Chen, Jiwen Lu, and Jie Zhou. Structure-preserving super resolution with gradient guidance. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7769–7778, 2020. 3, 5

[41] 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 *IEEE Conference on International Conference on Computer Vision*, pages 416–423, 2001. 5, 8

[42] Matan Protter, Michael Elad, Hiroyuki Takeda, and Peyman Milanfar. Generalizing the nonlocal-means to super-resolution reconstruction. *IEEE Transactions on image processing*, 18(1):36–51, 2008. 1

[43] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 102–118. Springer, 2016. 7

[44] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016. 2, 3, 4, 5

[45] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, Lei Zhang, Bee Lim, et al. Ntire 2017 challenge on single image super-resolution: Methods and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017. 7

[46] Radu Timofte, Vincent De Smet, and Luc Van Gool. Anchored neighborhood regression for fast example-based super-resolution. In *IEEE Conference on International Conference on Computer Vision*, pages 1920–1927, 2013. 1

[47] Radu Timofte, Vincent De Smet, and Luc Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *ACCV*, 2014. 1

[48] Radu Timofte, Rasmus Rothe, and Luc Van Gool. Seven ways to improve example-based single image super resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1865–1873, 2016. 1

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017. 2

[50] Qilong Wang, Banggu Wu, Pengfei Zhu, Peihua Li, Wangmeng Zuo, and Qinghua Hu. Eca-net: Efficient channel at-

tention for deep convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11534–11542, 2020. 4, 5

[51] 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 *European Conference on Computer Vision Workshops*, pages 701–710, 2018. 1

[52] Lei Xiao, Salah Nouri, Matt Chapman, Alexander Fix, Douglas Lanman, and Anton Kaplanyan. Neural supersampling for real-time rendering. *ACM Transactions on Graphics (TOG)*, 39(4):142–1, 2020. 2

[53] Ren Yang, Radu Timofte, Xin Li, Qi Zhang, Lin Zhang, Fanglong Liu, Dongliang He, Fu Li, He Zheng, Weihang Yuan, et al. Aim 2022 challenge on super-resolution of compressed image and video: Dataset, methods and results. In *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 174–202. Springer, 2023. 2

[54] Hongliang Yuan, Boyu Zhang, Mingyan Zhu, Ligang Liu, and Jue Wang. High-quality supersampling via mask-reinforced deep learning for real-time rendering. *arXiv preprint arXiv:2301.01036*, 2023. 2

[55] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International Conference on Curves and Surfaces*, pages 711–730, 2010. 5, 7, 8

[56] Kai Zhang, Martin Danelljan, Yawei Li, and et al. AIM 2020 challenge on efficient super-resolution: Methods and results. In *ECCV Workshops*, 2020. 3

[57] Kai Zhang, Martin Danelljan, Yawei Li, Radu Timofte, Jie Liu, Jie Tang, Gangshan Wu, Yu Zhu, Xiangyu He, Wenjie Xu, et al. Aim 2020 challenge on efficient super-resolution: Methods and results. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 5–40, 2020. 2

[58] Kai Zhang, Luc Van Gool, and Radu Timofte. Deep unfolding network for image super-resolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3217–3226, 2020. 1

[59] Kaihao Zhang, Dongxu Li, Wenhan Luo, Wenqi Ren, Björn Stenger, Wei Liu, Hongdong Li, and Ming-Hsuan Yang. Benchmarking ultra-high-definition image super-resolution. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 14769–14778, 2021. 3

[60] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 5

[61] Xindong Zhang, Hui Zeng, and Lei Zhang. Edge-oriented convolution block for real-time super resolution on mobile devices. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 4034–4043, 2021. 2

[62] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *European Conference on Computer Vision*, pages 286–301, 2018. 1, 3, 4

[63] Yulun Zhang, Kunpeng Li, Kai Li, Bineng Zhong, and Yun Fu. Residual non-local attention networks for image restoration. *arXiv preprint arXiv:1903.10082*, 2019. 3

[64] 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. 2

[65] Lin Zhou, Haoming Cai, Jinjin Gu, Zheyuan Li, Yingqi Liu, Xiangyu Chen, Yu Qiao, and Chao Dong. Efficient image super-resolution using vast-receptive-field attention. In *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pages 256–272. Springer, 2023. 2