FreqNet: A Frequency-domain Image Super-Resolution Network with Discrete Cosine Transform

Runyuan Cai[a], Yue Ding[b], Hongtao Lu[a,*]

[a]*Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China*
[b]*School of Software, Shanghai Jiao Tong University, Shanghai, 200240, China*

Abstract

Single image super-resolution(SISR) is an ill-posed problem that aims to obtain high-resolution (HR) output from low-resolution (LR) input, during which extra high-frequency information is supposed to be added to improve the perceptual quality. Existing SISR works mainly operate in the spatial domain by minimizing the mean squared reconstruction error. Despite the high peak signal-to-noise ratios(PSNR) results, it is difficult to determine whether the model correctly adds desired high-frequency details. Some residual-based structures are proposed to guide the model to focus on high-frequency features implicitly. However, how to verify the fidelity of those artificial details remains a problem since the interpretation from spatial-domain metrics is limited. In this paper, we propose FreqNet, an intuitive pipeline from the frequency domain perspective, to solve this problem. Inspired by existing frequency-domain works, we convert images into discrete cosine transform (DCT) blocks, then reform them to obtain the DCT feature maps, which serve as the input and target of our model. A specialized pipeline is designed, and we further propose a frequency loss function to fit the nature of our frequency-domain task. Our SISR method in the frequency domain can learn the high-frequency information explicitly, provide fidelity and good perceptual quality for the SR images. We further observe that our model can be merged with other spatial super-resolution models to enhance the quality of their original SR output.

Keywords: Single Image Super-Resolution, Frequency Domain, Deep

---

*Corresponding author

---

1. Introduction

Single image super-resolution(SISR) aims to recover high-frequency details for a high-resolution(HR) image from one of its degraded low-resolution(LR) version. After years of development, the SISR has been widely used in many computer vision tasks, such as media content enhancement[1], medical imaging[2] and satellite imaging[3]. Traditional state-of-the-art SR methods mainly adopt the example-based[1] strategy, exploiting internal similarities or learning a mapping from the external dictionary. The sparse-coding-based SR[4] is one of the most representative methods.

Recently, deep convolutional neural network (CNN) based SISR methods have achieved significant improvements over traditional methods. Deep learning-based methods treat this problem as a dense image regression task, which learns an end-to-end image mapping function represented by a CNN between LR and HR images. Dong et al.[5] proposed SRCNN that first adopted deep learning into SISR using a three-layer CNN to represent the mapping function. Residual block[6] was later introduced into SISR in SRResNet[7] and improved in EDSR[8]. Residual block makes it possible to build deeper or wider networks. Zhang et al.[9] and Tong et al.[10] adopted dense blocks[11] to combine features from different levels. Zhang et al.[12] improved residual block by adding channel attention. Based on the progress of non-blind methods, blind super-resolution methods[13], which aim at complex degradation models in real scenarios, have received increasing attention recently.

The SISR methods mentioned above commonly use the minimization of the mean squared error (MSE) between the recovered SR image and the HR ground truth as the optimization target. Minimizing spatial MSE also maximizes the peak signal-to-noise ratio (PSNR), which is a common measure used to evaluate SR algorithms. However, such a pipeline often results in blurry effects because the high-frequency textures have been excessively destructed in the degrading process and are hard to predict. Generative adversarial networks (GANs)[14] based SISR approaches are proposed to relieve the above problems. However, the unpleasant hallucinations and artifacts caused by GANs further pose more challenges. Zhang et al.[12] further

proposed a residual-in-residual (RIR) structure to bypass the redundant low-frequency information through multiple skip connections, implicitly guiding the network to focus on learning high-frequency information. However, since the commonly used PSNR and structural similarity index measure(SSIM) are based on per-pixel loss and picture global information, respectively, their perception of high-frequency details is limited. To the best of our knowledge, current spatial domain-based methods do not have an explicit approach for learning high-frequency information and verifying the fidelity of output artificial details.

To practically resolve this problem, we propose FreqNet, a frequency-domain-based super-resolution network, to directly learn the reconstruction of high-frequency features. The proposed network contains two parallel flows: the Spatial Extraction Network(SEN) and the Frequency Reconstruction Network(FRN), in order to make use of both domains' information. We first convert both LR and HR images to frequency coefficients using discrete cosine transform (DCT)[15], then reshape them to obtain DCT feature maps. The SEN takes standard LR images as input, through the spatial feature reconstruction trunk and the down-sampling shrinking trunk to obtain one component of target HR DCT feature maps. The FRN is purely operated on frequency domain, which takes LR DCT feature maps as input, through the frequency-domain reconstruction trunk to obtain the other component. The weighted sum of two components makes our final frequency domain output, which can be converted to SR image through inverse discrete cosine transform(iDCT)[15]. Thanks to the characteristic of DCT, we can easily merge our output with any other SR model to enhance the high-frequency details of its output. We further propose depth-wise residual block(DWRB) and deformable residual block(DRB) to be implemented respectively in FRN and SEN that can better use the characteristics of the frequency domain feature maps. As the ability of spatial MSE (and PSNR) to capture high-frequency detail is very limited, we propose a frequency-domain loss function to evaluate the quality of the output SR image.

Overall, our contributions are three-fold: (1) We propose FreqNet, a frequency-domain-based SISR network, to learn the high-frequency features explicitly with a specially designed pipeline. Our network can produce perceptually satisfying results with high-fidelity details. (2) We propose depth-wise residual block structure and deformable residual block structure to fit the nature of frequency-domain feature extraction. Both structures can improve our network's reconstruction quality and feature extraction ability. (3)

We propose a frequency-domain loss function and a corresponding metric that measures output quality from the accuracy of high-frequency detail reconstruction.

2. Related Works

Numerous image deep learning-based SR methods have been studied in the computer vision community. Here we focus on works related to CNN-based methods and the works on frequency-domain learning.

2.1. Image Super-Resolution with CNN

Numerous methods have proven the effectiveness of the CNN-based pipeline on image super-resolution tasks. The pioneering work was done by Dong et al.[5], their proposed SRCNN for image SR achieved superior performance against previous works. Kim et al. proposed VDSR[16] and DRCN[17] by introducing residual learning to ease the training difficulty and significantly improve accuracy. Tai et al. introduced recursive blocks in DRRN[18] and memory blocks in MemNet[19]. A faster network structure FSRCNN[20] was proposed to accelerate the pipeline of SRCNN. Ledig et al.[7] introduced ResNet[6] to construct a deeper network, SRResNet, for image SR. They also proposed SRGAN with perceptual losses[21] and generative adversarial network (GAN)[14] for photo-realistic SR. Such GAN based model was then introduced in ESRGAN[22], which confirmed that dropping the batch normalization layers can result in better performance. Although SRGAN and ESRGAN can alleviate the blurring and over smoothing artifacts, their predicted results may not be faithfully reconstructed and produce unpleasing artifacts. By removing unnecessary modules in conventional residual networks, Lim et al.[8] proposed EDSR and MDSR, which achieve significant improvement. Zhang et al.[12] introduce channel attention to residual block. Blind-SR methods have also received increasing attention recently, aiming at complex degradation models in real scenarios by estimating degradation kernel using an extra module[23]. The methods of [24],[25] and [26] achieved state-of-the-art performance in real-world scenario with multiple modelling strategies.

However, all these CNN-based methods operate on the spatial domain. The information on the frequency domain is not directly used, though the recovery of high-frequency information is precisely the target of the image super-resolution task.
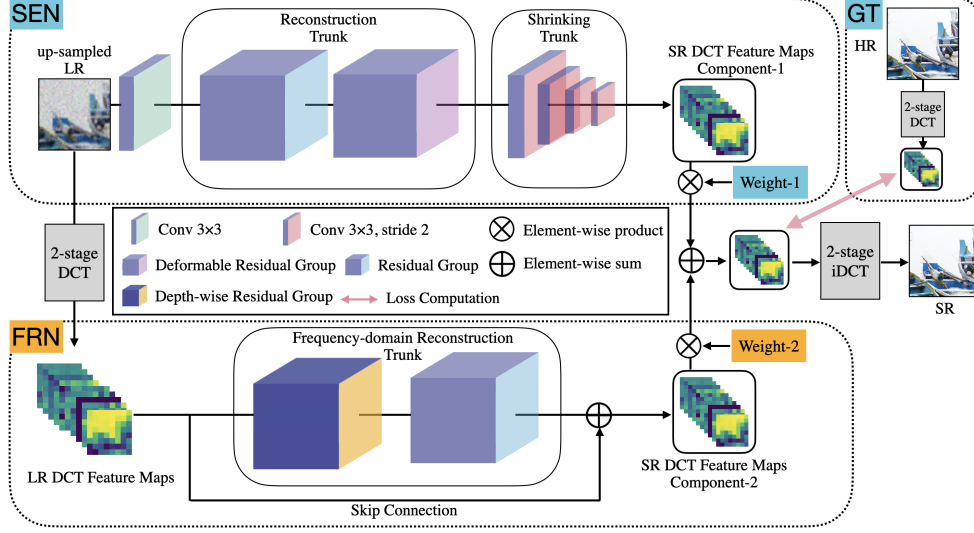
2.2. Frequency-Domain based Deep Learning

Projecting image to frequency domain provides a new perspective for various computer vision tasks. Remarkable performance has been achieved in some frequency-domain works. Works of [27] ,[28] and [29] jointly train auto-encoder-based networks on compression and inference tasks with frequency-domain input. [30] extracts features from the frequency domain to classify images. [31] proposes a model conversion algorithm to convert the spatial-domain CNN models to the frequency domain. [32] propose a method of learning in the frequency domain using DCT-based sparse image representations, proving that we can use frequency-domain information directly in current CNN models without a complex model transition procedure. [33] further translate the DCT representation into a sequence of DCT channel, spatial location, and DCT coefficient triples, and achieve state-of-art performance on image generation and restoration tasks with a Transformer-based auto-regressive architecture.

The essence of the SR task is to recover the information of high-frequency channels in the image. Hence, frequency-domain features are informative for HR reconstruction and can potentially enhance the performance with proper methods. However, there is no existing SR method using the characteristics of DCT feature maps. Hence, we propose a super-resolution pipeline on the DCT domain, which we will present in detail in the following section.

Precisely, the overall architecture is given in figure 1. In section 3.1, we introduce the image conversion process that projects the spatial image to spatial domain. In section 3.2 and 3.3, we explain the architecture and components of FreqNet in detail. The propsed frequency-domain loss function will be presented in section 3.4.
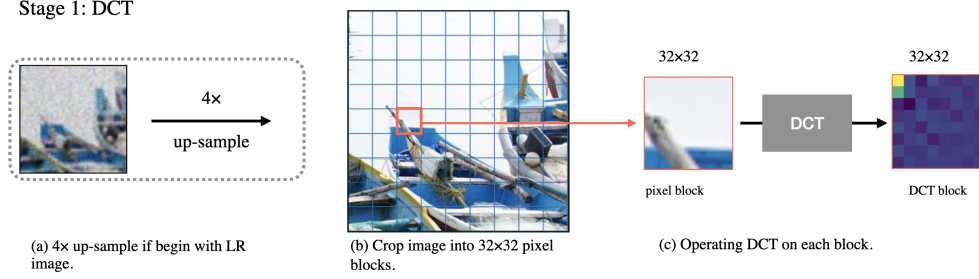
3. Method

Figure 1: The architecture of FreqNet. Our FreqNet contains two parallel data flows: the Spatial Extraction Network(SEN) and the Frequency Reconstruction Network(FRN), taking LR image I_{LR} and LR DCT feature maps as input, respectively. The final output is the weighted sum of predicted DCT feature maps from two sub-networks. Loss is computed between GT DCT feature maps(on the right-top of the figure) and the final output.

We propose a frequency-domain based pipeline for $4\times$ image super-resolution. Our method consists of an image conversion process that converts the spatial image to the frequency domain and a specialized network for training with the frequency domain information. As shown in 1, our proposed network consists of two parallel sub-network, respectively operates on spatial-domain and frequency-domain inputs to make use of both domains' information. We will first explain the image conversion process in the following section. Details of architecture will be discussed in 3.2.

3.1. Image Conversion to the Frequency Domain

Following the JPEG codec, we first transform the original RGB images to zero-centered normalized YCrCb color space, containing a brightness component Y (luma) and two color components Cb and Cr (chroma). Then we upsample the LR image to make it the same size as the HR image.

3.1.1. Generation of DCT Blocks

Figure 2: Converting pixel blocks to DCT blocks

To get frequency-domain information, we crop the images into uniform size of pixel blocks, then pass them through Discrete Cosine Transform(DCT) module. The DCT projects an image into a collection of cosine components which stands for different frequencies of 2D signals. Given a block size M, the two-dimensional DCT converts a zero-centered $M \times M$ pixel blocks P to obtain an $M \times M$ DCT block D, as interpreted below:

$$D_{uv} = \frac{1}{4}\alpha(u)\alpha(v) \times \sum_{i=0}^{M-1}\sum_{j=0}^{M-1} P_{ij}cos(\frac{(2i+1)u\pi}{2M})cos(\frac{(2j+1)v\pi}{2M}) \quad (1)$$
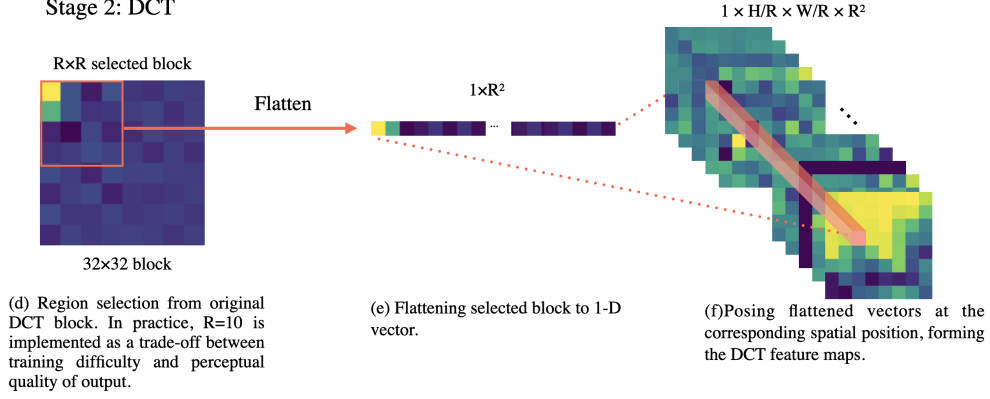
$$\alpha(x) = \begin{cases} \frac{1}{\sqrt{2}} & \text{, if } x = 0 \\ 1 & \text{, otherwise} \end{cases} \quad (2)$$

Where u and v are the horizontal and vertical index of frequencies in the DCT block, i and j stand for the horizontal and vertical index of pixel block, and α is a normalizing scale factor to enforce orthonormality.

For a standard DCT transform in JPEG codec, the block size M is 8, which indicates that any information in an 8×8 pixel block can be represented by a linear combination of 64 2D signals. However, in $4\times$ image super-resolution, the 8×8 block is upscaled to 32×32. Thus we perform DCT transform with the block size 32. At this stage, both the LR and HR images are converted into frequency-domain blocks that contain the DCT information of 1024 frequency channels.

3.1.2. Reforming DCT feature maps

Not all information in the 32×32 frequency range can be perceived for the perceptual ability of the human eye. Many other DCT based meth-

Stage 2: DCT

R×R selected block

Flatten

1×R²

1 × H/R × W/R × R²

32×32 block

(d) Region selection from original DCT block. In practice, R=10 is implemented as a trade-off between training difficulty and perceptual quality of output.

(e) Flattening selected block to 1-D vector.

(f)Posing flattened vectors at the corresponding spatial position, forming the DCT feature maps.

Figure 3: Reforming DCT blocks to DCT feature maps

ods induce sparsity to DCT blocks through quantization. However, for the super-resolution task, we tend to preserve the information as much as possible. Thus, as illustrated in figure 3, we perform a region selection on the DCT block. Only the values inside the left-top R×R selected region will be preserved for the next step. In practice, we choose $R = 10$ as a trade-off between training difficulty and perceptual quality. We will explain later how we handle the values outside the selected region.

Following the processing method proposed by [32], we flatten the DCT blocks to DCT vectors of length $1 \times R^2$. Then we pose these vectors at their corresponding spatial positions, forming a cuboid of size $H/R \times W/R \times R^2$, where H and W are the height and width of the original image. This cuboid is a collection of DCT feature maps, each channel at the third dimension is a frequency-domain feature map that contains the information of the frequency it represents.

3.1.3. Channel-wise Normalization

We further perform normalization on each frequency channel. For channel i of the frequency-domain feature maps M, we perform:

$$M_{i_{norm}} = \frac{(M_i - Mean_i)}{Std_i} \tag{3}$$

Where $Mean_i$ and Std_i denotes the mean and standard deviation of channel i that are pre-calculated on our training set.

8

Unlike quantization, this normalization process does not change the relative intensity of each feature map, thus guaranteeing the integrity of information. The purpose of this operation is to project the values to a suitable range for learning.

3.2. Architecture of FreqNet

As shown in figure 1, our FreqNet contains two parallel data flows: the Spatial Extraction Network(SEN) and the Frequency Reconstruction Network(FRN), in order to make use of both domains' information.

The SEN takes up-scaled LR image I_{LR} as input. Only one convolutional layer is used to extract the shallow feature $F_{shallow}$ from the LR input. $F_{shallow}$ is then passed through the Reconstruction Trunk(RT), which contains a sequence of multiple Residual Groups(RG)[12] and Deformable Residual Groups(DRG) to convert the spatial feature maps into frequency domain features F_{freq}. Then we feed F_{freq} to Shrinking Trunk(ST), which consists of 4 down-sampling convolution layers with $stride = 2$, to gradually shrink the scale of features maps while maintaining the channels. The final output M_{SR_1} is one component of target HR DCT feature maps. The overall process can be interpreted as:

$$F_{shallow} = H_{shallow}(I_{LR}) \tag{4}$$

$$F_{freq} = H_{RT}(F_{shallow}) \tag{5}$$

$$M_{SR_1} = H_{ST}(F_{freq}) \tag{6}$$

Where $H_{shallow}$ denotes the first convolution operation, H_{RT} and H_{ST} denote the RT and ST structure.

The FRN is purely operated on frequency domain. We take the pre-processed LR DCT feature maps M_{LR} as input, through the frequency-domain reconstruction trunk(FRT), which contains a sequence of depth-wise residual groups(DWRG) and RG to obtain the other component of target HR DCT feature maps, noted as M_{SR_2}. A skip-connection is added to take advantage of the similarities between input and the target, thus drawing attention towards the difference on high-frequency channels. The overall process can be interpreted as:

$$M_{SR_2} = H_{FRT}(M_{LR}) + M_{LR} \tag{7}$$

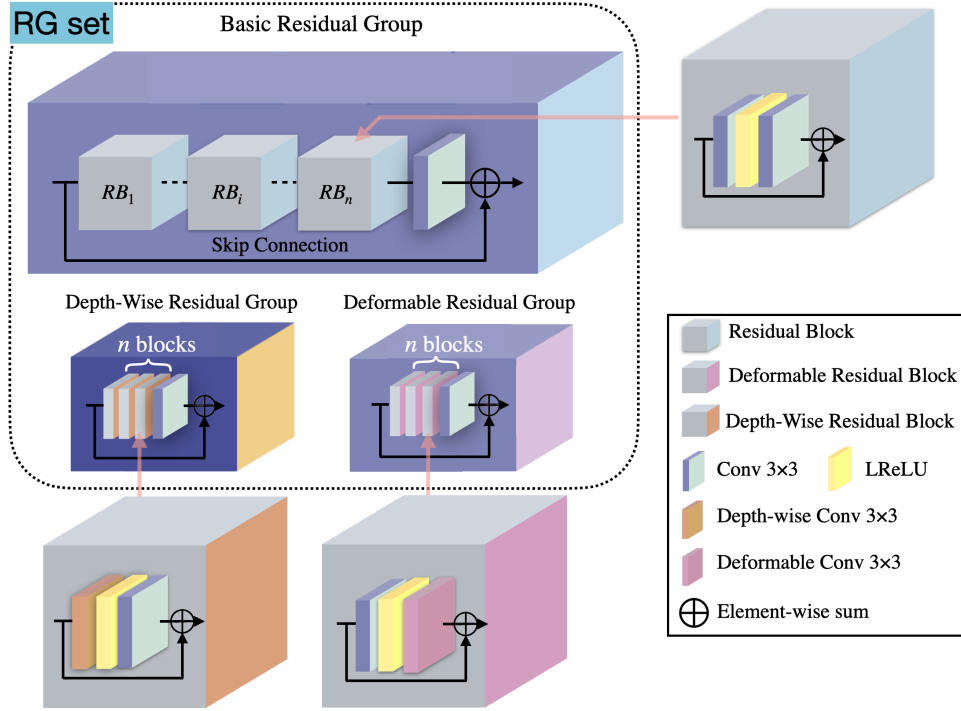Where H_{FRT} denotes the FRT structure.

9

The outputs of two sub-network have the same size, and a weighted element-wise sum is applied to get the final output:

$$M_{SR} = M_{SR_1} \odot W_1 + M_{SR_2} \odot W_2 \tag{8}$$

Where the W_1 and W_2 are the pre-defined weights for two components.

The output M_{SR} is further fed to a 2-stage inverse Discrete Cosine Transform(iDCT) module, which is an inverse flow of data-processing pipeline we defined in 3.1. We first project the M_{SR} back to its original range of values by performing denormalization on each channel. Then, in stage-1, we reform the $H/R \times W/R \times R^2$ DCT feature maps back to DCT blocks of size $R \times R$, and the rest of 32×32 block is filled with information from LR DCT blocks. Then we use iDCT to get the final SR image in stage-2.

3.3. Modified Residual Group

Figure 4: Different types of RG and RB implemented in FreqNet

10

Inspired by the success of residual groups(RG) in [12], we take it as the basic module of our network. As shown in figure 4, an RG is a sequence of n residual blocks(RB)[8] with an in-group skip connection between the input and output features. The original RB can be interpreted as:

$$F_{res} = H_{Conv_2}(LeakyReLu(H_{Conv_1}(F_{input})))$$ (9)

$$F_{output} = F_{input} + F_{res}$$ (10)

Where H_{Conv} denotes a convolution layer, F_{input} is the feature from last block and F_{output} is the feature towards next layer. As described in 3.1.3, the final output of network should be of zero-centered distribution, thus we replace ReLU layer by LeakyReLU with a high negative slope to fit our case.

3.3.1. Deformable Residual Group

The RG structure makes it possible to achieve large depth, consequently providing a large receptive field size. However, uniformly extending the receptive field does not always positively impact high-precision required tasks, such as the reconstruction of frequency-domain feature maps, due to the potential redundant information. Deformable convolution layer[34](DefConv) can be a solution. By learning an offset, DefConv provides the ability to constrain the sampling area. Each convolution operation only focuses on the valuable region, reducing the impact from the redundant receptive area.

Thus, as shown in figure 4 we further integrate DefConv into RB by partly replacing the original convolutional layers, introducing the deformable residual block(DRB), which is the basic module of deformable residual group(DRG):

$$F_{output} = F_{input} + H_{DefConv}(LeakyReLu(H_{Conv}(F_{input})))$$ (11)

Where $H_{DefConv}$ denotes the deformable convolution layer. The proposed DRB structure has better guidance on the receptive field, thus yield more accurate feature extraction from last layer. We implement DRG sequence in the reconstruction trunk of SEN sub-network, after a sequence of regular RG, to improve the robustness of reconstructed F_{freq}.

3.3.2. Depth-wise Residual Group

For most spatial domain tasks, the intermediate deep feature maps are abstract and strongly correlated. However, through the reforming method we defined in 3.1.2, the frequency-domain feature maps have concrete semantic information and share less correlation between each other. To better reflect

this characteristic, we propose the depth-wise residual block(DWRB) that replace the first convolution layer in RB by depth-wise convolution layer[35]:

$$F_{output} = F_{input} + H_{Conv}(LeakyReLu(H_{DWConv}(F_{input}))) \qquad (12)$$

Where H_{DWConv} denotes the depth-wise convolution layer. A depth-wise convolution layer performs 2-D convolution on each channel of the input without merging information from other channels, which is suitable to make the module focus on extracting information from own channel for the next stage of reconstruction, rather than relying on global information. Depth-wise residual group(DWRG) is the RG that deploy the DWRB instead of RB.

3.4. Frequency-domain Loss Function

The definition of our frequency-domain loss function L_{freq} is critical to the performance of our network. Commonly, the loss function of super-resolution task is based on pixel-wise Mean Square Error(MSE), as minimizing spatial MSE also maximizes the peak signal-to-noise ratio(PSNR). However, solutions from MSE optimization can achieve high PSNR while lacking high-frequency content, which results in unsatisfying perceptual quality with overly smooth textures[7].

For our frequency domain super-resolution, this problem can be solved in a intuitive method. Since the target is a series of frequency-domain feature maps with semantic meaning assigned to each channel, we can allocate different weights to each frequency channel while computing the loss, thus explicitly guide the network to focus on the reconstruction of selected high-frequency channels. Following [36], we further replace the MSE backbone by Charbonnier Loss that can better handle the outliers, which are more likely to appear in frequency-domain samples. The proposed frequency-domain loss function L_{freq} is calculated as:

$$L_{Char}(x_1, x_2) = \sqrt{(x_1 - x_2)^2 + \epsilon^2} \qquad (13)$$

$$L_{freq} = \frac{1}{R^2 W_{map} H_{map}} \sum_{c=1}^{R^2} \beta_c \sum_{x=1}^{W_{map}} \sum_{y=1}^{H_{map}} L_{Char}(M_{SR_{c,x,y}}, M_{HR_{c,x,y}}) \qquad (14)$$

Where L_{Char} is the backbone of Charbonnier Loss, W_{map} and H_{map} denotes the width and height of output feature maps, β_c denotes the weight assigned to channel c and M denotes the frequency-domain feature maps, as we previously define in equation 8.

4. Experiment Results

4.1. Experimental Settings

Our experimental settings about datasets, degradation models, evaluation metric and training settings are declared below:

Datasets and degradation model. Following [37] We use 800 training images from DIV2K dataset[37] as training set. For testing, we use four standard benchmark datasets: Set5[38], Set14[39], BSD100[40] and MANGA109[41]. We conduct experiments with Bicubic degradation model.

Evaluation Metrics. The SR results are evaluated with PSNR on the luminance channel(Y channel) of transformed YCrCb space. We also propose a frequency-domain reconstruction metric(FRM) on the luminance channel that measures the quality of high-frequency feature reconstructed:

$$FRM = 10 * log_{10}(\frac{1}{L_{freq}}) \tag{15}$$

Training Settings. We crop 800 training images into mini patches. Respectively, the size of cropped LR image is 32×32 and the size of cropped HR image is 256×256. The relative location of each pair of LR and HR patches is strictly identical. Our model is trained by ADAM optimizor[42], with $\beta_1 = 0.9$, $\beta_2 = 0.99$ and $\epsilon = 10^-8$. We implement Cosine Learning Rate(CosLR) strategy, which periodically adjust the learning rate at t_{th} epoch of i_{th} period with the equation:

$$\eta_{i,t} = \eta_{min} + \frac{1}{2}(\eta_{max} - \eta_{min})(1 + \cos \frac{t}{T_i}\pi) \tag{16}$$

Where the η_{max} is 10^-4 and η_{min} is 10^-7, the number of epochs in each period is 30. We use PyTorch[43] to implement our method with Nvidia Geforce RTX 2080 ti GPU.

The channel-wise weights allocation of our proposed loss function L_{freq}(Equation 13) will be discussed in detail in section4.3.

4.2. Results with Bicubic Degradation Model

We quantitatively compare our method with 8 State-of-the-art methods, including SRCNN[5], FSRCNN[20], EDSR[8], EDN[9], RRDB[22] and its perceptual-driven method ESRGAN[22], MSRResNet[7] and its perceptual-driven method MSRResNet-GAN[7]. We further perform visual comparisons with these two GAN-based methods and their PSRN-oriented version to demonstrate the perceptual quality and fidelity of the output from our model.
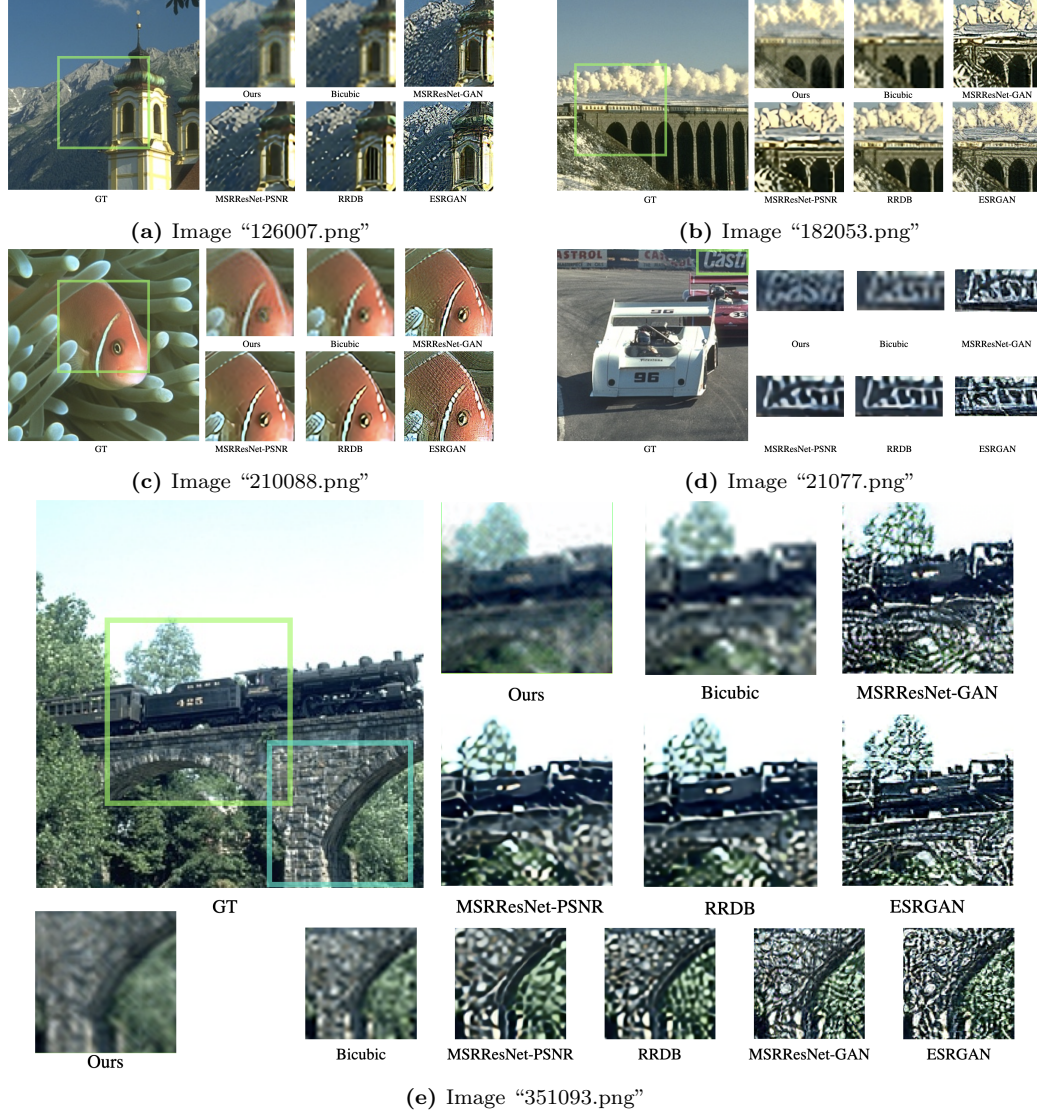
13

4.2.1. Quantitative Results by PSNR/FRM

Table 1 shows quantitative comparisons for our $4\times$ SR task, we compare the average PSNR and FRM on Y channel. The PSNR results of ESR-GAN and MSRResNet pair are computed using the released model. For the other models, the results are cited from their papers. All the FRM results are computed using released models. Our model has the best FRM with a slight decrease in PSNR value, which shows that our method has a more accurate reconstruction of key high-frequency information. Meanwhile, although GAN-based methods visually provide more high-frequency details, their FRM values are generally low, reflecting the lack of accuracy of high-frequency information reconstructed by such methods. We will discuss the visual behavior in detail in section 4.2.2.

Table 1: Quantitative results with Bicubic degradation model on Y channel. Best and second best results are **highlighted** and <u>underlined</u>.

Method	Set5		Set14		Manga109		BSD100	
	PSNR	FRM	PSNR	FRM	PSNR	FRM	PSNR	FRM
Bicubic	28.78	40.06	26.38	39.11	24.89	39.65	26.33	38.97
SRCNN	30.48	40.01	27.50	39.09	27.58	39.71	26.90	39.11
FSRCNN	30.72	40.13	27.61	39.12	27.90	39.77	26.98	39.09
MSRResNet	32.22	40.19	28.63	39.26	30.48	40.04	27.59	39.31
MSRResNet-GAN	29.40	39.64	26.02	38.84	27.69	39.12	25.16	39.01
EDSR	32.46	40.32	28.80	39.61	<u>31.02</u>	40.46	27.71	39.25
RDN	<u>32.47</u>	40.27	<u>28.81</u>	39.47	31.00	<u>40.71</u>	<u>27.71</u>	39.23
RRDB	**32.60**	<u>40.34</u>	**28.88**	<u>40.14</u>	**31.16**	40.63	**27.76**	<u>39.52</u>
ESRGAN	29.56	39.38	26.19	38.79	28.03	39.28	25.32	38.86
FreqNet(Ours)	32.08	**43.56**	28.47	**42.60**	30.23	**40.91**	27.51	**40.87**

4.2.2. Visual Results

(a) Image "126007.png"

(b) Image "182053.png"

(c) Image "210088.png"

(d) Image "21077.png"

(e) Image "351093.png"

Figure 5: Visual comparison for 4× SR with Bicubic Degradation model on BSD100 datasets.

In figure 5, we show visual comparisons of SR results with the Bicubic Degradation model on BSD100 datasets. For images "126007.png" and "351093.png", we observe that our method has more precise building contours than the PSNR method, contains more details, and does not have the

excessive texture as in the GAN method. For image "210088.png", we observe that our method produces the best face pattern and eye details for the clownfish. For image "21077.png", our method better restores the text "cas" over the other methods. And in image "182053.png", our method predicts the arches correctly while having fewer unnecessary artifacts.

4.3. Effects of Frequency-domain Loss Function and Modified RG

We study the effects of proposed Deformable Residual Group, Depth-wise Residual Group and the Frequency-domain Loss Function.

4.3.1. Settings and Effects of Frequency-domain Loss Function.
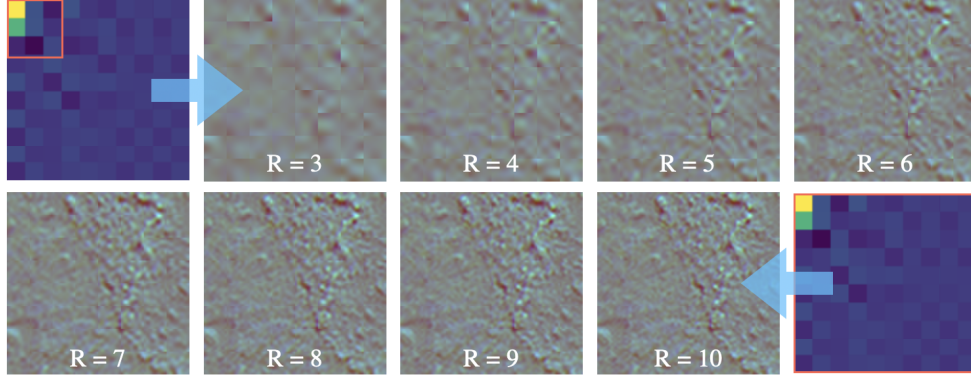
As we defined in Equation 13, each channel has a pre-assigned weight. We propose a statistical solution to decide the weight of each channel coarsely. As shown in Figure 6, given a pair of HR and up-sampled LR DCT blocks, after the region selection process(i.e. Figure 3, (d)), of size $R \times R$, we perform 8 times of computation in total. For the i-th computation, we keep the $(i+2) \times (i+2)$ DCT region at the left-top of original DCT block unchanged, and set the values outside the selected region to be 0. Then we perform iDCT on both LR and HR DCT blocks and compute the mean pixel-wise residual res_i of two converted pixel blocks as:

$$res_i = \left| \frac{I_{HR} - I_{LR}}{R^2} \right| \tag{17}$$

Table 2: Weight Allocation.

Region:	3	4-3	5-4	6-5	7-6	8-7	9-8	10-9
β_c	1	1	5	10	10	5	1	1

We randomly picked 1000 samples from the training set to perform the statistics by accumulating res_i. We define $res_0 = 0$, then for each i, the value $v_i = res_i - res_{i-1}$ reflects the difference between HR and LR images while considering the addition frequency channels of $R = i + 2$, which is proportional to their importance. Therefore, based on $[v_i]_{i \in [1,8]}$, we allocate weights as the table 2 shows, where Region $i - (i-1)$ denotes the additional channels between the left-top $i \times i$ region and $(i-1) \times (i-1)$ region of the $R \times R$ DCT block, and β denotes the weight assigned to these channels involved in Equation 13.

16

Figure 6: Progressively calculate residuals between HR and LR pixel-blocks under different size of region selection.

To demonstrate the effect of the proposed frequency-domain loss function L_{freq}, we run the training process with MSE and L_{freq} respectively, and compare the output of two models on Set5. Both the PSNR and FRM of L_{freq} supervised model is higher than the MSE supervised model, and the output images contain more accurate high-frequency texture. Figure 7 shows the comparison of the SR results of image "bird.png" between MSE-supervised and L_{freq}-supervised FreqNet after same number of iterations. The L_{freq}-supervised model can produce more high-frequency details.

GT   FreqNet-L_{freq} FreqNet-MSE

Figure 7: Comparison of MSE-supervised and L_{freq}-supervised FreqNet. L_{freq} supervised result contains more high-frequency details.

4.3.2. Effects of Deformable(DRG) and Depth-wise Residual Group(DWRG).

We perform a series of ablation experiments by replacing DRG or/and DWRG with original RG, to demonstrate the effect of our modified RG structure.

Table 3: Ablation Experiments on DWRG and DWG. We use PSNR and our proposed FRM as the metric.
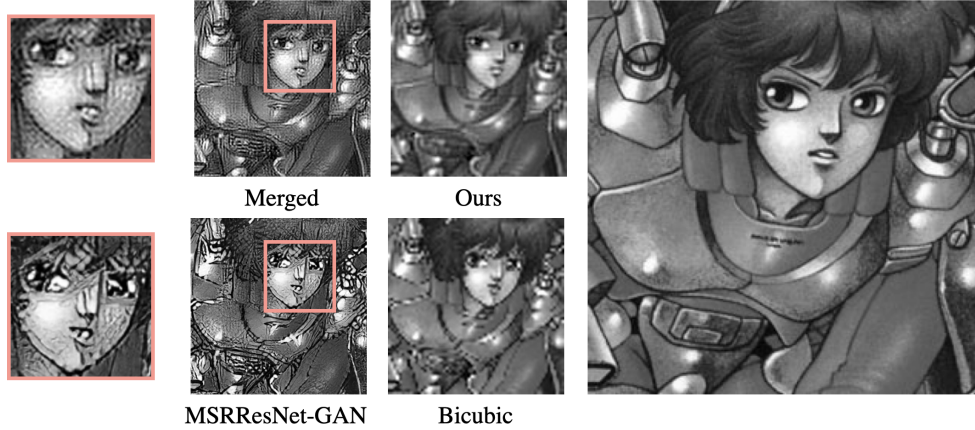
DRG	✗	✓	✗	✓
DWRG	✗	✗	✓	✓
PSNR on Set5	31.88	32.06	31.91	**32.08**
FRM on Set5	43.24	43.51	43.29	**43.56**

Respectively, in Spatial Extraction Network(SEN) we set $num_{DRG} = 3$ and $num_{RG} = 7$, in Frequency Reconstruction Network(FRN) we set $num_{DWRG} = 3$ and $num_{RG} = 7$. For each group, the number of residual blocks is set as 10. As shown in Table 3, the PSNR on Set5 increased by 0.18 dB when we replace specific RG with DRG, increased by 0.03 dB when we replace specific DWRG, and we can have the best performance by using both of them. The FRM on Set5 also increased when we replace RG with DRG and DWRG, by 0.27 and 0.05 respectively. The comparison shows the effectiveness of our proposed modified Residual Group architectures.

4.4. High-frequency Detail Enhancement based on other SR Models

As the output of our proposed model is a group of separated frequency-domain feature maps, we can easily merge our output with the output of other SR models, thus realize the enhancement on selected high-frequency channels. We first perform a similar process as 3.1 to convert the output from other SR model $I_{SR_{ori}}$ to its frequency-domain feature maps group $F_{SR_{ori}}$, then we replace certain channels in $F_{SR_{ori}}$ with the corresponding channels in F_{output} from FreqNet to get the merged output F_{merge}.

Figure 8: Merging with the output of FreqNet can reduce unreasonable artifacts from the GAN method while maintaining the details in the picture.

Specifically, we can merge our output with GAN-based SR models. As shown in Figure 8, we merge the output of MSRResNet-GAN[22] with the output of our model, for image "ARMS.png" in "Manga109"[41], the results are presented in Y-channel. The excessive artifacts from GAN can be corrected by channel replacement, and the reasonable high-frequency information that doesn't ruin the fidelity can be preserved. This method is practical when the output is blurred due to the difficulty of prediction.

5. Conclusions

We propose FreqNet, a frequency-domain image super-resolution model that explicitly learn the reconstruction of high-frequency details from LR images. We propose the depth-wise residual group(DWRG) and deformable residual group(DRG) structure to fit the characteristics of frequency-domain task and improve the ability of our network. Meanwhile, we propose a frequency-domain loss function and the frequency-domain reconstruction metric(FRM) that can measure the quality of high-frequency detail reconstruction. The quantitative and visual results demonstrate the effectiveness of our method, and we can further merge the output of our network with the other SR models as a post-processing enhancement.

References

[1] D. Glasner, S. Bagon, M. Irani, Super-resolution from a single image, 2009, pp. 349 – 356. `doi:10.1109/ICCV.2009.5459271`.

[2] O. Oktay, W. Bai, M. Lee, R. Guerrero, K. Kamnitsas, J. Caballero, A. de Marvao, S. Cook, D. O'Regan, D. Rueckert, Multi-input cardiac image super-resolution using convolutional neural networks, in: S. Ourselin, L. Joskowicz, M. R. Sabuncu, G. Unal, W. Wells (Eds.), Medical Image Computing and Computer-Assisted Intervention - MICCAI 2016, Springer International Publishing, Cham, 2016, pp. 246–254.

[3] D. Yıldırım, O. Gungor, A novel image fusion method using ikonos satellite images, Journal of Geodesy and Geoinformation 1 (2012) 27–34. `doi:10.9733/jgg.170512.1`.

[4] J. Yang, J. Wright, T. Huang, Y. Ma, Image super-resolution as sparse representation of raw image patches, in: 2008 IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8. `doi:10.1109/CVPR.2008.4587647`.

[5] C. Dong, C. C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks, IEEE Transactions on Pattern Analysis and Machine Intelligence 38 (2016) 295–307.

[6] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition (2015). `arXiv:1512.03385`.

[7] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, W. Shi, Photo-realistic single image super-resolution using a generative adversarial network, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017) 105–114.

[8] B. Lim, S. Son, H. Kim, S. Nah, K. M. Lee, Enhanced deep residual networks for single image super-resolution (2017). `arXiv:1707.02921`.

[9] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, Y. Fu, Residual dense network for image super-resolution (2018). `arXiv:1802.08797`.

[10] T. Tong, G. Li, X. Liu, Q. Gao, Image super-resolution using dense skip connections, in: 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 4809–4817. `doi:10.1109/ICCV.2017.514`.

[11] G. Huang, Z. Liu, K. Q. Weinberger, Densely connected convolutional networks, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017) 2261–2269.

[12] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, Y. R. Fu, Image super-resolution using very deep residual channel attention networks, in: ECCV, 2018.

[13] A. Liu, Y. Liu, J. Gu, Y. Qiao, C. Dong, Blind image super-resolution: A survey and beyond, ArXiv abs/2107.03055 (2021).

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Commun. ACM 63 (11) (2020) 139–144. doi:10.1145/3422622. URL https://doi.org/10.1145/3422622

[15] K. R. Rao, P. Yip, Discrete Cosine Transform: Algorithms, Advantages, Applications, Academic Press Professional, Inc., USA, 1990.

[16] J. Kim, J. K. Lee, K. M. Lee, Accurate image super-resolution using very deep convolutional networks, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016) 1646–1654.

[17] J. Kim, J. K. Lee, K. M. Lee, Deeply-recursive convolutional network for image super-resolution, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016) 1637–1645.

[18] Y. Tai, J. Yang, X. Liu, Image super-resolution via deep recursive residual network, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2790–2798. doi:10.1109/CVPR.2017.298.

[19] Y. Tai, J. Yang, X. Liu, C. Xu, Memnet: A persistent memory network for image restoration, 2017 IEEE International Conference on Computer Vision (ICCV) (2017) 4549–4557.

[20] C. Dong, C. C. Loy, X. Tang, Accelerating the super-resolution convolutional neural network, in: ECCV, 2016.

[21] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in: ECCV, 2016.

[22] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, C. C. Loy, Y. Qiao, X. Tang, Esrgan: Enhanced super-resolution generative adversarial networks, in: ECCV Workshops, 2018.

[23] J. Gu, H. Lu, W. Zuo, C. Dong, Blind super-resolution with iterative kernel correction, 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019) 1604–1613.

[24] K. Zhang, J. Liang, L. V. Gool, R. Timofte, Designing a practical degradation model for deep blind image super-resolution, ArXiv abs/2103.14006 (2021).

[25] S. Bell-Kligler, A. Shocher, M. Irani, Blind super-resolution kernel estimation using an internal-gan, in: NeurIPS, 2019.

[26] X. Wang, L. Xie, C. Dong, Y. Shan, Real-esrgan: Training real-world blind super-resolution with pure synthetic data, ArXiv abs/2107.10833 (2021).

[27] R. Torfason, F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, L. V. Gool, Towards image understanding from deep compression without decoding, ArXiv abs/1803.06131 (2018).

[28] K. Xu, Z. Zhang, F. Ren, Lapran: A scalable laplacian pyramid reconstructive adversarial network for flexible compressive sensing reconstruction, ArXiv abs/1807.09388 (2018).

[29] C.-Y. Wu, M. Zaheer, H. Hu, R. Manmatha, A. Smola, P. Krähenbühl, Compressed video action recognition, 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018) 6026–6035.

[30] L. Gueguen, A. Sergeev, B. Kadlec, R. Liu, J. Yosinski, Faster neural networks straight from jpeg, in: S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 31, Curran Associates, Inc., 2018.

[31] M. Ehrlich, L. S. Davis, Deep residual learning in the jpeg transform domain, 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (2019) 3483–3492.

[32] K. Xu, M. Qin, F. Sun, Y. Wang, Y. kuang Chen, F. Ren, Learning in the frequency domain, 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020) 1737–1746.

[33] C. Nash, J. Menick, S. Dieleman, P. W. Battaglia, Generating images with sparse representations, ArXiv abs/2103.03841 (2021).

[34] X. Zhu, H. Hu, S. C.-F. Lin, J. Dai, Deformable convnets v2: More deformable, better results, 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019) 9300–9308.

[35] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, Mobilenets: Efficient convolutional neural networks for mobile vision applications, ArXiv abs/1704.04861 (2017).

[36] W.-S. Lai, J.-B. Huang, N. Ahuja, M.-H. Yang, Fast and accurate image super-resolution with deep laplacian pyramid networks, IEEE Transactions on Pattern Analysis and Machine Intelligence 41 (2019) 2599–2613.

[37] R. Timofte, E. Agustsson, L. V. Gool, M.-H. Yang, L. Zhang, B. Lim, S. Son, H. Kim, S. Nah, K. M. Lee, X. Wang, Y. Tian, K. Yu, Y. Zhang, S. Wu, C. Dong, L. Lin, Y. Qiao, C. C. Loy, W. Bae, J. Yoo, Y. Han, J. C. Ye, J.-S. Choi, M. Kim, Y. Fan, J. Yu, W. Han, D. Liu, H. Yu, Z. Wang, H. Shi, X. Wang, T. S. Huang, Y. Chen, K. Zhang, W. Zuo, Z. Tang, L. Luo, S. Li, M. Fu, L. Cao, W. Heng, G. Bui, T. Le, Y. Duan, D. Tao, R. Wang, X. Lin, J. Pang, J. Xu, Y. Zhao, X. Xu, J. Pan, D. Sun, Y. Zhang, X. Song, Y. Dai, X. Qin, X.-P. Huynh, T. Guo, H. S. Mousavi, T. H. Vu, V. Monga, C. Cruz, K. Egiazarian, V. Katkovnik, R. Mehta, A. K. Jain, A. Agarwalla, C. V. S. Praveen, R. Zhou, H. Wen, C. Zhu, Z. Xia, Z. Wang, Q. Guo, Ntire 2017 challenge on single image super-resolution: Methods and results, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2017, pp. 1110–1121. doi:10.1109/CVPRW.2017.149.

[38] M. Bevilacqua, A. Roumy, C. Guillemot, M. line Alberi Morel, Low-complexity single-image super-resolution based on nonnegative neighbor embedding, in: Proceedings of the British Machine Vision Conference, BMVA Press, 2012, pp. 135.1–135.10. doi:http://dx.doi.org/10.5244/C.26.135.

[39] R. Zeyde, M. Elad, M. Protter, On single image scale-up using sparse-representations, in: J.-D. Boissonnat, P. Chenin, A. Cohen, C. Gout, T. Lyche, M.-L. Mazure, L. Schumaker (Eds.), Curves and Surfaces, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 711–730.

[40] D. Martin, C. Fowlkes, D. Tal, J. 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, Vol. 2, 2001, pp. 416–423 vol.2. doi:10.1109/ICCV.2001.937655.

[41] Y. Matsui, K. Ito, Y. Aramaki, A. Fujimoto, T. Ogawa, T. Yamasaki, K. Aizawa, Sketch-based manga retrieval using manga109 dataset, Multimedia Tools and Applications 76 (2016) 21811–21838.

[42] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, CoRR abs/1412.6980 (2015).

[43] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch, 2017.