Rethinking Image Super Resolution from Long-Tailed Distribution Learning Perspective

Yuanbiao Gou[1], Peng Hu[1], Jiancheng Lv[1], Hongyuan Zhu[2], Xi Peng[1]*
[1] College of Computer Science, Sichuan University, China
[2] Institute for Infocomm Research (I^{2}R), A*STAR, Singapore
{gouyuanbiao, penghu.ml, hongyuanzhu.cn, pengx.gm}@gmail.com; lvjiancheng@scu.edu.cn

Abstract

Existing studies have empirically observed that the resolution of the low-frequency region is easier to enhance than that of the high-frequency one. Although plentiful works have been devoted to alleviating this problem, little understanding is given to explain it. In this paper, we try to give a feasible answer from a machine learning perspective, i.e., the twin fitting problem caused by the long-tailed pixel distribution in natural images. With this explanation, we reformulate image super resolution (SR) as a long-tailed distribution learning problem and solve it by bridging the gaps of the problem between in low- and high-level vision tasks. As a result, we design a long-tailed distribution learning solution, that rebalances the gradients from the pixels in the low- and high-frequency region, by introducing a static and a learnable structure prior. The learned SR model achieves better balance on the fitting of the low- and high-frequency region so that the overall performance is improved. In the experiments, we evaluate the solution on four CNN- and one Transformer-based SR models w.r.t. six datasets and three tasks, and experimental results demonstrate its superiority.

1. Introduction

Image super resolution aims to restore a high-resolution (HR) image from a low-resolution (LR) one, which is an important technique in image processing [13,26,27,52] and computer vision [7,14,18,45,51]. In the past decades, plentiful SR methods have been proposed [19,53], and applied to a wide range of real-world applications [21,47,49,54].

Among existing studies, the learning-based methods that learn a mapping between LR and HR image spaces have achieved the state-of-the-art performance [17,39,43,58,59]. Nonetheless, they have empirically observed that the high-frequency regions are harder to be super-resolved than the low-frequency ones in the natural image. To alleviate that, various SR methods have been proposed following the below two paradigms, *i.e.*, developing generalized models with larger capacities [31,36] or specific models with high-frequency enhancements [37,48]. The former obtains better results in both the high- and low-frequency regions via constantly enlarging the capacities, while the latter enhances the high-frequency regions through specific auxiliary subnetworks, loss functions, training strategies, *etc*. Although the promising results have been obtained, they involve the following three limitations. First, the large capacity models take a lot of time and computations in the training and infer-

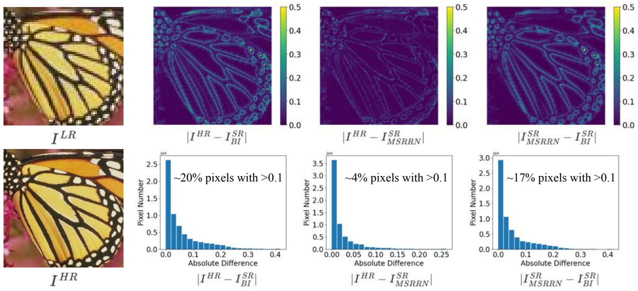


Figure 1. The long-tailed pixel distribution in the natural image. For given a HR image I^{HR}, we take ×4 LR version I^{LR} as a showcase, and utilize Bicubic Interpolation (BI) and MSRResNet [25] (MSRRN) to super-resolve it, *i.e.*, I^{SR}_{BI} and I^{SR}_{MSRRN}, respectively. The top row shows the absolute difference (AD) in the luminance channel, and the bottom row shows the pixel number at different AD intervals. From the top row, one could observe that i) both BI and MSRRN achieve better results in the low- than high-frequency regions; ii) MSRRN performs significantly better than BI in the high-frequency regions while slightly better in the low ones. From the bottom row, one could see that iii) the pixel distribution w.r.t. the low- and high-frequency region is long-tailed, *i.e.*, the number of pixels in the low-frequency regions is far more than that in the high-frequency ones. Clearly, such an imbalanced pixel distribution necessarily results in the twin fitting problem, *i.e.*, overfitting majority pixels in the low-frequency region while underfitting minority pixels in the high-frequency one.

*Corresponding author

ring, which is unavailable to mobile scenarios. Second, the specific models need ingenious designs about the architecture and training strategy, which is difficult to training and prone to artifacts. Third, they don't dive into the problem and give a reasonable explanation, thus alleviate the problem not in the most cost-effective way.

In this paper, we dive into the problem and explain it from a machine learning perspective, *i.e*., the twin fitting problem caused by the long-tailed pixel distribution in the natural images. Taking the Fig. 1 as an example, the number of pixels in the low-frequency region is far more than that in the high-frequency one, *i.e*., the long-tailed pixel distribution. Since majority pixels in the low-frequency region dominate minority pixels in the high-frequency one, the gradients of SR model are mainly from the former instead of the latter. As a result, the SR model is optimized to mainly fit the pixels in the low-frequency region, and thus overfitting them while underfitting those in the high-frequency region, *i.e*., the twin fitting problem.

Motivated by the above explanation, we reformulate SR as the long-tailed distribution learning problem. With this reformulation, the twin fitting problem could be alleviated during training in a model-agnostic way, and thus applicable to different SR models. However, although the long-tailed distribution learning problem has been extensively studied in high-level vision tasks, there are few works on it in low-level ones. Therefore, we bridge the gaps of the problem between in low- and high-level vision ones, and design a simple and effective solution to verify the feasibility of our reformulation. To be specific, we design a novel long-tailed distribution learning method for SR, termed as Focal Pixel Learning (FPL), which adaptively re-weights the loss contribution of pixels by combining two complementary structure priors. In this way, the gradients of SR model could be rebalanced, leading it to achieve better balance on the fitting of the high- and low-frequency regions.

The contributions of this work are summarized below.

- For the first time, this work dives into the observation that the high-frequency regions are harder to be super-resolved than the low-frequency ones, and gives a reasonable explanation, *i.e*., the long-tailed pixel distribution and it caused twin fitting problem.
- With our explanation, this work reformulates SR as a long-tailed distribution learning problem and designs a novel solution to verify its feasibility, which could be the first long-tailed distribution learning solution for SR, as far as we know.
- Extensive analyses and experiments are conducted to demonstrate the explanation, verify the reformulation, and validate the solution. The results demonstrate that our works could consistently improve the performance of SR models with different complexities.

2. Related Works

Here, we briefly review the related works of image super resolution and long-tailed distribution learning.

2.1. Image Super Resolution

Although a lot of SR models have been proposed [11, 16, 33, 42], they are advanced along two directions. One is to develop generalized models with larger capacities. For example, SRCNN [9] introduced the convolution neural network into SR for the first time, and outperformed the traditional methods. RDN [58] introduced the dense connections to utilize the hierarchical features from all convolutional layers. SwinIR [31] introduced the shifted windowing scheme to model the local attention and long-range dependency. ESRT [36] utilized a convolution neural network to extract deep features first and then used a Transformer to model the long-range dependency. The other direction is to develop specific models with high-frequency enhancements. For instance, PCL [48] proposed a contrastive learning framework to enhance LR images into sharp HR images. SPSR [37] introduced a gradient map SR network to guide the high-frequency region recovery in the image SR network. WDST [8] decomposed an image into high- and low-frequency sub-bands, and separately enhanced them via different subnetworks. SA [44] introduced a heuristic metric to exploit informative LR-HR patch pairs for training.

Different from them, this work dives into the observation that the high-frequency regions are harder to be super-resolved than the low-frequency regions, and explains it from the twin fitting problem caused by the long-tailed pixel distribution. Inspired by this explanation, this work reformulates SR as a long-tailed distribution learning problem, and designs a novel model-agnostic solution. This solution could endow the SR model with the better recovery capability of the high-frequency regions, without introducing extra model capacities or auxiliary strategies.

2.2. Long-tailed Distribution Learning

In real-world applications, samples typically exhibit a long-tailed distribution, where a small portion of classes have massive samples but the others are with only a few samples. With the unbalanced training data, models would be biased towards majority classes with massive samples, leading to poor performance on minority classes that have limited samples [4, 32, 46]. To address the problem, massive studies have been conducted in recent years [6, 22, 34, 55, 56], which could be categorized into class re-balancing, information augmentation and module improvement. Here, we briefly review the related category of class re-balancing, and more details could be referred to [57].

Class re-balancing is the main paradigm in the long-tailed distribution learning, which balances training samples of different classes through re-sampling, cost-sensitive learning, and/or logit adjustment. As logit adjustment post-hoc shifts the logits based on label frequencies, we focus on re-sampling and cost-sensitive learning which act on the training process. To be specific, re-sampling usually under-samples the majority classes and/or over-samples the minority classes, *e.g.*, SMOTE [5] linearly interpolates samples for the minority classes, and UNSAM [41] learns a data sampler to discard samples. Besides, cost-sensitive learning re-balances classes by re-weighting the loss contribution of different classes during the training. For example, CB [6] re-weights the loss value to be inverse to the effective number of samples per class. FL [32] down-weights the loss values assigned to the majority classes and the well-classified examples. GHM [28] re-weights the loss values of samples based on their gradients per iteration.

Although the long-tailed distribution learning has been extensively studied in high-level vision tasks, such as classification and detection, few studies are conducted in low-level ones. Different from the existing works, this work first reveals the long-tailed pixel distribution in natural images and the twin fitting problem in SR, and then achieves SR in a long-tailed distribution learning paradigm. To the best of our knowledge, this work could be the first study on this topic for SR.

3. The Proposed Solution

In this section, we first theoretically explain why the SR models suffer from the twin fitting problem, and then elaborate on how does the proposed solution learn from the long-tailed pixel distribution and alleviates the problem.

3.1. Twin Fitting Problem in SR

SR models are generally trained through pixel-wise loss function ℓ_p on a large number of natural images. Specifically, for a given HR image y as well as the LR counterpart x, the pixel-wise loss function ℓ_p is

$$\mathcal{L} = \frac{1}{I}\sum_{i=1}^{I} |f(x)_i - y_i|^p, \tag{1}$$

where $f(\cdot)$ denotes the SR model, i and I indicate the pixel index and number, respectively. With the formula, we could observe that ℓ_p treats all pixels equally, *i.e.*, every pixel, either in the high- or low-frequency region, equally contribute to the overall loss value. However, as depicted in Fig. 1, the pixels in the natural images show a long-tailed distribution, *i.e.*, the number of pixels in the low-frequency region is far more than that in the high-frequency region. For clarity, let M and N denote the pixel number in the low- and high-frequency region, respectively. Therefore, ℓ_p in Eq. (1) could be reformulated as the sum of the loss values in the two regions, *i.e.*,

$$\mathcal{L} = \frac{1}{I}\sum_{m=1}^{M} |f(x)_m - y_m|^p + \frac{1}{I}\sum_{n=1}^{N} |f(x)_n - y_n|^p, \quad s.t.\ M \gg N \text{ and } M + N = I. \tag{2}$$

Obviously, the majority pixels in the low-frequency region could easily dominate the minority pixels in the high-frequency region on the contribution of overall loss value. Therefore, the gradients of SR model are mainly from the low-frequency regions and be biased toward fitting them. As a result, SR model would overfit the low-frequency regions while underfitting the high-frequency ones. Here, we term this competing fitting issue as the twin fitting problem, which is particularly severe in the limited capacity models.

3.2. Long-tailed Distribution Learning for SR

As the twin fitting problem arises from the long-tailed pixel distribution, it is highly expected to solve it from the long-tailed distribution learning perspective, *i.e.*, recasting SR as a long-tailed distribution learning problem. Although plentiful methods have been proposed to solve it in high-level vision tasks, there are few studies in low-level vision tasks due to the following two obstacles. First, no semantic class label to indicate a pixel belonging to the high- or low-frequency region since it depends on the surroundings. Second, no specific boundary between high- and low-frequency regions since the frequencies are distributed continuously. In other words, it is daunting to conduct long-tailed distribution learning for low-level vision tasks due to the absence of discrete and semantic labels for pixels. Therefore, to solve the twin fitting problem through long-tailed pixel distribution learning, we should overcome the two obstacles.

3.2.1 Two Structure Priors

To overcome the first obstacle, we introduce two structure priors to jointly indicate the pixel distribution about the low- and high-frequency region in a heuristic way.

The structure prior y_{sp} comes from the observation illustrated in Fig. 1, *i.e.*, BI usually achieves better results in the low- than high-frequency regions. Namely, pixels with large absolute differences between HR image y and BI image $f_{BI}(x)$ have a high probability of being in the high-frequency region. Based on the observation, we formulate a static structure prior as below,

$$y_{sp} = |y - f_{BI}(x)|, \tag{3}$$

in which the pixels with small values are more likely to be in the low-frequency region, while the pixels with large values are more likely to be in the high-frequency region. Therefore, y_{sp} could be viewed as the sketchy labels for the pixels in different frequency regions.

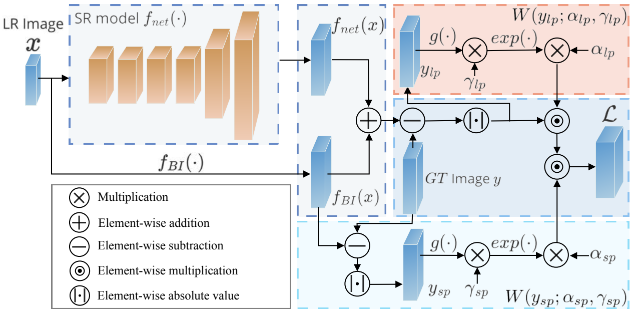


Figure 2. Overview of the proposed FPL derived from Eq. (7). For a given SR model $f(\cdot)$, FPL endows it with the capacity of learning from the long-tailed pixel distribution through adaptive re-sampling and re-weighting.

Real-time fitting degree is significant for distinguishing pixels in different frequency regions, and solving the twin fitting problem. Thus, as a remedy to y_{sp}, a learnable structure prior y_{lp} is designed based on the observation illustrated in Fig. 1, *i.e.*, SR models easily achieve better results in the low- than high-frequency regions. In other words, the pixels with large absolute differences between HR image y and SR image $f(x)$ are more likely to be those not be and hard to be fitted in the high-frequency region. With this observation, we formulate a learnable structure prior as below,

$$y_{lp} = |y - f(x)|, \tag{4}$$

in which the small values indicate the pixels that are easy to be fitted in the low-frequency regions, while the large values indicate those hard to be fitted in the high-frequency ones. Hence, y_{lp} could be viewed as the subtle labels for pixels labeled through the criterion of fitting difficulty. Meanwhile, y_{lp} could endow the model with the capacity of knowing itself fitting degree, thus dynamically adjusting the training process for solving the twin fitting problem.

In summary, we introduce the two structure priors as the heuristic labels to indicate the pixels (*i.e.*, y_{sp}) as well as those easy and hard to be fitted (*i.e.*, y_{lp}) in the low- and high-frequency regions. As a result, the first obstacle could be overcome.

3.2.2 Focal Pixel Learning

The second obstacle is no specific boundaries among the different frequencies, and thus the pixels cannot be divided into either high- or low-frequency ones. To overcome that, we follow the cost-sensitive learning paradigm, which reweights the loss contributions of different classes during the training. As a result, we transform the structure priors into soft weights for re-weighting the loss contribution per pixel.

Given the structure priors y_{sp} and y_{lp}, we first harmonize them so that they are in a comparable magnitude, *i.e.*, $g(z) := \frac{z - min(z)}{max(z) - min(z)}$ where $z \in \{y_{sp}, y_{lp}\}$. Further, as their lower bounds are zero, we introduce the exponential function to transform them into the non-zero weights, *i.e.*, $W(z) := \exp(g(z))$. Meanwhile, two hyper-parameters α, γ are introduced into $W(z)$, so that the weights are more flexible and controllable, *i.e.*,

$$W(z; \alpha, \gamma) := \alpha \cdot \exp(\gamma \cdot g(z)). \tag{5}$$

With the weighting function, the structure priors could be transformed into the weights, while keeping the relative size of values, *i.e.*, the large weights correspond to the pixels in the high-frequency region, while the small weights correspond to those in the low-frequency one. With the weighting matrices, long-tailed distribution learning for SR could be achieved in a cost-sensitive learning way.

As illustrated in Fig. 1, SR models usually perform remarkably better in the high-frequency regions while slightly better in the low-frequency ones than BI. Therefore, we introduce BI into the solution and endow it with a novel connotation. Specifically, BI is essentially a re-sampling strategy for long-tailed pixel distribution learning, which undersamples the pixels in the low-frequency region, *i.e.*,

$$\hat{y} = y - f_{BI}(x). \tag{6}$$

In other words, the number of pixels in the low-frequency region that need to be fitted is decreased and thus allowing SR models to fit the residual pixels that on a less extremely long-tailed distribution.

Combining the two strategies, we formulate a long-tailed distribution learning solution for SR, termed as Focal Pixel Learning (FPL), *i.e.*,

$$\mathcal{L}_i = W(y_{sp}; \alpha_{sp}, \gamma_{sp})_i \times W(y_{lp}; \alpha_{lp}, \gamma_{lp})_i \times |f(x)_i - \hat{y}_i|^p, \tag{7}$$

and the overall loss value is $\mathcal{L} = \frac{1}{I}\sum_{i=1}^{I} \mathcal{L}_i$. Intuitively, FPL introduces BI to under-sample the majority pixels, while introducing two structure priors and a weighting function to re-balance the pixel-wise contribution in Eq. (2). As a result, SR models focus more on learning from the pixels and those hard to be fitted in the high-frequency regions. Note that as y_{lp} involves the current fitting degree, the HR image y in Eq. (4) should be substituted with $\hat{y}$. Besides, the gradients are blocked in the weighting function.

4. Experiments

In this section, we devote to verifying the feasibility of our rethinkings, *i.e.*, achieving SR in a long-tailed distribution learning paradigm, so that the twin fitting problem could be alleviated. In the following, we will detail the experimental settings first, and then assess FPL on four CNN- and one Transformer-based methods about three SR tasks and six SR datasets. Finally, we will conduct some analysis experiments to demonstrate the effectiveness of FPL.

Due to the space limitation, we present more experiments in supplementary material.

4.1. Experimental Settings

We use DIV2K [1] as the training dataset which contains 800 images of 2K resolution. Following ClassSR [24], we densely crop 159M LR and HR image pairs with the sizes of i) 32×32 and 128×128 for $4\times$ SR, and ii) 32×32 and 64×64 for $2\times$ SR, respectively. For evaluations, six widely used datasets are employed, *i.e.*, Set5 [3], Set14 [50], BSD100 [38], Manga109 [12], Urban100 [20], and Test2K [24]. To measure the performance, two metrics of PSNR and SSIM in RGB color space are used.

As FPL is compatible with existing SR models, we introduce the models of FSRCNN [10] (tiny, 25K/468M[1]), CARN [2] (small, 295K/1.16G), SRResNet [25] (middle, 1.5M/4.56G) and MSRN [29] (large, 6.1M/13.4G), which are representative CNN models at different complexities, and SwinIR [31] (lightweight, 1.2M/3.4G) which is a Transformer model. All models are implemented in PyTorch [40], and the experiments are conducted on Ubuntu 18.04 with GeForce RTX TITAN GPUs.

We train CARN, SRResNet, and MSRN with the iterations of 1000K, FSRCNN and SwinIR with that of 500k, based on the batch size 16 and the patch size 32 for LR images. Meanwhile, the images are augmented by flipping and rotation during the training. The initial learning rate is set to 1e-3 for FSRCNN and CARN, and 2e-4 for SRResNet, MSRN, and SwinIR. To optimize the models, we adopt the Adam [23] optimizer with the default settings, as well as the cosine annealing learning strategy [35] with the minimum learning rate of 1e-7. For a better reproducibility, we do not exhaustively tune the models on FPL. Instead, we fix $\alpha_{sp} = 0.5$, $\gamma_{sp} = 1$, $\alpha_{lp} = 1$, and $\gamma_{lp} = 1$ based on FSRCNN throughout the experiments, regardless the differences in models, datasets, SR tasks, *etc*.

4.2. Comparison Experiments

We first extend the models with our FPL to obtain the variants, and then respectively train and evaluate them and their variants using the same settings on the six datasets and the three SR tasks.

Comparisons on $4\times$ SR task. Tab. 1 shows that "+FPL" consistently improves the performance, where the small capacity models obtain significant improvements. To be specific, FPL boosts FSRCNN with the PSNR/SSIM gains of 0.63dB/0.0127 on the Manga109, and CARN with that of 0.24dB/0.0098 on the Urban100. As model capacity grows, the performance improvements are less significant but still attractive. For example, the PSNR/SSIM gains of SRResNet are 0.21dB/0.0082 on the Urban100. With such an improvement, SRResNet+FPL (1.5M/4.56G, the middle size) outperforms MSRN (6.1M/13.4G, the large size model) by 0.07dB/0.0033 in PSNR/SSIM. Analogously, MSRN+FPL outperforms SwinIR by 0.05dB/0.0034 on the Urban100. There are two reasons that affect the performance gains. First, larger capacity models fit the pixels better both in the low- and high-frequency regions, and thus less suffer from the long-tailed distribution. Second, higher resolution images have a higher proportion of low-frequency regions, and thus less benefit from alleviating the twin fitting problem.

Fig. 3 shows the qualitative and quantitative results, from which one could see that FPL enables the models to produce clearer structures, richer details, and higher PSNR/SSIM values. Some areas are highlighted by color rectangles, and more results could be referred to supplementary material.

Comparisons on $2\times$ SR task. The quantitative results are shown in Tab. 2, which demonstrate that the FPL improves the performance of the SR models in most cases. From the table, one could see that FPL boosts FSRCNN with the gains from 0.16dB/0.0012 to 0.58dB/0.0089, and CARN with that from 0.14dB/0.0008 to 0.49dB/0.0063 in PSNR/SSIM on test datasets. Meanwhile, SRResNet+FPL obtains 0.14dB∼0.57dB gains in PSNR, while MSRN+FPL and SwinIR+FPL achieve PSNR gains at most 0.11dB and 0.09dB, respectively.

Comparisons on $4\times$ SR with multiple degradations. We introduce multiple degradations including downsampling, blurring, and noise into $4\times$ SR task, *i.e.*,

$$x = (y \downarrow_s) \otimes k + n, \tag{8}$$

where $\downarrow_s$ is the s scale bicubic downsampler, k is the blur kernel, and n is the additive white Gaussian noise. For better reproducibility, we fix the blur kernel k with the kernel size of 3×3 and the kernel standard deviation of 5, and the noise n with the noise level of 5. The quantitative results are shown in Tab. 3, from which one could see that FPL still improves the performance of the SR models, *e.g.*, FSRCNN, CARN, SRResNet, and MSRN achieve PSNR/SSIM gains at most 0.29dB/0.0116, 0.22dB/0.0089, 0.09dB/0.0049, and 0.05dB/0.0043, respectively.

4.3. Analysis Experiments

Effectiveness on State-of-the-Art Lightweight Model. The comparison experiments have shown the effects of FPL on different capacity models. As a supplementary, we conduct the evaluations on BSRN [30], which is the first place in the model complexity track of NTIRE 2022 Efficient SR Challenge. The results are shown in Tab. 4, which shows the attractive performance improvements of BSRN from FPL.

Ablation Studies and Parameter Analyses. To investigate the parameters in the weighting functions, we change one of them while remaining the others unchanged. Specifically, we first change γ_{sp} and γ_{lp} with 0, 1, 2, where $\gamma_{sp} = 0$

[1] The FLOPs on 32×32 sub-image.

Table 1. Quantitative results on 4× SR task. "+FPL" indicates the corresponding model trained with FPL on the same settings, and "Gains" denotes the performance improvement from it.

Method	Set5		Set14		BSD100		Manga109		Urban100		Test2K	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
FSRCNN	28.71	0.8500	25.84	0.7389	25.58	0.7122	25.78	0.8346	22.99	0.7184	25.64	0.7567
+FPL	29.00	0.8565	26.01	0.7446	25.70	0.7178	26.41	0.8473	23.26	0.7313	25.72	0.7616
Gains	0.29	0.0065	0.17	0.0057	0.12	0.0056	0.63	0.0127	0.27	0.0129	0.08	0.0049
CARN	29.94	0.8737	26.61	0.7605	26.07	0.7296	28.00	0.8787	24.12	0.7639	26.03	0.7734
+FPL	30.11	0.8765	26.69	0.7641	26.16	0.7338	28.33	0.8836	24.36	0.7737	26.10	0.7779
Gains	0.17	0.0028	0.08	0.0036	0.09	0.0042	0.33	0.0049	0.24	0.0098	0.07	0.0045
SRResNet	30.13	0.8771	26.75	0.7648	26.21	0.7342	28.44	0.8844	24.47	0.7772	26.12	0.7781
+FPL	30.27	0.8795	26.85	0.7677	26.26	0.7376	28.63	0.8874	24.68	0.7854	26.20	0.7819
Gains	0.14	0.0024	0.10	0.0029	0.05	0.0034	0.19	0.0030	0.21	0.0082	0.08	0.0038
MSRN	30.24	0.8786	26.84	0.7672	26.25	0.7361	28.69	0.8875	24.61	0.7821	26.23	0.7811
+FPL	30.36	0.8800	26.89	0.7695	26.28	0.7385	28.81	0.8895	24.75	0.7877	26.25	0.7839
Gains	0.12	0.0014	0.05	0.0023	0.03	0.0024	0.12	0.0020	0.14	0.0056	0.02	0.0028
SwinIR	30.34	0.8795	26.87	0.7670	26.27	0.7365	28.76	0.8898	24.70	0.7843	26.24	0.7821
+FPL	30.39	0.8805	26.91	0.7688	26.29	0.7393	28.83	0.8908	24.78	0.7880	26.26	0.7847
Gains	0.05	0.0010	0.04	0.0018	0.02	0.0028	0.07	0.0010	0.08	0.0037	0.02	0.0026

Table 2. Quantitative results on 2× SR task. "+FPL" indicates the corresponding model trained with FPL on the same settings, and "Gains" denotes the performance improvement from it.

Method	Set5		Set14		BSD100		Manga109		Urban100		Test2K	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
FSRCNN	34.74	0.9456	30.56	0.8944	30.01	0.8909	34.13	0.9608	27.92	0.8936	30.21	0.9158
+FPL	34.97	0.9468	30.74	0.8967	30.21	0.8948	34.47	0.9621	28.50	0.9025	30.37	0.9184
Gains	0.23	0.0012	0.18	0.0023	0.20	0.0039	0.34	0.0013	0.58	0.0089	0.16	0.0026
CARN	35.30	0.9489	31.14	0.9007	30.47	0.8977	35.19	0.9655	29.34	0.9127	30.74	0.9235
+FPL	35.48	0.9497	31.30	0.9031	30.61	0.9006	35.49	0.9669	29.83	0.9190	30.93	0.9263
Gains	0.18	0.0008	0.16	0.0024	0.14	0.0029	0.30	0.0014	0.49	0.0063	0.19	0.0028
SRResNet	35.48	0.9496	31.26	0.9018	30.51	0.8994	35.52	0.9668	29.71	0.9171	30.50	0.9246
+FPL	35.62	0.9503	31.48	0.9041	30.74	0.9021	35.78	0.9680	30.28	0.9240	30.99	0.9239
Gains	0.14	0.0007	0.22	0.0023	0.23	0.0027	0.26	0.0012	0.57	0.0069	0.49	-0.0007
MSRN	35.66	0.9506	31.55	0.9049	30.75	0.9012	35.87	0.9682	30.36	0.9240	31.24	0.9294
+FPL	35.67	0.9507	31.62	0.9065	30.75	0.9023	35.90	0.9683	30.47	0.9259	31.23	0.9302
Gains	0.01	0.0001	0.07	0.0016	0.00	0.0011	0.03	0.0001	0.11	0.0019	-0.01	0.0008
SwinIR	35.70	0.9507	31.55	0.9050	30.75	0.9010	35.94	0.9688	30.37	0.9242	31.32	0.9304
+FPL	35.73	0.9509	31.59	0.9059	30.75	0.9022	35.93	0.9688	30.46	0.9257	31.32	0.9312
Gains	0.03	0.0002	0.04	0.0009	0.00	0.0012	-0.01	0.00	0.09	0.0015	0.00	0.0008

or $\gamma_{lp} = 0$ denotes to remove one of the weighting function, and $\gamma_{sp} = 0, \gamma_{lp} = 0$ disables both of the weighting functions, while BI is remained for under-sampling. The results are shown in Fig. 4, from which one could see that the indispensable roles of them. Note that as the BI under-sampling revises the pixel distribution to a less extremely long-tailed one, it obtains better performance gains than the priors which balance the fitting degree of the model for improvements. But generally, the best performance is obtained by integrating them together. Second, as α_{sp} and α_{lp} should be greater than zero, we change them with 0.1, 0.5, 1, 2. As shown in Fig. 4, there is no very obvious performance changes among different parameters' settings. Overall, although the parameters affect the performance, FPL could achieve the consistent improvements.

Weighting matrix visualizations. We visualize weighting matrices from the two structure priors and their element-wise product for observing the attentions on HR images. As shown in Fig. 5, the weighting matrix (b) from the static structure prior has large values in the high-frequency regions, while that (c) from the learnable structure prior has large values on the hard pixels in high-frequency regions.

Table 3. Quantitative results on $4\times$ SR task with multiple degradations. "+FPL" indicates the corresponding model trained with FPL on the same settings, and "Gains" denotes the performance improvement from it.

Method	Set5		Set14		BSD100		Manga109		Urban100		Test2K	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
FSRCNN	24.73	0.6859	23.13	0.5809	23.45	0.5598	22.08	0.6861	20.77	0.5640	23.55	0.6129
+FPL	24.91	0.6957	23.23	0.5869	23.54	0.5658	22.37	0.6977	20.89	0.5737	23.57	0.6140
Gains	0.18	0.0098	0.10	0.0060	0.09	0.0060	0.29	0.0116	0.12	0.0097	0.02	0.0011
CARN	25.69	0.7330	23.91	0.6188	23.90	0.5865	23.84	0.7588	21.60	0.6210	23.64	0.6309
+FPL	25.80	0.7381	23.99	0.6236	23.96	0.5912	24.06	0.7668	21.72	0.6299	23.63	0.6338
Gains	0.11	0.0051	0.08	0.0048	0.06	0.0047	0.22	0.0080	0.12	0.0089	-0.01	0.0029
SRResNet	25.86	0.7403	24.08	0.6256	23.99	0.5918	24.29	0.7722	21.89	0.6382	23.61	0.6320
+FPL	25.95	0.7431	24.08	0.6276	24.02	0.5952	24.38	0.7760	21.92	0.6431	23.52	0.6337
Gains	0.09	0.0028	0.00	0.0020	0.03	0.0034	0.09	0.0038	0.03	0.0049	-0.09	0.0017
MSRN	26.01	0.7459	24.09	0.6266	24.02	0.5932	24.40	0.7757	21.95	0.6412	23.59	0.6324
+FPL	26.05	0.7480	24.10	0.6291	24.03	0.5958	24.45	0.7782	21.97	0.6455	23.48	0.6331
Gains	0.04	0.0021	0.01	0.0025	0.01	0.0026	0.05	0.0025	0.02	0.0043	-0.11	0.0007

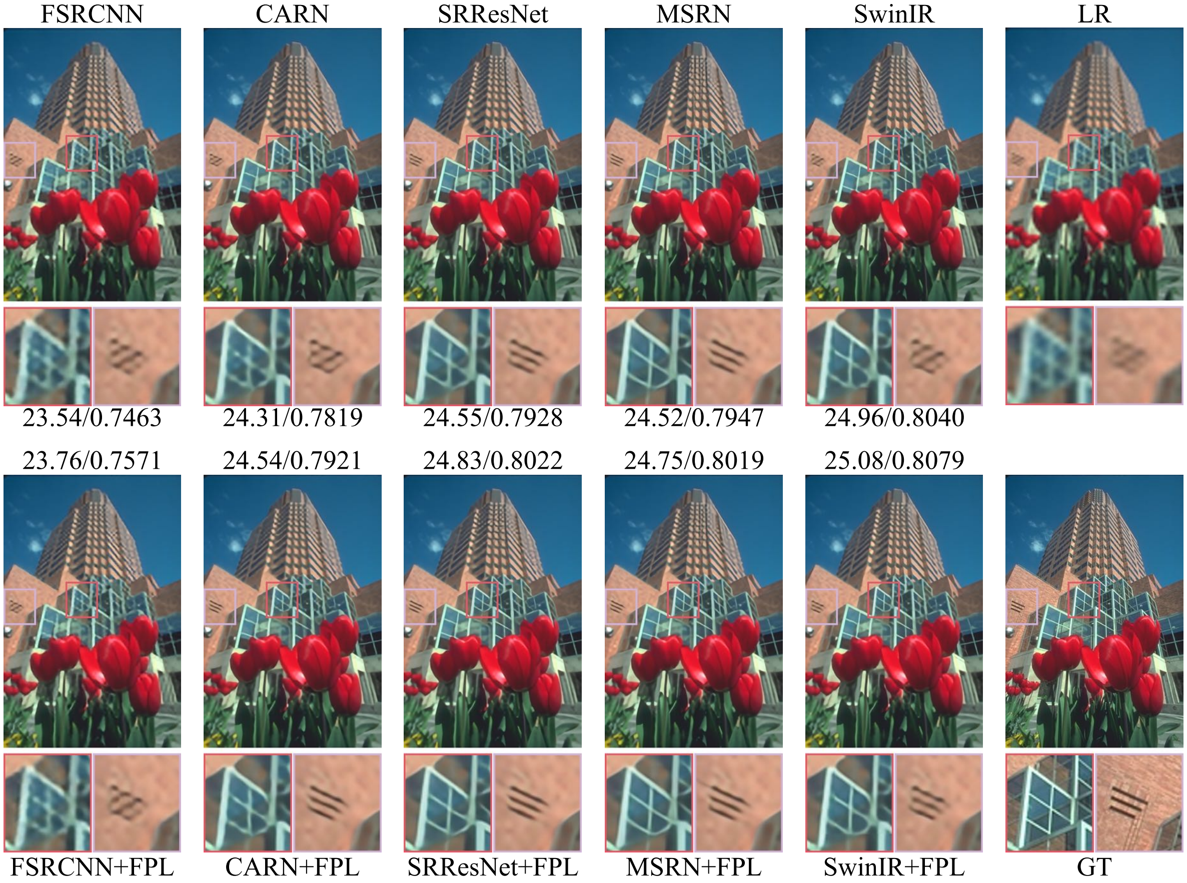


Figure 3. Qualitative and quantitative (PSNR/SSIM) results on $4\times$ SR task.

As a result, their element-wise product (d) integrates the two weighting matrices to down-weight the pixels in the low-frequency regions, while up-weighting the pixels in the high-frequency regions, in which the hard pixels are with larger weight values than the easy ones. For example, the edges of the "hat" are high-frequency regions according to the weighting matrix (b), and relatively easy to be super-resolved in terms of the weighting matrix (c), thus they are

Table 4. Quantitative results of BSRN [30] on 4× SR task.

Method	Set5		Urban100	
	PSNR	SSIM	PSNR	SSIM
BSRN	32.35	0.8966	26.27	0.7908
+FPL	32.50	0.8972	26.34	0.7942
Gains	0.15	0.0006	0.07	0.0034

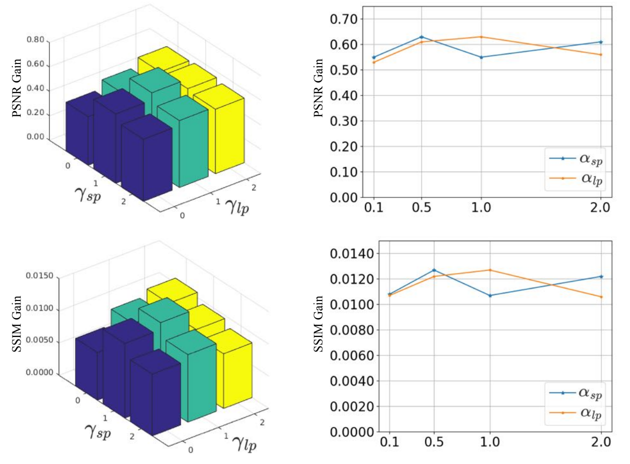
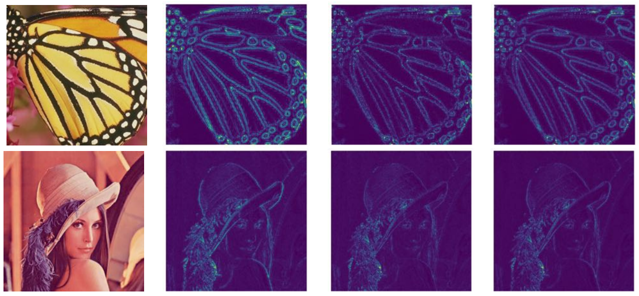


Figure 4. Ablation studies and parameter analyses on Manga109. The left column is the results of changing γ_{sp} and γ_{lp}, and the right column is the results of changing α_{sp} and α_{lp}, where PSNR/SSIM gain denotes the performance improvements over FSRCNN.



(a) GT images (b) Weights from static structure prior (c) Weights from learnable structure prior (d) Weights of element-wise products

Figure 5. The weighting matrices from the two structure priors and their element-wise product, which show significant attention on the pixels in the high-frequency regions.

assigned with eclectic weight values in the overall weighting matrix (d).

Attribution Analyses. To investigate the effects of FPL on SR models, we preform attribution analyses on FSRCNN and FSRCNN+FPL through the local attribution map (LAM) and the diffusion index (DI) proposed by [15]. In brief, LAM illustrates the contribution of each pixel in the LR image to the SR result of a given patch. DI is a statistical dispersion measure, and a smaller value represents fewer pixels significantly contribute to the SR result, while a higher one means more pixels significantly contribute to that. The attribution results are shown in Fig. 6, from which one could see that, i) FSRCNN and FSRCNN+FPL have the same receptive fields; ii) comparing with FSRCNN, FSRCNN+FPL involves more pixels which produce significant contributions, and thus obtaining a higher DI value; iii) the involved pixels mainly belong to the high-frequency region, which are distributed along the edges and textures. According to the observations, we could conclude that FPL boosts SR models by making them more effectively exploit the high-frequency information within the receptive fields, *i.e.*, SR models fit the high-frequency regions better. Such a conclusion demonstrates the effectiveness of FPL on alleviating the twin fitting problem.

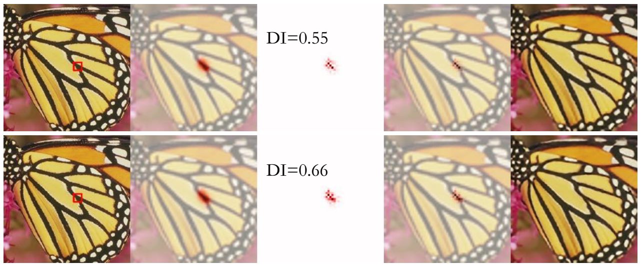


Figure 6. Attribution results with diffusion index (DI) w.r.t. FSRCNN (top) and FSRCNN+FPL (bottom). From left to right respectively are HR images, receptive fields, attribution results, attribution pixels, and SR results. A higher DI represents that more pixels significantly contribute to the SR result of the given patch.

5. Conclusion

In this work, we propose a novel understanding, *i.e.*, the twin fitting problem arises from the long-tailed pixel distribution, to explain why the high-frequency regions are harder to be super-resolved than the low-frequency ones. Inspired by the explanation, we first reformulate SR as a long-tailed distribution learning problem, and design a simple and effective solution by introducing two structure priors and an under-sampling strategy. Such a reformulation and solution enjoy the advantages of high interpretability and model agnostic, and the extensive experiments have demonstrated the superiority of them.

Acknowledgments

This work was supported in part by NSFC under Grant U21B2040, 62176171, and U19A2078; in part by Sichuan Science and Technology Planning Project under Grant 2022YFQ0014. A*STAR AME Programmatic Funding A18A2b0046, RobotHTPO Seed Fund under Project C211518008 and EDB Space Technology Development Grant under Project S22-19016-STDP.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: dataset and study. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, July 2017. 5

[2] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *Eur. Conf. Comput. Vis.*, pages 252–268, 2018. 5

[3] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. 2012. 5

[4] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Adv. Neural Inform. Process. Syst.*, 32, 2019. 2

[5] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *J. Artif. Intell. Res.*, 16:321–357, 2002. 3

[6] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9268–9277, 2019. 2, 3

[7] Dengxin Dai, Yujian Wang, Yuhua Chen, and Luc Van Gool. Is image super-resolution helpful for other vision tasks? In *Winter Conf. Appl. Comput. Vis.*, pages 1–9, 2016. 1

[8] Xin Deng, Ren Yang, Mai Xu, and Pier Luigi Dragotti. Wavelet domain style transfer for an effective perception-distortion tradeoff in single image super-resolution. In *Int. Conf. Comput. Vis.*, pages 3076–3085, 2019. 2

[9] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *Eur. Conf. Comput. Vis.*, pages 184–199, 2014. 2

[10] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *Eur. Conf. Comput. Vis.*, pages 391–407, 2016. 5

[11] Manuel Fritsche, Shuhang Gu, and Radu Timofte. Frequency separation for real-world super-resolution. In *Int. Conf. Comput. Vis. Worksh.*, pages 3599–3608, 2019. 2

[12] Azuma Fujimoto, Toru Ogawa, Kazuyoshi Yamamoto, Yusuke Matsui, Toshihiko Yamasaki, and Kiyoharu Aizawa. Manga109 dataset and creation of metadata. In *Int. Worksh. Comics Anal., Process., Underst.*, pages 1–5, 2016. 5

[13] Yuanbiao Gou, Peng Hu, Jiancheng Lv, and Xi Peng. Multi-scale adaptive network for single image denoising. *Adv. Neural Inform. Process. Syst.*, 2022. 1

[14] Yuanbiao Gou, Boyun Li, Zitao Liu, Songfan Yang, and Xi Peng. Clearer: Multi-scale neural architecture search for image restoration. *Adv. Neural Inform. Process. Syst.*, 33:17129–17140, 2020. 1

[15] Jinjin Gu and Chao Dong. Interpreting super-resolution networks with local attribution maps. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9199–9208, 2021. 8

[16] Shuhang Gu, Wangmeng Zuo, Qi Xie, Deyu Meng, Xiangchu Feng, and Lei Zhang. Convolutional sparse coding for image super-resolution. In *Int. Conf. Comput. Vis.*, pages 1823–1831, 2015. 2

[17] Chunle Guo, Chongyi Li, Jichang Guo, Runmin Cong, Huazhu Fu, and Ping Han. Hierarchical features driven residual learning for depth map super-resolution. *IEEE Trans. Image Process.*, 28(5):2545–2557, 2018. 1

[18] Muhammad Haris, Greg Shakhnarovich, and Norimichi Ukita. Task-driven super resolution: Object detection in low-resolution images. In *Int. Conf. Neural Inform. Process.*, pages 387–395, 2021. 1

[19] Yanting Hu, Nannan Wang, Dacheng Tao, Xinbo Gao, and Xuelong Li. Serf: A simple, effective, robust, and fast image super-resolver from cascaded linear regression. *IEEE Trans. Image Process.*, 25(9):4091–4102, 2016. 1

[20] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5197–5206, 2015. 5

[21] Yawen Huang, Ling Shao, and Alejandro F Frangi. Simultaneous super-resolution and cross-modality synthesis of 3d medical images using weakly-supervised joint convolutional sparse coding. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6070–6079, 2017. 1

[22] Muhammad Abdullah Jamal, Matthew Brown, Ming-Hsuan Yang, Liqiang Wang, and Boqing Gong. Rethinking class-balanced methods for long-tailed visual recognition from a domain adaptation perspective. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7610–7619, 2020. 2

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

[24] Xiangtao Kong, Hengyuan Zhao, Yu Qiao, and Chao Dong. Classsr: A general framework to accelerate super-resolution networks by data characteristic. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 12016–12025, 2021. 5

[25] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4681–4690, 2017. 1, 5

[26] Boyun Li, Yuanbiao Gou, Shuhang Gu, Jerry Zitao Liu, Joey Tianyi Zhou, and Xi Peng. You only look yourself: Unsupervised and untrained single image dehazing neural network. *Int. J. Comput. Vis.*, 129:1754–1767, 2021. 1

[27] Boyun Li, Yuanbiao Gou, Jerry Zitao Liu, Hongyuan Zhu, Joey Tianyi Zhou, and Xi Peng. Zero-shot image dehazing. *IEEE Trans. Image Process.*, 29:8457–8466, 2020. 1

[28] Buyu Li, Yu Liu, and Xiaogang Wang. Gradient harmonized single-stage detector. In *AAAI*, volume 33, pages 8577–8584, 2019. 3

[29] Juncheng Li, Faming Fang, Kangfu Mei, and Guixu Zhang. Multi-scale residual network for image super-resolution. In *Eur. Conf. Comput. Vis.*, pages 517–532, 2018. 5

[30] Zheyuan Li, Yingqi Liu, Xiangyu Chen, Haoming Cai, Jinjin Gu, Yu Qiao, and Chao Dong. Blueprint separable residual network for efficient image super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 833–843, 2022. 5, 8

[31] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Int. Conf. Comput. Vis.*, pages 1833–1844, 2021. 1, 2, 5

[32] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Int. Conf. Comput. Vis.*, pages 2980–2988, 2017. 2, 3

[33] Risheng Liu, Xiangyu Wang, Xin Fan, Haojie Li, and Zhongxuan Luo. Deep hybrid residual learning with statistic priors for single image super-resolution. In *Int. Conf. Multimedia and Expo*, pages 1111–1116, 2017. 2

[34] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2537–2546, 2019. 2

[35] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 5

[36] Zhisheng Lu, Hong Liu, Juncheng Li, and Linlin Zhang. Efficient transformer for single image super-resolution. *arXiv preprint arXiv:2108.11084*, 2021. 1, 2

[37] Cheng Ma, Yongming Rao, Jiwen Lu, and Jie Zhou. Structure-preserving image super-resolution. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021. 1, 2

[38] 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 *Int. Conf. Comput. Vis.*, volume 2, pages 416–423, 2001. 5

[39] Ben Niu, Weilei Wen, Wenqi Ren, Xiangde Zhang, Lianping Yang, Shuzhen Wang, Kaihao Zhang, Xiaochun Cao, and Haifeng Shen. Single image super-resolution via a holistic attention network. In *Eur. Conf. Comput. Vis.*, pages 191–207, 2020. 1

[40] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Adv. Neural Inform. Process. Syst.*, 32, 2019. 5

[41] Minlong Peng, Qi Zhang, Xiaoyu Xing, Tao Gui, Xuanjing Huang, Yu-Gang Jiang, Keyu Ding, and Zhigang Chen. Trainable undersampling for class-imbalance learning. In *AAAI*, volume 33, pages 4707–4714, 2019. 3

[42] Dehua Song, Chang Xu, Xu Jia, Yiyi Chen, Chunjing Xu, and Yunhe Wang. Efficient residual dense block search for image super-resolution. In *AAAI*, volume 34, pages 12007–12014, 2020. 2

[43] Longguang Wang, Xiaoyu Dong, Yingqian Wang, Xinyi Ying, Zaiping Lin, Wei An, and Yulan Guo. Exploring sparsity in image super-resolution for efficient inference. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 4917–4926, June 2021. 1

[44] Shizun Wang, Ming Lu, Kaixin Chen, Jiaming Liu, Xiaoqi Li, chuang Zhang, and Ming Wu. Samplingaug: On the importance of patch sampling augmentation for single image super-resolution. In *Brit. Mach. Vis. Conf.*, 2021. 2

[45] Wenxin Wang, Boyun Li, Yuanbiao Gou, Peng Hu, and Xi Peng. Relationship quantification of image degradations. *arXiv preprint arXiv:2212.04148*, 2022. 1

[46] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella Yu. Long-tailed recognition by routing diverse distribution-aware experts. In *Int. Conf. Learn. Represent.*, 2021. 2

[47] Bihan Wen, Ulugbek S Kamilov, Dehong Liu, Hassan Mansour, and Petros T Boufounos. Deepcasd: An end-to-end approach for multi-spectral image super-resolution. In *ICASSP*, pages 6503–6507, 2018. 1

[48] Gang Wu, Junjun Jiang, Xianming Liu, and Jiayi Ma. A practical contrastive learning framework for single image super-resolution. *arXiv preprint arXiv:2111.13924*, 2021. 1, 2

[49] Xiangyu Xu, Deqing Sun, Jinshan Pan, Yujin Zhang, Hanspeter Pfister, and Ming-Hsuan Yang. Learning to super-resolve blurry face and text images. In *Int. Conf. Comput. Vis.*, pages 251–260, 2017. 1

[50] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE Trans. Image Process.*, 19(11):2861–2873, 2010. 5

[51] Zizheng Yang, Mingde Yao, Jie Huang, Man Zhou, and Feng Zhao. Sir-former: Stereo image restoration using transformer. In *ACM Int. Conf. Multimedia*, pages 6377–6385, 2022. 1

[52] Mingde Yao, Dongliang He, Xin Li, Fu Li, and Zhiwei Xiong. Towards interactive self-supervised denoising. *IEEE Trans. Circuit Syst. Video Technol.*, 2023. 1

[53] Mingde Yao, Dongliang He, Xin Li, Zhihong Pan, and Zhiwei Xiong. Bidirectional translation between uhd-hdr and hd-sdr videos. *IEEE Trans. Multimedia*, 2023. 1

[54] Liangpei Zhang, Hongyan Zhang, Huanfeng Shen, and Pingxiang Li. A super-resolution reconstruction algorithm for surveillance images. *Signal Process.*, 90(3):848–859, 2010. 1

[55] Songyang Zhang, Zeming Li, Shipeng Yan, Xuming He, and Jian Sun. Distribution alignment: A unified framework for long-tail visual recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2361–2370, 2021. 2

[56] Yifan Zhang, Bryan Hooi, Lanqing Hong, and Jiashi Feng. Self-supervised aggregation of diverse experts for test-agnostic long-tailed recognition. In *Adv. Neural Inform. Process. Syst.*, 2022. 2

[57] Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep long-tailed learning: A survey. *arXiv preprint arXiv:2110.04596*, 2021. 2

[58] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2472–2481, 2018. 1, 2

[59] Tianyu Zhao, Wenqi Ren, Changqing Zhang, Dongwei Ren, and Qinghua Hu. Unsupervised degradation learning for single image super-resolution. *arXiv preprint arXiv:1812.04240*, 2018. 1