LSDIR: A Large Scale Dataset for Image Restoration

Yawei Li[1] Kai Zhang[1] Jingyun Liang[1] Jiezhang Cao[1]
Ce Liu[1] Rui Gong[1] Yulun Zhang[1] Hao Tang[1] Yun Liu[1]
Denis Demandolx[2] Rakesh Ranjan[2] Radu Timofte[1,3] Luc Van Gool[1,4]
[1]Computer Vision Lab, ETH Zürich [2]Meta Reality Labs [3]University of Würzburg [4]KU Leuven
https://data.vision.ee.ethz.ch/yawli/index.html

Abstract

The aim of this paper is to propose a large scale dataset for image restoration (LSDIR). Recent work in image restoration has been focused on the design of deep neural networks. The datasets used to train these networks 'only' contain some thousands of images, which is still incomparable with the large scale datasets for other vision tasks such as visual recognition and object detection. The small training set limits the performance of image restoration networks. To solve this problem, we collect high-resolution (HR) images from Flickr for image restoration. To ensure the pixel-level quality of the collected dataset, annotators were invited to manually inspect each of the collected image and remove the low-quality ones. The final dataset contains 84,991 high-quality training images, 1,000 validation images, and 1,000 test images. In addition, we showed that the model capacity of large networks could be fully exploited by training on the large scale dataset with significantly increased patch size and prolonged training iterations. The experimental results on image super-resolution (SR), denoising, JPEG deblocking, deblurring, and demosaicking, and real-world SR show that image restoration networks benefit a lot from the large scale dataset.

1. Introduction

Image restoration (IR) refers to the problem of recovering high-quality images from degraded images which are derived by a degradation model. Depending on the degradation model, the problem could be classified into several sub-problems such as image super-resolution (SR) [9, 11, 17, 22, 37, 44, 48, 51], image denoising [6, 28, 73, 79, 86, 93], image deblurring [47, 50, 59, 68], image demosaicing [29, 56, 80, 84, 95, 96], JPEG compression artifacts removal [13, 16, 33, 93], raindrop removal [23, 39, 42, 45], haze removal [4, 30, 40, 65, 66], and so on. Due to the different characteristics of those problems, resulting from the different degradation models, those problems were tackled independently by specifically designed algorithms before the deep learning era such as sparse coding and dictionary learning for image super-resolution [70, 71, 82, 83], filtering methods for image denoising [6, 14, 19], bidirectional interpolation methods for image demosaicking [95, 96]. During the past ten years, with the fast development of deep learning theories and computing hardware, deep neural networks emerged as a generalizable solution to those problems [11, 48, 86, 91, 100]. An indispensable component for success - at least thus far - is the adoption of supervised learning. Deep neural networks learn to restore the degraded image by utilizing the underlying mappings between paired degraded and high-quality images. Yet, collecting a real-world paired dataset for image restoration is extremely difficult, which usually involves paired camera setup, post-processing pipelines to estimate the ground truth, alignment between image pairs and so on [1, 7, 10, 36, 49, 61, 63]. As an alternative, the classical method to get paired data for image restoration is by data synthesis [54, 75, 90, 92]. The degraded images are derived by applying typical and synthetic degradation processes to the ground-truth images. The ground-truth images are carefully selected to exclude undesired artifacts such as blur and noise corruption and to ensure a good coverage of natural textures [3, 26, 51, 76, 78].

Most previous research in image restoration focuses on the design of deep models. Their performance increases as the networks get deeper and more complicated [17, 35, 37, 48, 51, 99, 100]. With the fast development of image restoration models, it becomes crucial to reconsider data based on the following facts. *First of all,* it is well-known that big models are data hungry [18, 53, 64, 72]. Training with large-scale data could improve the prediction accuracy of deep neural networks. For image restoration tasks, training with large-scale dataset leads to better performance for both small [17, 41] and large image restoration networks [11, 12]. *Second*, current IR benchmark datasets date back to decades ago. Limited by the imaging technique, the image quality of the old benchmark datasets are not well guaranteed (Fig. 1).

(a) Example image from the proposed dataset

(b) Example image from McMaster

(c) Example image from B100

(d) Example image from Set 5 and Set12

(e) Example image from Set14

Figure 1. Example images from different image restoration datasets. Images in the proposed dataset are in high resolution, of high quality, and with detailed contents.

Thus, there could be a domain gap between the training and test images, which could not faithfully reveal the performance of IR models. *Third*, training with synthetic data is not just out of research interest. Instead, it helps the image restoration model to learn priors under simplified setting. Recent works have also shown the great potential of purely utilizing data synthesis for real-world image restoration [75, 90, 92]. This removes the reliance on real-world data for real-world image restoration.

In this paper, we aim at building a large scale high-resolution dataset for image restoration tasks. Earlier datasets contain a limited number of images. For example, the most commonly used dataset DIV2K [3, 46, 88] contains 800 training images and 100 validation images. Fickr2K offers another 2,650 2K resolution images [51]. Those datasets have been used as the standard training sets for image restoration during the past five years. Recently, ImageNet [15] has also been used to train image restoration networks [11, 41]. Yet, since ImageNet is not collected specifically for image restoration, it comes with some undesirable artifacts such as noise and blur, both detrimental for image restoration. In addition, the resolution of the images is usually quite low. With the development of ever larger deep learning models (especially transformers [11, 12, 41, 48]) and large-scale datasets for high-level vision tasks (ImageNet [15], MSCOCO [52]) and vision-language tasks (YFCC100M [69], WIT [64]), a new large-scale dataset for image restoration is called for.

In response, we built a large-scale dataset for image restoration. It is composed of 84,991 training images, 1,000 validation images and 1,000 test images. To ensure the dataset's diversity, over 20,000 keywords were used to search for the images automatically. All images in the dataset are carefully checked to ensure the image quality. Beyond that, we also investigated the characteristics and benefits of training with this large scale dataset. We can summarize our conclusions and findings as follows. 1) The benefit of the large scale dataset is that it enables to fully exploit the capacity of large models with significantly increased patch size and prolonged training iterations. 2) By contrast, overfitting is observed when training with the previous small scale datasets DIV2K and DF2K (See Fig. 3). 3) Large models get a significant performance boost with the proposed dataset. For example, EDSR [51] achieves performances comparable with or even better than those reported in the SwinIR paper [48].

2. Related Work

Image restoration methods. Traditional image restoration methods includes the model-based [6, 8, 14, 20, 27, 28, 62, 95] methods and the exampled-based methods [9, 22, 67, 70, 71, 82]. With the thriving of deep learning, deep neural networks are used to deal with image restoration problems including image SR [17, 35, 37, 43, 44, 46, 51, 88, 89, 94], image denoising [2, 24, 74, 81, 91, 93], JPEG artifacts removal [48, 100] and so on. Recently, generalized deep neural networks have been developed to deal with image restoration tasks jointly [11, 41, 48, 100]. Among the image restoration networks, there is a trend that the network becomes deeper and more complicated. Most of the previous work focuses on the design of neural network. However, the training dataset is not paid enough attention to.

Image restoration datasets. Learning based image restoration methods rely on the external training dataset to learn the mapping between degraded and ground-truth images. Early methods uses small datasets such as the 91 images proposed in [82] and the 400 images from BSD dataset [57]. Since the resolution of those images is not high enough, DIV2K and Flickr2K datasets are proposed which contain 800 and 2,650 2K resolution training images respec-

tively. Yet, compared with datasets such as ImageNet [15] and MSCOCO [52], the diversity of the contents in the image restoration datasets is limited. In some work, ImageNet is also used to train image restoration networks. But the resolution of images in ImageNet is not very high (about 256×256) and the pixel-level quality of the images is not guaranteed. Thus, a much larger image restoration dataset with diverse content is called for. A couple of other datasets are also often used for image restoration tasks including FFHQ [34], WED [55], OST [76], SCUT-CTW1500 [85], and DIV8K [26]. Besides the training dataset, there are also a couple of test sets used to benchmark different image restoration methods, which includes Set5 [5], Set14 [87], BSD [57], Kodak24 [21], McMaster [96], Urban100 [31], Manga109 [58], DIV2K validation set [3] and so on.

3. Data Collection Pipeline

We introduce the LSDIR dataset, a new Large Scale Dataset for Image Restoration. Examples of the images in this dataset are shown in Fig. 2. The data collection pipeline is introduced in this section.

3.1. Blur and Noise Suppression

Different from high-level vision tasks such as visual recognition and object detection, it is important to ensure the pixel-level quality of the images in the restoration dataset. The major challenge for its collection is the variety of possible artifacts. The most commonly occurring artifacts in natural images are blur and noise corruption, which need to be suppressed during data collection [3,26,51]. Yet, blur detection and noise estimation are already non-trivial problems in their own right. Instead, we adopt multiple strategies to suppress the artifacts. First, during data crawling, we use a simple and efficient blur detection method via the variance of the image Laplacian [60]. Second, to suppress artifacts such as noise and JPEG image compression artifacts, the crawled original images are down-sampled. Third, the collected images are labelled by human annotators. Only images that pass two rounds of human inspection are labelled as high quality and are kept in the dataset. Additionally, we also use a simple method to detect images with too many flat regions and exclude those from the dataset. The blur detection and flat region detection are described next.

Blur detection. To detect blurry images, we use the variance of the Laplacian [60] as an indicator.

$$L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}, \tag{1}$$

$$\mathcal{S}_{blur} = \mathrm{Var}(L(x,y)), \tag{2}$$

where I denotes the input image, (x, y) are the coordinates of a pixel, and $\mathcal{S}_{blur}$ is the blur score. A small score $\mathcal{S}_{blur}$ means that the image is blurred. On the other hand, a large score often corresponds to noisy images. Thus, we only select images with score $\mathcal{S}_{blur}$ in the range $[\mathcal{S}^l_{blur}, \mathcal{S}^h_{blur}]$. The two thresholds $\mathcal{S}^l_{blur}$ and $\mathcal{S}^h_{blur}$ are empirically set to 150 and 8,000.

Flat region detection. To detect images dominated by flat regions, we develop a voting based method. Specifically, the whole image is divided into non-overlapping patches with patch size 240×240. The score for flat region detection is calculated by the Sobel filter, a good edge detector. The assumption about a flat region is that it does not contain a lot of edges. Thus, the variance of the magnitude of the image derivative along the two axis is used as indicator of the flatness of a region, namely,

$$G(x,y) = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2}, \tag{3}$$

$$\mathcal{S}_{flat} = \mathrm{Var}(G(x,y)). \tag{4}$$

If the score $\mathcal{S}_{flat}$ of an image patch is lower than a threshold $\mathcal{S}^t_{flat}$, then the patch is classified as a flat region. We empirically set the score $\mathcal{S}^t_{flat}$ to 800. If the percentage of flat patches in an image is more than 50%, then the image is classified as a flat image and is excluded from the dataset.

3.2. Data Collection

Source. We automatically crawled images from Flickr[1]. Data crawling is conducted by keyword search. Searching by different keywords via Flickr API can return scenes with quite different content and texture. Thus, to collect the large-scale dataset and guarantee the diversity of the images, we used a large set of keywords split into four subsets including *flickr2k*, *flickr_tag*, *imagenet*, and *imagenet_21k*. *1) flickr2k:* This subset contains the 133 queries used to create the Flickr2K dataset [51]. *2) flickr_tag*: This subset is collected by us and contains the hottest 200 tags and their related tags on the Flickr website. *3) imagenet & 4) imagenet_21k*: Additionally, we also used the 1,000 labels from ImageNet, and the 21,843 labels from ImageNet 21K. Note that we only used the labels from ImageNet while the images from ImageNet are not used nor included in the collected dataset. Repetitions of search keywords in the four subsets are removed.

Data selection criteria. During the data crawling phase, we iterate several rounds to validate the quality and content of the collected images. During each round, 1,000 images are collected and their quality and content are inspected. Depending on that, a set of automatic image selection criteria is deployed to ensure the quality of the images. The images are selected according to the license, resolution, aspect ratio, captured date, and tags. We also found out that the

[1] https://www.flickr.com/

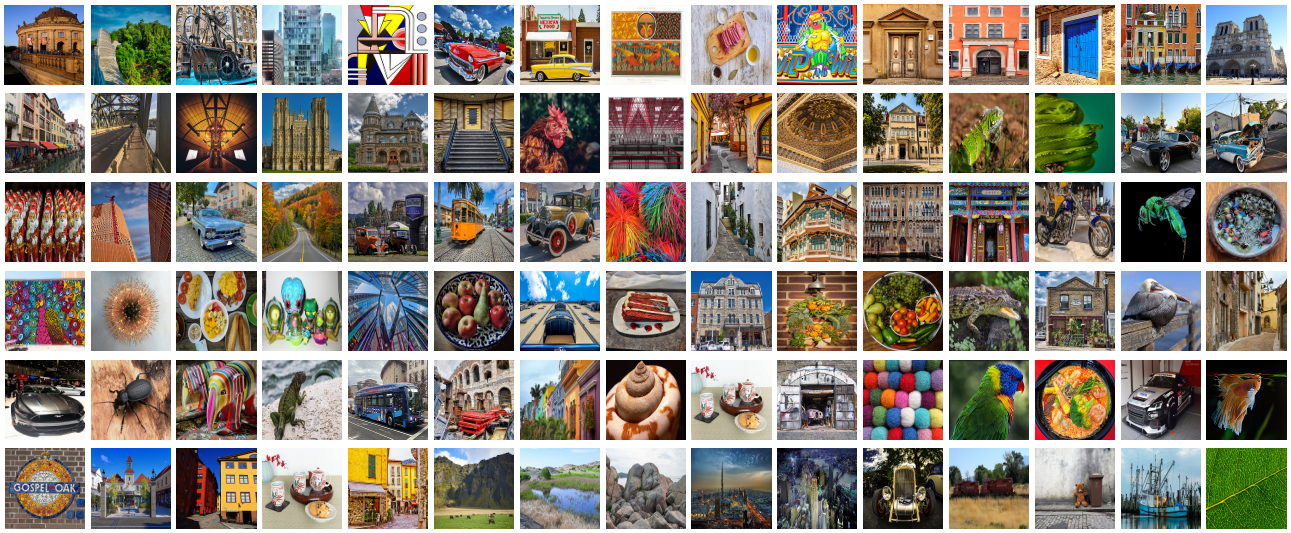


Figure 2. Examples of validation images from the LSDIR dataset.

most common artifact in the original images is blur. In addition, high-resolution images with flat regions occur very frequently, which defies our aim of providing diverse contents in the dataset. Thus, to solve those two issues, we use a blur detector and a flat region detector to filter the images. This filtering process guarantees a high quality for most images.

Data storage. Each image that passes the automatic selection criteria and its meta information are stored. The original images are cropped to a multiple of 48 pixels on both sides. This removes the additional cropping operation for image SR with different up-scaling factors such as $\times 2$, $\times 3$, $\times 4$. The meta information of the image includes the search keyword for the image, the download link, the camera information, the captured date, the resolution of the image, the blur score $\mathcal{S}_{blur}$, the percentage of flat patches in the image, etc. The rich meta information leaves the door open to other uses of the dataset. Since the camera information is provided in the meta data, one example use of this information is to study image translation between different cameras.

3.3. Image Quality Control

During the data crawling phase, 4K resolution images are collected from Flickr. To ensure the quality of the collected images, the quality of each image is manually checked following the procedure given below.

Annotator. Human annotators are asked to check the quality of the images in the collected dataset.

Selection criteria. A high quality standard is set for the quality inspection. The human annotators are required to do a binary classification of high-quality and low-quality images. The high-quality images are the sharp high-resolution images without blur and noise. For quality control, two major types of classification errors are considered including *false positive* and *false negative*. False positive means that a low-quality image is classified as a high-quality one while false negative means a high-quality one classified as low-quality. Since the aim of the quality inspection is to guarantee the quality of the images in the remaining dataset, we are more tolerant to false negative than false positive. Based on this philosophy, we design the two rounds of manual selection to guarantee both image quality and productivity of the process. The first round of inspection provides a coarse selection. Due to the high standard of the image quality and the tolerance to false negative, the annotators could go through the images quickly and remove most (more than 70%) of the images. After the first round, most of the low-quality images are removed. Then during the second round inspection, the annotators need to check the remaining images. And the aim of this round is to remove the false positive images.

Inspection Tool. To facilitate the manual inspection, an inspection tool was developed. This tool automatically finds the path of all the images in a folder. Then the tool loads the images one by one. The user needs to determine whether the current image is of high quality or not. The tool improves the productivity of the manual inspection significantly. The time spent for manual inspection and more information about the inspection tool are given in the supplementary.

Quality check. After the manual inspection phase, the remaining images are divided into nine parts. As different human annotators might have their own image quality standards and we are more tolerant to the false negative, the remaining number of images in each part is different. To check the image quality in the nine parts, we compare the image restoration performance by training networks with

Table 1. Quality check of the different parts. PSNR reported on Urban100 for image SR with upscaling factor $\times 4$.

Partition	Part1	Part2	Part3	Part4	Part5	Part6	Part7	Part8	Part9
Num. images	13587	2862	15923	4880	13193	7129	8655	12621	6141
MSRResNet [77]	26.01	26.00	26.01	26.01	26.00	25.98	26.00	25.99	26.01
EDSR [51]	27.15	26.70	27.14	26.86	27.09	26.99	26.98	27.09	26.96

Table 2. Investigation of downsampling schemes used to suppress blur and noise artifacts. Image SR results. The upscaling factor is $\times 4$.

Method	Dataset	Set 5 PSNR/SSIM	Set 14 PSNR/SSIM	BSD100 PSNR/SSIM	Urban100 PSNR/SSIM	Manga109 PSNR/SSIM	DIV2K Val. PSNR/SSIM
IMDN [32]	LSDIR_X2	32.10 / 0.885	28.60 / 0.769	27.57 / 0.725	26.08 / 0.776	30.51 / 0.898	30.38 / 0.827
	LSDIR_X4	32.02 / 0.884	28.60 / 0.769	27.56 / 0.725	26.13 / 0.778	30.44 / 0.897	30.37 / 0.827
	LSDIR_2160	32.10 / 0.885	28.61 / 0.769	27.57 / 0.724	26.05 / 0.775	30.59 / 0.898	30.38 / 0.827
EDSR [51]	LSDIR_X2	32.66 / 0.892	29.00 / 0.778	27.81 / 0.734	27.11 / 0.807	31.68 / 0.911	30.86 / 0.838
	LSDIR_X4	32.66 / 0.892	28.99 / 0.779	27.81 / 0.734	27.22 / 0.810	31.53 / 0.911	30.87 / 0.839
	LSDIR_2160	32.62 / 0.891	29.00 / 0.779	27.82 / 0.734	27.13 / 0.808	31.71 / 0.911	30.86 / 0.838
SwinIR [48]	LSDIR_X2	32.83 / 0.895	29.11 / 0.782	27.91 / 0.738	27.70 / 0.823	32.09 / 0.917	31.08 / 0.843
	LSDIR_X4	32.79 / 0.895	29.16 / 0.784	27.91 / 0.739	27.83 / 0.826	32.03 / 0.917	31.10 / 0.844
	LSDIR_2160	32.83 / 0.895	29.14 / 0.783	27.91 / 0.738	27.71 / 0.823	32.20 / 0.917	31.07 / 0.843

them separately. MSRResNet [77] and EDSR [51] are used as baseline networks. Observing Tab. 1, we see that the difference in the validation accuracy on Urban100 is largely due to the number of images. Thus, by this investigation, we can safely conclude that *the image quality of all the nine parts after quality inspection is good.* All nine parts are used in the later experiments.

3.4. Post Processing

As mentioned in Sec. 3.1, the original images are down-sampled to suppress the undesired artifacts. The Lanczos resampling method is used as down-sampling method to keep the high-frequency information as much as possible. The down-sampling factor needs to be chosen carefully to trade off noise suppression and detail preservation. If it is too small, the artifacts will not be removed. Yet, if it is too large, too much detail will be lost. For the older DIV2K dataset [3], the down-sampling factor is manually determined depending on the image content. For Flickr2K, the images are down-sampled to a fixed maximum dimension 2,040 [51]. To determine a proper down-sampling factor, we used three down-sampling schemes: down-sampling by a factor of 2, by a factor of 4, or such that the maximum image dimension is 2,160 pixels. And the corresponding versions of the dataset are represented by LSDIR_X2, LSDIR_X4, and LSDIR_2160, respectively. We investigate the down-sampling schemes based on empirical studies for image SR and grayscale image denoising. The experimental results for image SR are shown in Tab. 2 and the supplementary, respectively.

The final down-sampling scheme is determined by joint consideration of the artifact suppression effect, detail preservation, and the objective image restoration performance in terms of PSNR and SSIM. First, LSDIR_X2 is not the preferred version. One reason is that by analyzing the experimental results LSDIR_X2 could be outperformed by either LSDIR_X4, or LSDIR_2160. In addition, the small down-sampling factor might not have a good effect in noise suppression. Second, both LSDIR_X4 and LSDIR_2160 can be used for image restoration. Depending on the validation set in Tab. 2, either LSDIR_X4 or LSDIR_2160 could lead to higher accuracy. One property of LSDIR_2160 is that the down-sampling factor for images with different resolutions could be adjusted. Thus, different image content and information condensation could be achieved for images with different resolutions. But the disadvantage is that the down-sampling factors of most images are smaller than 4. Thus, the required storage space of LSDIR_2160 is larger than LSDIR_X4. Finally, during the quality inspection phase, the original images are also down-sampled by a factor of 4 to suppress noise. The manual inspection procedure might cause that the quality of remaining images are implicitly fitted to that setting. Thus, considering all the factors, we will use LSDIR_X4 for the following experiments.

3.5. Validation and Test Sets Split

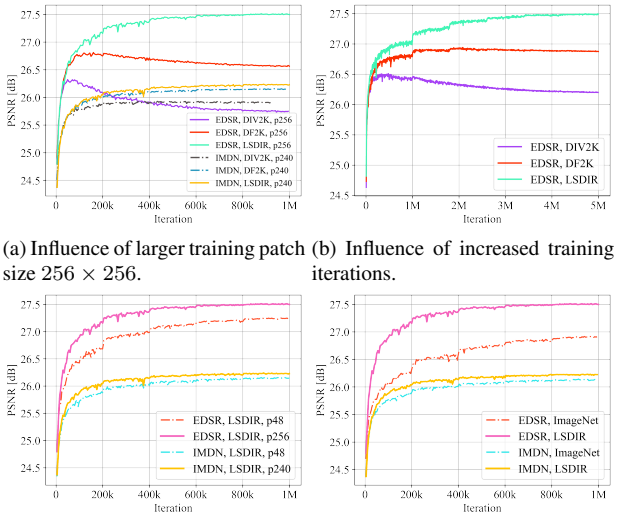
Apart from the 84,991 images in the training set, there are additionally 2,150 images used for validation and testing. Since the images contain different contents, some images might be more challenging when used for image restoration tasks. Thus, the 2,150 images should be properly split in order to maintain a balanced performance between the validation and test set. In Tab. 3, the 2,150 images are divided into 9 splits with 250 images in the first 8 splits and 150 images in the last split. The validation accuracy on the 9 splits is reported for 3 image SR networks and 2 image denoising networks. According to the validation results, the splits S1, S4, S6, S7 constitute the validation set while the splits S2, S3, S5, S8 constitute the test set. Split $S9$ is not included in either of the two sets. The validation set will be released along with the training set. The test set will be kept

Table 3. Validation and test set split. Upscaling factor is $\times 4$ for SR. Noise level is 15 for denoising.

Method	Split									Val. Set	Test Set
	S1	S2	S3	S4	S5	S6	S7	S8	S9		
IMDN [32]	26.23	26.88	26.69	26.37	25.94	27.13	26.31	26.46	27.20	26.51	26.49
EDSR [51]	26.77	27.40	27.23	26.84	26.43	27.69	26.80	26.91	27.76	27.03	26.99
SwinIR [48]	26.96	27.62	27.44	27.02	26.62	27.91	26.99	27.08	28.00	27.22	27.19
DnCNN [93]	31.90	32.28	32.17	31.88	31.53	32.29	31.79	31.65	32.33	31.97	31.91
DRUNet [91]	32.25	32.65	32.53	32.19	31.85	32.64	32.10	31.97	32.68	32.29	32.25

Table 4. Influence of training patch size on the performance of image SR and denoising networks. The networks are trained with the proposed LSDIR dataset. PSNR and SSIM reported on Urban100.

Method	Patch size	BSD100 / BSD68 PSNR/SSIM	Urban100 PSNR/SSIM	DIV2K PSNR/SSIM	LSDIR Val. PSNR/SSIM	LSDIR Test PSNR/SSIM
IMDN [32]	48×48	27.56 / 0.725	26.13 / 0.778	30.37 / 0.827	26.51 / 0.741	26.49 / 0.739
	240×240	27.58 / 0.726	26.22 / 0.781	30.41 / 0.828	26.55 / 0.743	26.53 / 0.741
EDSR [51]	48×48	27.81 / 0.734	27.22 / 0.810	30.87 / 0.839	27.03 / 0.761	26.99 / 0.760
	256×256	27.88 / 0.737	27.48 / 0.818	30.99 / 0.841	27.15 / 0.766	27.11 / 0.764
DnCNN [93]	40×40	31.68 / 0.883	32.77 / 0.921	33.76 / 0.901	31.97 / 0.906	31.91 / 0.906
	256×256	31.70 / 0.883	32.87 / 0.922	33.79 / 0.901	32.00 / 0.907	31.95 / 0.906
DRUNet [91]	128×128	31.89 / 0.888	33.50 / 0.931	34.11 / 0.908	32.29 / 0.913	32.25 / 0.913
	256×256	31.91 / 0.888	33.59 / 0.932	34.15 / 0.908	32.34 / 0.914	32.29 / 0.914



(a) Influence of larger training patch size 256×256.

(b) Influence of increased training iterations.

(c) Comparison between training patch sizes.

(d) Comparison between training with LSDIR and ImageNet.

Figure 3. Comparison of validation accuracy on the Urban100 dataset between different settings. The experiments are done for image SR with up-scaling factor $\times 4$. 'p*' denotes the patch size.

for benchmarking purposes.

4. Experimental Results and Analysis

The experimental results are reported in this section. We first analyze the importance of the two training hyperparameters patch size and number of iterations, when training with our large dataset. Then we compare the proposed dataset with three commonly used datasets including DIV2K [3], DF2K (DIV2K and Flickr2K), and ImageNet 1K [15]. Experiments are carried out on 6 image restoration tasks including *image SR*, *grayscale and color image denoising*, *image demosaicking*, *image deblurring with Gaussian kernels and real-world kernels* [38], *color and grayscale image JPEG compression artifact removal*, and *real-world image denoising*. During training, patches are extracted from the image to make up a mini-batch. Note that the patch size for image SR is the size of the LR patches. The training detail for different networks is given in the supplementary material. *All results in this paper are derived by training the network from scratch without any pretraining.* PSNR, SSIM, and LPIPS [97] are used as the evaluation metrics.The PSNR values are only calculated for the luminance channel. We conducted experiments on a bunch of image restoration networks including IMDN [32], EDSR [51], MSRResNet [77], SwinIR [48], SGN [25], DnCNN [93], DRUNet [91], Uformer [79], Restormer [86], RDN [100], RNAN [98], and BSRGAN [92].

Table 5. Study on training dataset size. PSNR reported on Urban100 for $\times 4$ SR.

Percentage	1%	5%	10%	20%	100%
PSNR	26.04	26.85	27.05	27.16	27.23

4.1. Benefits of A Large Scale Training Set

Influence of patch size. In Fig. 3a, the results of training with larger patch sizes are shown for EDSR and IMDN. It is clear that the heavyweight network EDSR overfits the training set DIV2K and DF2K when the training patch size is enlarged. On the other hand, with LSDIR, the accuracy keeps growing as the training continues. The overfitting phenomenon is easy to understand. When the patch size is very large, the number of actual useful patches gets smaller. Since both the DIV2K and DF2K datasets have 2K reso-

Table 6. Comparison between DIV2K, DF2K, and LSDIR for image SR, image denoising and image demosaicking.

(a) Image SR results. The upscaling factor is $\times 4$.

Method	Dataset	Set 5 PSNR/SSIM	Set 14 PSNR/SSIM	BSD100 PSNR/SSIM	Urban100 PSNR/SSIM	Manga109 PSNR/SSIM	DIV2K Val. PSNR/SSIM	LSDIR Val. PSNR/SSIM
EDSR [51]	DIV2K	32.32 / 0.889	28.68 / 0.773	27.65 / 0.730	26.48 / 0.791	30.74 / 0.904	30.58 / 0.833	26.62 / 0.750
111.79 ms	DF2K	32.62 / 0.891	28.91 / 0.778	27.80 / 0.733	26.93 / 0.803	31.50 / 0.909	30.81 / 0.838	26.89 / 0.757
43.09 M	LSDIR	32.72 / 0.893	29.08 / 0.782	27.88 / 0.737	27.48 / 0.818	31.85 / 0.914	30.99 / 0.841	27.15 / 0.766
SwinIR [48]	DIV2K	32.68 / 0.893	28.86 / 0.777	27.75 / 0.733	26.71 / 0.799	31.22 / 0.910	30.76 / 0.837	26.58 / 0.753
204.66 ms	DF2K	32.86 / 0.895	29.07 / 0.782	27.90 / 0.738	27.40 / 0.816	32.00 / 0.916	31.03 / 0.842	27.09 / 0.765
11.9 M	LSDIR	32.79 / 0.895	29.16 / 0.784	27.91 / 0.739	27.83 / 0.826	32.03 / 0.917	31.10 / 0.844	27.22 / 0.769

(b) Gray image denoising results. The noise level is 15.

Method	Dataset	Set 12 PSNR/SSIM	BSD68 PSNR/SSIM	Urban100 PSNR/SSIM	DIV2K Val. PSNR/SSIM	LSDIR Val. PSNR/SSIM
DnCNN [93]	DIV2K	32.87 / 0.890	31.70 / 0.883	32.82 / 0.922	33.81 / 0.902	31.99 / 0.907
2.75 ms	DF2K	32.88 / 0.889	31.69 / 0.882	32.74 / 0.920	33.76 / 0.900	31.94 / 0.906
0.56 M	LSDIR	32.87 / 0.890	31.70 / 0.883	32.89 / 0.922	33.79 / 0.901	32.01 / 0.907
DRUNet [91]	DIV2K	33.21 / 0.897	31.89 / 0.887	33.42 / 0.929	34.12 / 0.908	32.28 / 0.913
9.02 ms	DF2K	33.30 / 0.899	31.92 / 0.889	33.49 / 0.931	34.13 / 0.908	32.30 / 0.913
32.64 M	LSDIR	33.26 / 0.898	31.91 / 0.888	33.59 / 0.932	34.15 / 0.908	32.34 / 0.914

(c) Color image denoising results. The noise level is 15.

Method	Dataset	Set 12 PSNR/SSIM/LPIPS	BSD68 PSNR/SSIM/LPIPS	Urban100 PSNR/SSIM/LPIPS
Uformer [79]	DIV2K	37.04 / 0.9497 / 0.1561	36.24 / 0.9512 / 0.1378	37.25 / 0.9658 / 0.1054
35.12 ms	DF2K	37.11 / 0.9506 / 0.1488	36.26 / 0.9514 / 0.1342	37.27 / 0.9660 / 0.0998
20.63 M	LSDIR_X4	37.13 / 0.9507 / 0.1434	36.26 / 0.9515 / 0.1356	37.34 / 0.9664 / 0.1017
Restormer [86]	DIV2K	37.10 / 0.9500 / 0.1636	36.11 / 0.9477 / 0.1405	37.35 / 0.9656 / 0.1112
77.98 ms	DF2K	37.20 / 0.9511 / 0.1455	36.14 / 0.9480 / 0.1346	37.43 / 0.9661 / 0.1019
26.13 M	LSDIR_X4	37.24 / 0.9514 / 0.1600	36.15 / 0.9482 / 0.1419	37.50 / 0.9664 / 0.1030

(d) Image demosaicking results.

Method	Dataset	McMaster PSNR/SSIM/LPIPS	Kodak PSNR/SSIM/LPIPS	Urban100 PSNR/SSIM/LPIPS	LSDIR Val. PSNR/SSIM/LPIPS
RNAN [98]	DIV2K	45.23 / 0.9900 / 0.0274	47.64 / 0.9954 / 0.0181	44.40 / 0.9932 / 0.0164	43.49 / 0.9852 / 0.0219
	DF2K	45.25 / 0.9900 / 0.0277	47.70 / 0.9954 / 0.0182	44.67 / 0.9937 / 0.0152	43.51 / 0.9853 / 0.0219
	LSDIR_X4	45.45 / 0.9905 / 0.0265	47.73 / 0.9954 / 0.0183	44.65 / 0.9939 / 0.0158	43.70 / 0.9858 / 0.0216

lution, the number of useful patches is reduced to several thousands. Those small sets of patches are repeatedly used during the 1 million training iterations with mini-batch size 32. Therefore, it is very easy for the large network to overfit to the training set. Two other observations back the analysis of the reason for overfitting. First, compared with DF2K, the overfitting effect is severer on the smaller DIV2K which contains only 800 training images. Second, when training IMDN with the smaller DIV2K, there is a slight trend of overfitting. By contrast, training IMDN with DF2K does not show any sign of overfitting.

Based on the above analysis, we increase the patch size to train different networks with the proposed LSDIR dataset. As shown in Tab. 4, the performance of the network improves consistently with increasing patch size. Meanwhile, compared with small networks, larger networks benefit more from enlarged patch sizes. The validation accuracy during training with different patch sizes is shown in Fig. 3c. Larger patch sizes lead to consistently better performance.

Influence of training iterations. We also tried to increase the number of iterations to 5 million. As shown in Fig. 3b, prolonging the training on DIV2K and DF2K shows severe and slight overfitting effects, resp. By contrast, the validation accuracy increases steadily with LSDIR as training set.

In conclusion, *the benefit of a large scale dataset is that it enables performance boosts with increasing patch sizes and iteration numbers for training.*

Influence of training dataset size. We conduct an additional ablation study on the training dataset size. The percentage of LSDIR_X4 images used for training gradually increases from 1% to 100%. As shown in Tab. 5, the more the number of images used for training, the higher the PSNR values. Yet, the increase of PSNR tends to saturate when there are already plenty of images ($> 20\%$).

4.2. Performance on Different IR Tasks

The different training datasets are directly compared in this subsection and the result for image SR, gray image denoising, color image denoising, image demosaicking, image deblurring, color image JPEG compression artifacts removal are shown in Tab. 6 and Tab. 7. By comparing the experimental results between different datasets, we can conclude that the proposed dataset LSDIR could generally lead to improved performances for image restoration tasks. Secondly, to encourage research on lightweight and efficient

Table 7. Comparison between DIV2K, DF2K, and LSDIR for image deblurring and JPEG compression artifacts removal.

(a) Image deblurring results. Note that noise is also added.

Method	Kernel type Noise level	Dataset	McMaster PSNR/SSIM/LPIPS	Kodak PSNR/SSIM/LPIPS	Urban100 PSNR/SSIM/LPIPS
RDN [100]	Gaussian 2	DIV2K	35.20 / 0.9377 / 0.1822	32.25 / 0.8896 / 0.2490	30.59 / 0.9020 / 0.1896
		DF2K	35.16 / 0.9378 / 0.1830	32.23 / 0.8895 / 0.2503	30.58 / 0.9021 / 0.1893
		LSDIR_X4	35.33 / 0.9382 / 0.1832	32.35 / 0.8904 / 0.2487	30.82 / 0.9042 / 0.1880
	Kernel 4 [38] 2.55	DIV2K	35.07 / 0.9359 / 0.1930	33.99 / 0.9145 / 0.2261	32.81 / 0.9355 / 0.1590
		DF2K	35.19 / 0.9367 / 0.1936	34.12 / 0.9154 / 0.2268	32.96 / 0.9364 / 0.1594
		LSDIR_X4	35.39 / 0.9386 / 0.1906	34.34 / 0.9179 / 0.2239	33.35 / 0.9401 / 0.1565

(b) Color image JPEG compression artifacts removal. The quality factor is 40.

Method	Dataset	LIVE1 PSNR/SSIM/LPIPS	Classic5 PSNR/SSIM/LPIPS	Urban100 PSNR/SSIM/LPIPS
SwinIR [48]	DIV2K	35.04 / 0.9386 / 0.1897	35.63 / 0.9211 / 0.2386	35.61 / 0.9536 / 0.1425
	DF2K	35.07 / 0.9387 / 0.1886	35.68 / 0.9215 / 0.2378	35.67 / 0.9538 / 0.1409
	LSDIR_X4	35.11 / 0.9392 / 0.1871	35.70 / 0.9218 / 0.2376	35.83 / 0.9546 / 0.1397

Figure 4. Real image SR results with BSRGAN [92]. The upscaling factor is 4. First row: low-quality images; second row: training with DF2K; third row: training with LSDIR_X4.

models, the runtime and number of parameters are also reported for SR and denoising networks in Tab. 6a, Tab. 6b, and Tab. 6c. Although recent methods achieves higher evaluation values, the model size and runtime are also increased. Thirdly, to compare the perceptual quality of the restored images fairly and objectively, LPIPS score is also reported. The change of LPIPS score is generally consistent with PSNR and SSIM despite a few mismatches. Finally, we also conduct experiments on real-world image SR [75, 92]. Since there is no ground-truth images for images in the test set, the visual results are shown in Fig. 4. As shown in this figure, increasing the size of the training dataset helps to improve the visual quality of the SR images.

From Tab. 6 and Tab. 7, we have a few other important conclusions. First, compared with the other tasks, image SR and image deblurring benefit more from the enlarged dataset. Second, larger networks gain more from the large scale dataset. Third, by comparing the EDSR entries in

Tab. 4 and Tab. 6a, the diverse content of the large-scale dataset contributes more to PSNR gain than the enlarged patch size. Fourthly, with the large scale dataset and increased patch size, EDSR can lead to comparable or even better performance than SwinIR values reported in [48]. Fifth, in Fig. 3d, with larger patch size, training with LSDIR leads to better performance than training with ImageNet.

5. Conclusion

In this paper, we proposed a large scale dataset for image restoration (LSDIR). The proposed dataset contains 84,991 training images, 1,000 validation images, and 1,000 test images. Human annotators visually inspected the quality of the images. Additionally, we investigated the characteristics of the proposed dataset. Most importantly, experiments show that the large scale dataset enables training with much larger patch sizes and more training iterations, leading to superior performance. Compared with previous datasets for image restoration, the new dataset leads to better performance on the representative test set.

References

[1] Abdelrahman Abdelhamed, Stephen Lin, and Michael S Brown. A high-quality denoising dataset for smartphone cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1692–1700, 2018. 1

[2] Forest Agostinelli, Michael R Anderson, and Honglak Lee. Adaptive multi-column deep neural networks with application to robust image denoising. *Advances in Neural Information Processing Systems*, 26, 2013. 2

[3] Eirikur Agustsson and Radu Timofte. NTIRE 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 126–135, 2017. 1, 2, 3, 5, 6

[4] Dana Berman, Tali Treibitz, and Shai Avidan. Non-local image dehazing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1674–1682, 2016. 1

[5] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie Line Alberi-Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *Proceedings of the British Machine Vision Conference*, 2012. 3

[6] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 60–65. IEEE, 2005. 1, 2

[7] Jianrui Cai, Hui Zeng, Hongwei Yong, Zisheng Cao, and Lei Zhang. Toward real-world single image super-resolution: A new benchmark and a new model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3086–3095, 2019. 1

[8] David Capel and Andrew Zisserman. Computer vision applied to super resolution. *IEEE Signal Processing Magazine*, 20(3):75–86, 2003. 2

[9] Hong Chang, Dit-Yan Yeung, and Yimin Xiong. Super-resolution through neighbor embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages I–I, 2004. 1, 2

[10] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3291–3300, 2018. 1

[11] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12299–12310, 2021. 1, 2

[12] Xiangyu Chen, Xintao Wang, Jiantao Zhou, and Chao Dong. Activating more pixels in image super-resolution transformer. *arXiv preprint arXiv:2205.04437*, 2022. 1, 2

[13] Yunjin Chen and Thomas Pock. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1256–1272, 2016. 1

[14] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007. 1, 2

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE, 2009. 2, 3, 6

[16] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. Compression artifacts reduction by a deep convolutional network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 576–584, 2015. 1

[17] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *Proceeding of the European Conference on Computer Vision*, pages 184–199. Springer, 2014. 1, 2

[18] 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. 1

[19] Michael Elad. On the origin of the bilateral filter and ways to improve it. *IEEE Transactions on Image Processing*, 11(10):1141–1151, 2002. 1

[20] Sina Farsiu, M Dirk Robinson, Michael Elad, and Peyman Milanfar. Fast and robust multiframe super resolution. *IEEE transactions on image processing*, 13(10):1327–1344, 2004. 2

[21] Rich Franzen. Kodak lossless true color image suite. *source: http://r0k. us/graphics/kodak*, 4(2), 1999. 3

[22] 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, 2

[23] Xueyang Fu, Jiabin Huang, Delu Zeng, Yue Huang, Xinghao Ding, and John Paisley. Removing rain from single images via a deep detail network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3855–3863, 2017. 1

[24] Lovedeep Gondara. Medical image denoising using convolutional denoising autoencoders. In *IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 241–246. IEEE, 2016. 2

[25] Shuhang Gu, Yawei Li, Luc Van Gool, and Radu Timofte. Self-guided network for fast image denoising. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2511–2520, 2019. 6

[26] Shuhang Gu, Andreas Lugmayr, Martin Danelljan, Manuel Fritsche, Julien Lamour, and Radu Timofte. DIV8K: Diverse 8K resolution image dataset. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3512–3516. IEEE, 2019. 1, 3

[27] Shuhang Gu, Qi Xie, Deyu Meng, Wangmeng Zuo, Xiangchu Feng, and Lei Zhang. Weighted nuclear norm minimization and its applications to low level vision. *IJCV*, 121(2):183–208, 2017. 2

[28] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2862–2869, 2014. 1, 2

[29] Yu Guo, Qiyu Jin, Gabriele Facciolo, Tieyong Zeng, and Jean-Michel Morel. Residual learning for effective joint demosaicing-denoising. *arXiv preprint arXiv:2009.06205*, 2020. 1

[30] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12):2341–2353, 2010. 1

[31] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2015. 3

[32] Zheng Hui, Xinbo Gao, Yunchu Yang, and Xiumei Wang. Lightweight image super-resolution with information multi-distillation network. In *Proceedings of the ACM International Conference on Multimedia*, pages 2024–2032, 2019. 5, 6

[33] Jiaxi Jiang, Kai Zhang, and Radu Timofte. Towards flexible blind jpeg artifacts removal. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4997–5006, 2021. 1

[34] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 3

[35] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1646–1654, 2016. 1, 2

[36] Thomas Köhler, Michel Bätz, Farzad Naderi, André Kaup, Andreas Maier, and Christian Riess. Toward bridging the simulated-to-real gap: Benchmarking super-resolution on real data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(11):2944–2959, 2019. 1

[37] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4681–4690, 2017. 1, 2

[38] Anat Levin, Yair Weiss, Fredo Durand, and William T. Freeman. Understanding and evaluating blind deconvolution algorithms. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1964–1971, 2009. 6, 8

[39] Ruoteng Li, Loong-Fah Cheong, and Robby T Tan. Heavy rain image restoration: Integrating physics model and conditional adversarial learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1633–1642, 2019. 1

[40] Runde Li, Jinshan Pan, Zechao Li, and Jinhui Tang. Single image dehazing via conditional generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8202–8211, 2018. 1

[41] Wenbo Li, Xin Lu, Jiangbo Lu, Xiangyu Zhang, and Jiaya Jia. On efficient transformer and image pre-training for low-level vision. *arXiv preprint arXiv:2112.10175*, 2021. 1, 2

[42] Xia Li, Jianlong Wu, Zhouchen Lin, Hong Liu, and Hongbin Zha. Recurrent squeeze-and-excitation context aggregation net for single image deraining. In *Proceedings of the European Conference on Computer Vision*, pages 254–269, 2018. 1

[43] Yawei Li, Eirikur Agustsson, Shuhang Gu, Radu Timofte, and Luc Van Gool. CARN: convolutional anchored regression network for fast and accurate single image super-resolution. In *Proceeding of the European Conference on Computer VisionW*, pages 166–181. Springer, 2018. 2

[44] Yawei Li, Yuchen Fan, Xiaoyu Xiang, Denis Demandolx, Rakesh Ranjan, Radu Timofte, and Luc Van Gool. Efficient and explicit modelling of image hierarchies for image restoration. *arXiv preprint arXiv:2303.00748*, 2023. 1, 2

[45] Yu Li, Robby T Tan, Xiaojie Guo, Jiangbo Lu, and Michael S Brown. Rain streak removal using layer priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2736–2744, 2016. 1

[46] Yawei Li, Kai Zhang, Radu Timofte, Luc Van Gool, 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 (CVPR) Workshops*, 2022. 2

[47] Jingyun Liang, Jiezhang Cao, Yuchen Fan, Kai Zhang, Rakesh Ranjan, Yawei Li, Radu Timofte, and Luc

Van Gool. VRT: A video restoration transformer. *arXiv preprint arXiv:2201.12288*, 2022. 1

[48] 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 Workshops*, pages 1833–1844, 2021. 1, 2, 5, 6, 7, 8, 9

[49] Zhetong Liang, Shi Guo, Hong Gu, Huaqi Zhang, and Lei Zhang. A decoupled learning scheme for real-world burst denoising from raw images. In *European Conference on Computer Vision*, pages 150–166. Springer, 2020. 1

[50] Wei Liao, Xiang Zhang, Lei Yu, Shijie Lin, Wen Yang, and Ning Qiao. Synthetic aperture imaging with events and frames. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17735–17744, 2022. 1

[51] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1132–1140, 2017. 1, 2, 3, 5, 6, 7

[52] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2, 3

[53] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 1

[54] Andreas Lugmayr, Martin Danelljan, and Radu Timofte. Unsupervised learning for real-world super-resolution. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3408–3416. IEEE, 2019. 1

[55] Kede Ma, Zhengfang Duanmu, Qingbo Wu, Zhou Wang, Hongwei Yong, Hongliang Li, and Lei Zhang. Waterloo exploration database: New challenges for image quality assessment models. *IEEE Transactions on Image Processing*, 26(2):1004–1016, 2016. 3

[56] Henrique S Malvar, Li-wei He, and Ross Cutler. High-quality linear interpolation for demosaicing of bayer-patterned color images. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages iii–485. IEEE, 2004. 1

[57] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 2, pages 416–423. IEEE, 2001. 2, 3

[58] Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. *Multimedia Tools and Applications*, 76(20):21811–21838, 2017. 3

[59] Jinshan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang. Deblurring text images via l0-regularized intensity and gradient prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2908, 2014. 1

[60] José Luis Pech-Pacheco, Gabriel Cristóbal, Jesús Chamorro-Martinez, and Joaquín Fernández-Valdivia. Diatom autofocusing in brightfield microscopy: a comparative study. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, volume 3, pages 314–317. IEEE, 2000. 3

[61] Tobias Plotz and Stefan Roth. Benchmarking denoising algorithms with real photographs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1586–1595, 2017. 1

[62] 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. 2

[63] Chengchao Qu, Ding Luo, Eduardo Monari, Tobias Schuchert, and Jürgen Beyerer. Capturing ground truth super-resolution data. In *Proceedings of the IEEE International Conference on Image Processing*, pages 2812–2816. IEEE, 2016. 1

[64] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1, 2

[65] Wenqi Ren, Si Liu, Hua Zhang, Jinshan Pan, Xiaochun Cao, and Ming-Hsuan Yang. Single image dehazing via multi-scale convolutional neural networks. In *European Conference on Computer Vision*, pages 154–169. Springer, 2016. 1

[66] Wenqi Ren, Lin Ma, Jiawei Zhang, Jinshan Pan, Xiaochun Cao, Wei Liu, and Ming-Hsuan Yang. Gated fusion network for single image dehazing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3253–3261, 2018. 1

[67] Yaniv Romano, John Isidoro, and Peyman Milanfar. Raisr: rapid and accurate image super resolution. *IEEE Transactions on Computational Imaging*, 3(1):110–125, 2016. 2

[68] Qi Shan, Jiaya Jia, and Aseem Agarwala. High-quality motion deblurring from a single image. *ACM Transactions on Graphics (TOG)*, 27(3):1–10, 2008. 1

[69] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016. 2

[70] Radu Timofte, Vincent De Smet, and Luc Van Gool. Anchored neighborhood regression for fast example-based super-resolution. In *Proceedings of the IEEE international conference on computer vision*, pages 1920–1927, 2013. 1, 2

[71] Radu Timofte, Vincent De Smet, and Luc Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *Asian conference on computer vision*, pages 111–126. Springer, 2014. 1, 2

[72] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. *arXiv preprint arXiv:2012.12877*, 2020. 1

[73] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9446–9454, 2018. 1

[74] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, and Léon Bottou. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11(12), 2010. 2

[75] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 1905–1914, 2021. 1, 2, 8

[76] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 606–615, 2018. 1, 3

[77] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018. 5, 6

[78] Yingqian Wang, Longguang Wang, Jungang Yang, Wei An, and Yulan Guo. Flickr1024: A large-scale dataset for stereo image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 1

[79] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17683–17693, 2022. 1, 6, 7

[80] Jiqing Wu, Radu Timofte, and Luc Van Gool. Demosaicing based on directional difference regression and efficient regression priors. *IEEE Transactions on Image Processing*, 25(8):3862–3874, 2016. 1

[81] Junyuan Xie, Linli Xu, and Enhong Chen. Image denoising and inpainting with deep neural networks. *Advances in Neural Information Processing Systems*, 25, 2012. 2

[82] Jianchao Yang, John Wright, Thomas Huang, and Yi Ma. Image super-resolution as sparse representation of raw image patches. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 1, 2

[83] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*, 19(11):2861–2873, 2010. 1

[84] Wei Ye and Kai-Kuang Ma. Color image demosaicing using iterative residual interpolation. *IEEE Transactions on Image Processing*, 24(12):5879–5891, 2015. 1

[85] Liu Yuliang, Jin Lianwen, Zhang Shuaitao, and Zhang Sheng. Detecting curve text in the wild: New dataset and new solution. *arXiv preprint arXiv:1712.02170*, 2017. 3

[86] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5728–5739, 2022. 1, 6, 7

[87] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *Proceedings of International Conference on Curves and Surfaces*, pages 711–730. Springer, 2010. 3

[88] 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 *Proceedings of the European Conference on Computer Vision Workshops*, pages 5–40. Springer, 2020. 2

[89] Kai Zhang, Shuhang Gu, Radu Timofte, et al. Aim 2019 challenge on constrained super-resolution: Methods and results. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019. 2

[90] Kai Zhang, Yawei Li, Jingyun Liang, Jiezhang Cao, Yulun Zhang, Hao Tang, Radu Timofte, and Luc Van Gool. Practical blind denoising via swin-conv-unet and data synthesis. *arXiv preprint arXiv:2203.13278*, 2022. 1, 2

[91] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte. Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1, 2, 6, 7

[92] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4791–4800, 2021. 1, 2, 6, 8

[93] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 1, 2, 6, 7

[94] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Deep plug-and-play super-resolution for arbitrary blur kernels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1671–1681, 2019. 2

[95] Lei Zhang and Xiaolin Wu. Color demosaicking via directional linear minimum mean square-error estimation. *IEEE Transactions on Image Processing*, 14(12):2167–2178, 2005. 1, 2

[96] Lei Zhang, Xiaolin Wu, Antoni Buades, and Xin Li. Color demosaicking by local directional interpolation and nonlocal adaptive thresholding. *Journal of Electronic imaging*, 20(2):023016, 2011. 1, 3

[97] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 6

[98] 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. 6, 7

[99] 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*, 2018. 1

[100] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(7):2480–2495, 2020. 1, 2, 6, 8