Learning to Zoom Inside Camera Imaging Pipeline

Chengzhou Tang[1*†] Yuqiang Yang[2*] Bing Zeng[2] Ping Tan[1] Shuaicheng Liu[2†]
[1]Simon Fraser University
[2]University of Electronic Science and Technology of China

Abstract

Existing single image super-resolution methods are either designed for synthetic data, or for real data but in the RGB-to-RGB or the RAW-to-RGB domain. This paper proposes to zoom an image from RAW to RAW inside the camera imaging pipeline. The RAW-to-RAW domain closes the gap between the ideal and the real degradation models. It also excludes the image signal processing pipeline, which refocuses the model learning onto the super-resolution. To these ends, we design a method that receives a low-resolution RAW as the input and estimates the desired higher-resolution RAW jointly with the degradation model. In our method, two convolutional neural networks are learned to constrain the high-resolution image and the degradation model in lower-dimensional subspaces. This subspace constraint converts the ill-posed SISR problem to a well-posed one. To demonstrate the superiority of the proposed method and the RAW-to-RAW domain, we conduct evaluations on the RealSR and the SR-RAW datasets. The results show that our method performs superiorly over the state-of-the-arts both qualitatively and quantitatively, and it also generalizes well and enables zero-shot transfer across different sensors.

1. Introduction

Digital zoom is necessary for devices with limited optical zoom capability such as smart phones or consumer level cameras. The digital zoom can be implemented either as built-in functionality in the camera preview, where selected distant image areas are enlarged in real-time, or as post-processing in the editing software, where a single input image is up-sampled into higher-resolution for more details.

In the past few decades, numerous works [3, 8, 12, 18, 24, 33, 37, 38, 43, 52] have been done on single image super-resolution (SISR) because it is less demanding on the input. Some of these works, especially the learning based ones [39], assume the degradation kernel follows a manually defined formulation such as a Gaussian distribution or a Bicubic interpolation, and synthesize paired low-resolution (LR) and high-resolution (HR) image pairs to design and evaluate the methods. However, the degradation model on real data is more complicated than the made assumptions. This domain gap limits the performances of these methods on real data.

Recently, some works start to explore the SISR problem on real data and have pushed this field forward by several progresses. Real datasets such as RealSR [5], SR-RAW [51] and ImagePairs [19], have been proposed, where a LR-HR pair is captured towards the same scene either with varying camera focal lengths or with a beam splitter. These datasets bridge the domain gap and enable to train a model on real RGB pairs [5, 40, 46]. X. Zhang *et al.* [51] and Z. Zhang *et al.* [53] even take a step further by designing models that output a higher-resolution RGB image but receive a RAW image as the input. These models not only zoom an input image into higher resolution but also serve as implicit image signal processing (ISP) pipelines to convert a RAW image to RGB, which could complicate the learning by involving ISP units and bias the learned model to specific cameras.

In this paper, we also work on SISR for real data but in the *RAW-to-RAW* domain. We design a digital **Z**oom component that can be built **I**nto a **C**amera imaging pipeline, namely ZIC. ZIC excludes the ISP and only focuses on zooming an image with increased details from RAW to RAW. The superiority of working in RAW-to-RAW domain is threefold. First, the degradation kernels from RAW-HR to RAW-LR images are closer to ideal distributions, which potentially contributes to better results. Second, RAW images are independent of the ISP, so the model trained focuses on the resolution enhancement and is less biased to specific cameras. Third, RAW images are more flexible during post-processing, because they contain the original information from sensors. To the best of our knowledge, ZIC is the first SISR method in the RAW-to-RAW domain for real data. To that end, ZIC solves the following minimization problem:

$$\min_{\boldsymbol{x},\boldsymbol{k}} \|\boldsymbol{y} - (\boldsymbol{k} \otimes \boldsymbol{x}) \downarrow \|, \text{ s.t. } \boldsymbol{x} \in \mathcal{X} \text{ and } \boldsymbol{k} \in \mathcal{K}, \quad (1)$$

where $\boldsymbol{y}$ is the input RAW image, $\boldsymbol{x}$ is the desired higher-resolution output along with a pixel-adaptive degradation kernel map $\boldsymbol{k}$, $\otimes$ denotes the convolution operator, and $\downarrow$

[*] Equal contribution
[†] Corresponding author

denotes the down-sampling operator which is omitted for simplicity in the following content of the paper.

Minimizing the objective function only in Eq. (1) is an ill-posed inverse problem where multiple $\boldsymbol{x}$ correspond to the same observation $\boldsymbol{y}$, and it becomes even more challenging when jointly estimating $\boldsymbol{k}$. Inspired by the recent progress made in other low-level vision tasks [35], we assume the output $\boldsymbol{x}$ and the kernel map $\boldsymbol{k}$ belong to two latent subspaces $\mathcal{X}$ and $\mathcal{K}$ respectively. This subspace constraint regularizes the solution space and reformulates an ill-posed SISR problem into a better posed one. In order to generate the two subspace $\mathcal{X}$ and $\mathcal{K}$, we feed $\boldsymbol{y}$ to two standalone convolutional networks, and they are trained in a supervised manner with RAW LR-HR pairs from existing datasets [5, 51].

To evaluate the proposed method ZIC, we made comparisons against existing SISR methods on RealSR [5] and SR-RAW [51]. Since our output is in RAW format, we converted it into RGB by the official software of each camera. Therefore, fair comparisons were made quantitatively with SSIM, PSNR, and the learned perceptual metric LPIPS [50]. We also made qualitative comparisons and demonstrated the superiority of our method in visual quality. Both the quantitative and the qualitative results indicate the proposed method is superior than previous ones by a large margin, and the contribution of each component was also verified in ablation studies. Furthermore, ZIC generalizes well and enables zero-shot transfer across different sensors.

2. Related Works

Single image super-resolution(SISR) is an ill-posed inverse problem, where the desired higher-resolution image has infinite solutions when the degradation model is unknown, or can be recovered only up to a limit [21, 27] with a given degradation model. To deal with this problem, numerous earlier works have been proposed such as the example-based methods [2, 10, 12, 16, 34, 36] and the sparse-coding-based methods [20, 43, 47].

SISR on Synthetic Data: More recent researches have been done on learning deep neural networks for SISR. Beginning from SRCNN [8], where a convolution neural network(CNN) is trained on Bicubic downsampled LR-HR pairs, various following up works have been done in this direction. First, improved network architectures have been explored for either RGB images [24, 52] or multi-spectral images [29, 30]. Second, advanced learning strategies such as the perceptual loss [18] and the generative-adversarial-network [23, 38, 44] are applied during training. Third, some works decouple the degradation model from the neural network, and require an external kernel as the input. This kernel can be either manually given [48, 49] or estimated from the internal statics within a single image [4, 28]. Most of these methods are trained and evaluated on synthetic data [1, 42], where a LR image is synthesized from the HR image with manually defined degradation.

SISR on Real Data: However, the degradation models on real data are more complicated. This domain gap motivates the recent works to explore the SISR problem on real LR-HR pairs. Cai *et al.* [5] capture a benchmark data that contains aligned LR-HR image pairs in RGB, and design a network that estimates the per-pixel kernels at multiple-scales. Zamir *et al.* [46] also use this dataset and learn enriched multi-scale features for SISR on real RGB images. Inspired by the recent learning based RAW image processing [7, 11, 17, 41, 45], X.Zhang *et al.* [51] propose a SISR method that zooms a RAW image into a higher-resolution RGB image, where a contextual bilateral loss is designed to deal with misaligned training pairs. Instead, Z.Zhang *et al.* [53] directly reduce the misalignment during training by guiding the joint learning of the ISP network and the global color mapping with a pre-trained optical flow model [32].

We recommend [39] for a more comprehensive survey about the aforementioned methods, especially the deep learning-based ones. Different from the previous learning-based methods, our method minimizes the objective function in Eq. (1) with learned subspace constraints. We do not engineer network architectures or training strategies to learn the subspaces, but adopt standard networks and loss functions. Another equally important difference is that we work in the RAW-to-RAW data domain where the ISP is excluded. Therefore, our method is relieved from learning the ISP and refocused onto the SISR problem.

3. The Method

3.1. Overview

Fig. 1 gives an overview of the proposed ZIC framework, where we receive a RAW image $\boldsymbol{y}$ as the input and send it to the image basis generator as well as the kernel basis generator in parallel. The image basis $\mathcal{X}$ contains $d_{\boldsymbol{x}}$ basis images for each Bayer pattern channel but with higher-resolution, while the kernel basis $\mathcal{K}$ contains $d_{\boldsymbol{k}}$ degradation basis kernels at each input pixel coordinate. Then the higher-resolution RAW $\boldsymbol{x}$ and the corresponding degradation kernel map $\boldsymbol{k}$ are jointly estimated by Eq. (1). Finally, $\boldsymbol{x}$ is sent to an existing ISP for further processing into RGB format.

3.2. Why RAW?

As shown in Fig. 2a, a typical ISP receives the original RAW image from an image sensor, and then post-processes the input through several ISP stages, including black level correction, white balancing, demosaicing, noise reduction, etc. Before introducing ZIC in detail, we explain why to exclude the ISP and work in the RAW-to-RAW domain.

We investigate the degradation kernel between a LR image $\boldsymbol{y}$ and its ground-truth HR counterpart $\boldsymbol{x}^*$ because the kernel estimation is critical for the quality of the output.

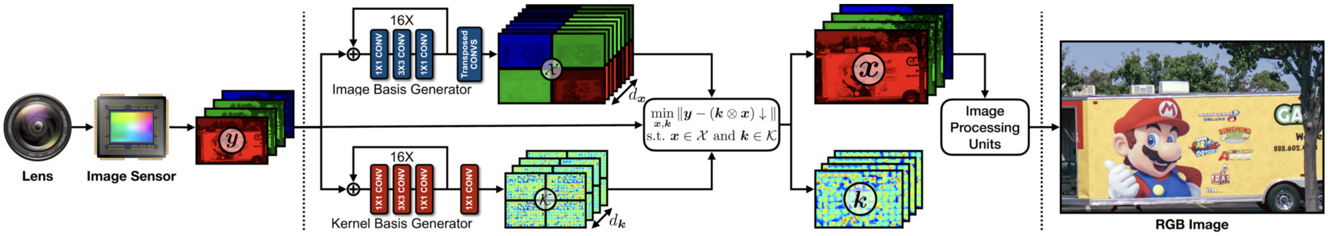
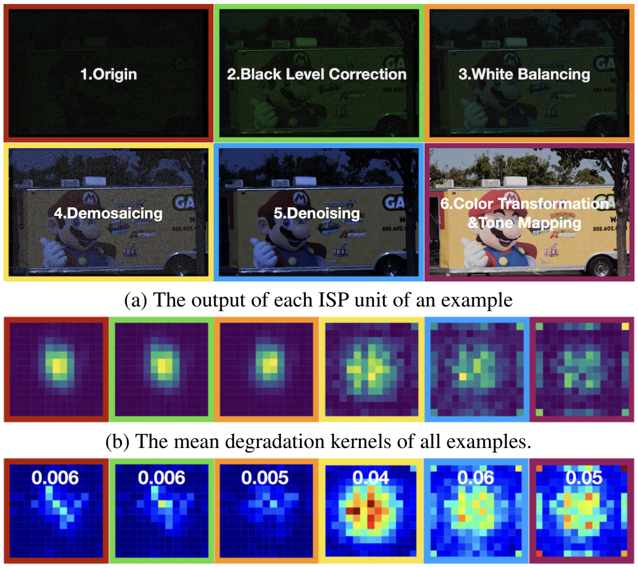


Figure 1. The framework of the proposed learning to **Z**oom **I**nside **C**amera pipeline.

Once the kernel has been decided, we can directly recover the high-resolution image by Fast Fourier Transform (FFT) or variational methods [37]. Ideally, the degradation kernel follows a Gaussian distribution depends on the depth at each pixel and the focal length of the camera [6]:

$$g(\boldsymbol{p}, \boldsymbol{q}) = \frac{1}{2\pi\sigma^2(d_{\boldsymbol{p}}, f)} \exp(-\frac{\|\boldsymbol{p} - \boldsymbol{q}\|_2^2}{2\sigma^2(d_{\boldsymbol{p}}, f)}), \qquad (2)$$

where $\boldsymbol{p}$ is a pixel, $\boldsymbol{q}$ is one of its neighbors, $\sigma^2(d_{\boldsymbol{p}}, f)$ is a function about the depth $d_{\boldsymbol{p}}$ of $\boldsymbol{p}$ and the focal length f.



(a) The output of each ISP unit of an example

(b) The mean degradation kernels of all examples.

(c) The mean Gaussian fitting error of all examples.

Figure 2. Kernel statistics on the output of each ISP stage.

However, the above assumption is violated on real RGB images because of the ISP. For ease of explanation, we only consider planar scenes parallel to the camera plane, where the scene captured has a uniform depth. Therefore, a uniform kernel can be estimated from the $\boldsymbol{x}^*, \boldsymbol{y}$ pair by the following constrained quadratic minimization:

$$\min_{\boldsymbol{k}} \|\boldsymbol{y} - \boldsymbol{k} \otimes \boldsymbol{x}^*\|, \text{ s.t. } \boldsymbol{k}_{ij} > 0, \text{ and } \sum \boldsymbol{k}_{ij} = 1, \qquad (3)$$

where the estimated kernel $\boldsymbol{k}$ is expected to be close to the distribution in Eq. (2). As shown in Fig. 2b, the estimated kernel starts to diverge and deviate after demosaicing, while it is more compact and centralized before that. We also fit each kernel to a Gaussian distribution and average the fitting error at each kernel element in Fig. 2c. The mean fitting error has also increased by $8\times$ after demosaicing. This observation indicates the kernel assumption Eq. (2) is well satisfied until demosaicing, which motivates our method to not only receive RAW images as inputs but also produce RAW images as outputs.

The insight behind this observation is that there are two groups of processing units inside an ISP: first are the ones that operate pixel-wisely such as the black level correction and the white balancing, while second are the ones that operate in a local neighborhood such as the demosaicing and the noise reduction. The neighborhood information involved in the second group of units complicates the kernel estimation, which violates the assumption in Eq. (2). Please refer to the *supplementary* for detailed explanation. In practice, we preserve the black level correction and the white balancing as our data normalization, because they do not complicate the kernel assumption while remap pixel values to $[0, 1]$ with balanced color.

3.3. How to Zoom?

Now we have explained the reason for working on the SISR problem in the RAW-to-RAW domain. Next, we will introduce the technical details of the proposed ZIC method.

Kernel Estimation: Although the underlying degradation kernels become closer to Gaussian distributions on RAW images, they are still spatially-variant because each pixel has a different distance to the camera in a 3D scene. Given an existing input-output pair $\{\boldsymbol{x}, \boldsymbol{y}\}$, the kernel $\boldsymbol{k_p}$ at a pixel coordinate $\boldsymbol{p}$ satisfies the following equation:

$$\boldsymbol{y_p} = \boldsymbol{k_p} \cdot \boldsymbol{x}_{\mathcal{N}_{\boldsymbol{p}}}, \qquad (4)$$

where $\mathcal{N}_{\boldsymbol{p}}$ represents the neighborhood around $\boldsymbol{p}$ and has the same spatial size with $\boldsymbol{k_p}$, so the convolution operator can be implemented as dot product between flattened vectors. However, Eq. (4) is under-determined to decide the kernel $\boldsymbol{k_p}$. Inspired by recent works on other low-level vision tasks [35], we parameterize the kernel map $\boldsymbol{k}$ as a linear combination of $d_{\boldsymbol{k}}$ basis kernel maps $\mathcal{K} = \{\boldsymbol{k}^1, \boldsymbol{k}^2, \cdots \boldsymbol{k}^{d_{\boldsymbol{k}}}\}$ and estimate the combination coefficients $\boldsymbol{c_k} = [c_{\boldsymbol{k}}^1, c_{\boldsymbol{k}}^2, \cdots c_{\boldsymbol{k}}^{d_{\boldsymbol{k}}}]$ via:

$$\min_{\boldsymbol{c_k}} \sum_{\boldsymbol{p}} \|\boldsymbol{y_p} - \sum_{i=1}^{d_{\boldsymbol{k}}} c_{\boldsymbol{k}}^i (\boldsymbol{k}_{\boldsymbol{p}}^i \cdot \boldsymbol{x}_{\mathcal{N}_{\boldsymbol{p}}})\|. \tag{5}$$

These kernel basis maps are generated from a ResNet [15] with 16 ResBlocks, where no stride is applied to generate basis kernel maps in the same resolution as the input.

Image Estimation: Meanwhile, once $\boldsymbol{k}$ is given, we can recover the desired high-resolution image $\boldsymbol{x}$ by:

$$\min_{\boldsymbol{x}} \sum_{\boldsymbol{p}} \|\boldsymbol{y_p} - \boldsymbol{k_p} \cdot \boldsymbol{x}_{\mathcal{N}_{\boldsymbol{p}}}\|. \tag{6}$$

Although Eq. (6) is over-determined, estimating $\boldsymbol{x}$ pixel-wisely usually introduce artifacts [13]. To achieve better regularization, we also parameterize $\boldsymbol{x}$ as a linear combination of $d_{\boldsymbol{x}}$ basis images $\mathcal{X} = \{\boldsymbol{x}^1, \boldsymbol{x}^2, \cdots \boldsymbol{x}^{d_{\boldsymbol{x}}}\}$ and estimate the combination coefficients $\boldsymbol{c_x} = [c_{\boldsymbol{x}}^1, c_{\boldsymbol{x}}^2, \cdots c_{\boldsymbol{x}}^{d_{\boldsymbol{x}}}]$ via:

$$\min_{\boldsymbol{c_x}} \sum_{\boldsymbol{p}} \|\boldsymbol{y_p} - \sum_{j=1}^{d_{\boldsymbol{x}}} c_{\boldsymbol{x}}^j (\boldsymbol{k_p} \cdot \boldsymbol{x}_{\mathcal{N}_{\boldsymbol{p}}}^j)\|. \tag{7}$$

These basis images are generated from a similar network as the basis kernel maps. The main difference is that we further up-sample the output from the final convolutional layer by transposed convolutions, which converts the basis images to the desired higher-resolution. Besides regularization, these basis images also reduce the number of variables to $d_{\boldsymbol{x}}$ for an image, which achieves better efficiency and makes the following joint minimization possible.

Joint Minimization: Now, we have the basis kernel maps $\{\boldsymbol{k}^1, \boldsymbol{k}^2, \cdots \boldsymbol{k}^{d_{\boldsymbol{k}}}\}$ and the basis images $\{\boldsymbol{x}^1, \boldsymbol{x}^2, \cdots \boldsymbol{x}^{d_{\boldsymbol{x}}}\}$. The combination coefficients $\boldsymbol{c_k}$ and $\boldsymbol{c_x}$ are estimated via:

$$\min_{\boldsymbol{c_k}, \boldsymbol{c_x}} \sum_{\boldsymbol{p}} \|\boldsymbol{y_p} - \sum_{i=1}^{d_{\boldsymbol{k}}} \sum_{j=1}^{d_{\boldsymbol{x}}} c_{\boldsymbol{k}}^i c_{\boldsymbol{x}}^j z_{\boldsymbol{p}}^{ij}\|, \tag{8}$$

where $z_{\boldsymbol{p}}^{ij} = \boldsymbol{k}_{\boldsymbol{p}}^i \cdot \boldsymbol{x}_{\mathcal{N}_{\boldsymbol{p}}}^j$. Instead of alternating between Eq. (5) and Eq. (7), we solve Eq. (8) jointly as follows:

- Let $\boldsymbol{e} = \boldsymbol{c}_{\boldsymbol{k}}^\top \boldsymbol{c_x}$, *i.e.*, $e_{ij} = c_{\boldsymbol{k}}^i c_{\boldsymbol{x}}^j$, we first minimize Eq. (8) as a linear problem respect to $\boldsymbol{e}$. To get the initial $\boldsymbol{c_k}$ and $\boldsymbol{c_x}$, we decompose the rank-1 matrix $\boldsymbol{e}$ by a single iteration of the power method [25].
- Second, we iteratively update $\boldsymbol{c_k}$ and $\boldsymbol{c_x}$ as $\boldsymbol{c_k} \leftarrow \boldsymbol{c_k} + \Delta\boldsymbol{c_k}$ and $\boldsymbol{c_x} \leftarrow \boldsymbol{c_x} + \Delta\boldsymbol{c_x}$. The incremental updates $\Delta\boldsymbol{c_k}$ and $\Delta\boldsymbol{c_x}$ are solved as follows at each iteration:

  $$\min_{\Delta\boldsymbol{c_k}, \Delta\boldsymbol{c_x}} \sum_{\boldsymbol{p}} \|\boldsymbol{y_p} - \sum_{i=1}^{d_{\boldsymbol{k}}} \sum_{j=1}^{d_{\boldsymbol{x}}} (c_{\boldsymbol{k}}^i + \Delta c_{\boldsymbol{k}}^i)(c_{\boldsymbol{x}}^j + \Delta c_{\boldsymbol{x}}^j) z_{\boldsymbol{p}}^{ij}\|, \tag{9}$$

  where $(c_{\boldsymbol{k}}^i + \Delta c_{\boldsymbol{k}}^i)(c_{\boldsymbol{x}}^j + \Delta c_{\boldsymbol{x}}^j) \approx c_{\boldsymbol{k}}^i c_{\boldsymbol{x}}^j + c_{\boldsymbol{k}}^i \Delta c_{\boldsymbol{x}}^j + \Delta c_{\boldsymbol{k}}^i c_{\boldsymbol{x}}^j$ by omitting the second-order term $\Delta c_{\boldsymbol{k}}^i \Delta c_{\boldsymbol{x}}^j$, which makes Eq. (9) linear respect to $[\Delta\boldsymbol{c_k}, \Delta\boldsymbol{c_x}]$.
- Finally, we compose the desired high-resolution image as $\boldsymbol{x} = \sum_{j=1}^{d_{\boldsymbol{x}}} \boldsymbol{c}_{\boldsymbol{x}}^j \boldsymbol{x}^j$ and align the color space of the output $\boldsymbol{x}$ with the input $\boldsymbol{y}$.

3.4. Training

We use Eq. (8) for inference but utilize the underly properties of the basis kernel maps and the basis images to achieve direct supervision during training.

Kernel Basis: Given the ground-truth high-resolution image $\boldsymbol{x}^*$, we estimate the combination coefficients of basis kernel maps by Eq. (5). Then we compose the kernel map $\boldsymbol{k} = \sum_{i=1}^{d_{\boldsymbol{k}}} \boldsymbol{c}_{\boldsymbol{k}}^i \boldsymbol{k}^i$ and generate the degraded ground-truth $\boldsymbol{y}^*$ by Eq. (4) at each pixel. The training loss is applied between $\boldsymbol{y}^*$ and the input $\boldsymbol{y}$.

Image Basis: Since $\boldsymbol{x}^*$ is given during training, we estimate the combination coefficients $\boldsymbol{c_x}$ as:

$$\min_{\boldsymbol{c_x}} \sum_{\boldsymbol{p}} \|\boldsymbol{x}_{\boldsymbol{p}}^* - \sum_{j=1}^{d_{\boldsymbol{x}}} c_{\boldsymbol{x}}^j \boldsymbol{x}_{\boldsymbol{p}}^j\|, \tag{10}$$

and apply the loss between the ground-truth $\boldsymbol{x}^*$ and the composed output $\boldsymbol{x} = \sum_{j=1}^{d_{\boldsymbol{x}}} \boldsymbol{c}_{\boldsymbol{x}}^j \boldsymbol{x}^j$.

Training Loss: We apply the same loss functions for both the kernel basis generator and the image basis generator. We follow previous works [18, 51] and measure the L_1 distance between $\boldsymbol{x}^*$ and $\boldsymbol{x}$ as well as the corresponding VGG-19 features $\boldsymbol{f}(\boldsymbol{x}^*)$ and $\boldsymbol{f}(\boldsymbol{x})$. To send $\boldsymbol{x}^*$ and $\boldsymbol{x}$ to a pre-trained VGG-19 network [31], we convert the single-channel Bayer data into a 3-channel image by interpolating the missing values in a Bayer pattern image from its closest 2 or 4 neighbors, and then align the color space of $\boldsymbol{x}^*$ and $\boldsymbol{x}$ with the RGB version of $\boldsymbol{x}^*$. We also use a confidence map $\boldsymbol{w}$ to compensate misalignment and will describe it further in the next section. In summary, the loss to train the image basis generator is:

$$\mathcal{L} = \sum_{\boldsymbol{p}} \boldsymbol{w_p}(|\boldsymbol{x}_{\boldsymbol{p}}^* - \boldsymbol{x_p}| + |\boldsymbol{f_p}(\boldsymbol{x}^*) - \boldsymbol{f_p}(\boldsymbol{x})|), \tag{11}$$

which is the same for $\boldsymbol{y}^*$ and $\boldsymbol{y}$ in kernel basis training.

4. Experiments

4.1. Datasets

In order to make fair comparisons with previous methods, we train our model and made evaluations on two existing datasets which are captured with RAW images.

RealSR: RealSR [5] captures real-world LR-HR image pairs by adjusting the focal-length of a camera towards the same scene. RealSR captures both the indoor and the outdoor images with one Nikon and one Canon camera. These images are carefully captured with small parallax and further aligned with photometric error in the RGB domain. The number of training pairs for scale factors 2×, 3× and 4× are 183, 234 and 178, respectively. While 30 testing pairs are provided for each scale.

SR-RAW: SR-RAW [51] captures a dataset of 500 scenes with a Sony camera, where the focal-lengths are also adjusted to generate image pairs with ×4 and ×8 scale factors. However, SR-RAW is more challenging than RealSR

Types	Method	RealSR						SR-RAW					
		x2			x4			x2			x4		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
	Bicubic	30.56	0.884	0.248	26.04	0.756	0.406	29.32	0.835	0.269	26.12	0.775	0.360
Class A	LapSRN [22]	30.79	0.888	0.246	26.31	0.768	0.400	29.49	0.852	0.259	26.20	0.781	0.357
	EDSR [24]	30.87	0.890	0.244	26.38	0.770	0.399	29.51	0.853	0.258	26.24	0.781	0.356
	RCAN [52]	30.88	0.889	0.245	26.42	0.771	0.399	29.52	0.853	0.257	26.25	0.782	0.354
	ESRGAN [38]	-	-	-	26.30	0.764	0.409	-	-	-	26.19	0.776	0.363
	IKC [14]	-	-	-	26.32	0.771	0.405	-	-	-	26.16	0.779	0.358
Class B	SRMD [49]	29.52	0.899	0.285	26.50	0.779	0.407	29.43	0.854	0.253	25.82	0.775	0.347
	ZSSR [28]	30.93	0.899	0.243	24.72	0.733	0.416	27.72	0.811	0.270	24.30	0.739	0.381
	ZIC(Ours)	35.64	0.956	0.189	32.58	0.926	0.212	36.66	0.976	0.141	32.89	0.927	0.190

Table 1. Quantitative comparisons with methods designed on synthetic data. The models of the other methods are officially released and can be categorized into Class A which only receive an LR image as the input, and Class B which require additional degradation kernels. While ours was trained on RealSR [5] and SR-RAW [51] simultaneously. The best and the second best are marked in red and blue at each column. '-' indicates there is no released model for such a test.

because of larger perspective change, illumination change, inconsistency between the input and the output caused by lens distortion, *etc*. Although alignment has been done by ECC [9], the produced RAW-to-RGB pairs still contain mild but non-negligible misalignment.

RAW-to-RAW Alignment: The above two datasets are designed for SISR in the RGB-to-RGB or the RAW-to-RGB domain, so aligned RAW-to-RAW data are unavailable. Therefore, we post-process RealSR and SR-RAW by aligning the LR-HR pairs from RAW to RAW, and the processed data are *only* used during training. In general, we first fit an initial global motion model from sparse feature matching [26] in RGB images, and then refine this model via the photometric alignment from RealSR [5], but on demosaiced RAW images. Finally, we produce a confidence map that measures the quality of the global alignment. The confidence map is calculated from the photometric difference and the residual optical flow fields between globally aligned images, and is used to reduce the weights of the misaligned pixels during training. Please refer to the *supplementary* for more details of our data alignment, and we will release the processed dataset to encourage further researches.

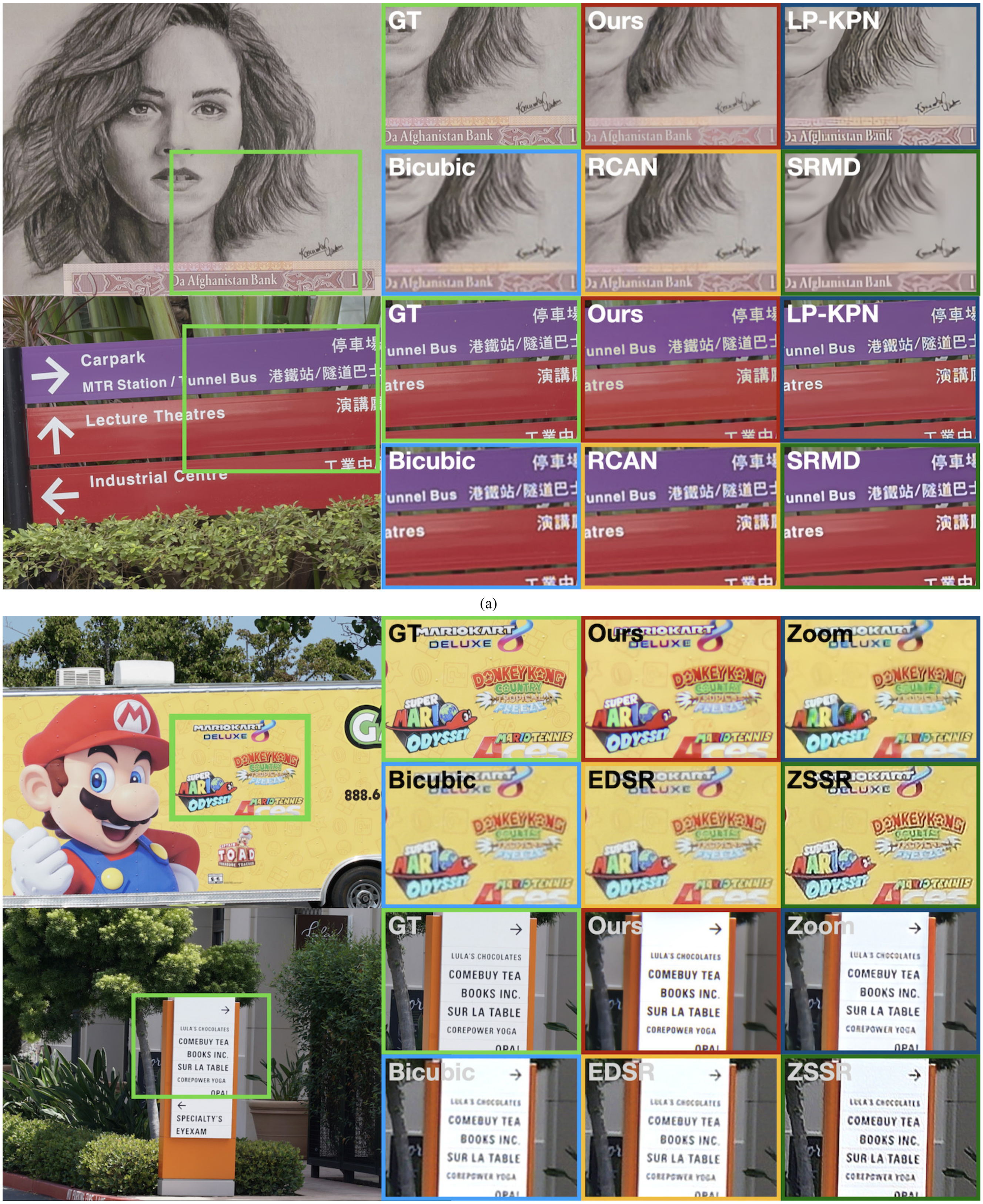
4.2. Comparisons with Other Methods

To demonstrate the superiority of our method on real data, we made comparisons with the *officially* released models from several representative deep learning based methods. These methods are categorized into the ones that are designed for real data including LP-KPN [5], MIRNet [46], RAW-to-sRGB [53], Zoom-learn-Zoom [51], as well as the ones that are trained on synthetic data with manually defined degradation including LapSRN [22], EDSR [24], RCAN [52], ESRGAN [38], and IKC [14]. In addition, we also compare with SRMD [49] and ZSSR [28] which requires external kernels as inputs for inference. **Quantitative Evaluation:** To quantitatively evaluate the proposed method, we use standard metrics including SSIM, PSNR, and the learned perceptual measurement LPIPS [50] to measure the difference between the prediction $\boldsymbol{x}$ and the ground-truth $\boldsymbol{x}^*$. To reduce the negative impact caused by the misalignment, we also estimate confidence maps for the *original* RealSR and the SR-RAW datasets, and exclude the pixels with low confidence from evaluation. Our method performs significantly better than previous models trained either in RGB-to-RGB or in RAW-to-RGB domain. In Tab. 1, the models trained on synthetic RGB data consistently perform worse than ours, and this disadvantage is also significant for the models that require external kernel inputs, both of which indicate that the degradation model is oversimplified in synthetic data. In Tab. 2, our method is still superior to the other models trained on real data but in the RGB-to-RGB or the RAW-to-RGB domain by a large margin.

Dataset	Method	PSNR↑	SSIM↑	LPIPS↓
RealSR	LP-KPN [5]	27.89	0.819	0.349
	MIRNet [46]	27.58	0.805	0.370
	Ours	**32.58**	**0.926**	**0.212**
SR-RAW	Zoom-learn-Zoom [51]	27.37	0.809	0.336
	RAW-to-sRGB [53]	27.56	0.813	0.326
	Ours	**32.89**	**0.927**	**0.190**

Table 2. Comparisons with methods trained on RealSR [5] and SR-RAW [51] datasets under 4× scale ratio.

Qualitative Evaluation: Since the quantitative metrics are not always consistent with human judgment [50], we also made qualitative comparisons against other methods under 4× scale ratio. Our method achieves better visual quality in Fig. 3a and Fig. 3b. The models trained on synthetic data, such as EDSR [24] and RCAN [52], give results close to Bicubic upsampling, which has also been observed by [5].



(a)

(b)

Figure 3. Qualitative comparisons on (a) RealSR [5] and (b) SR-RAW [51]

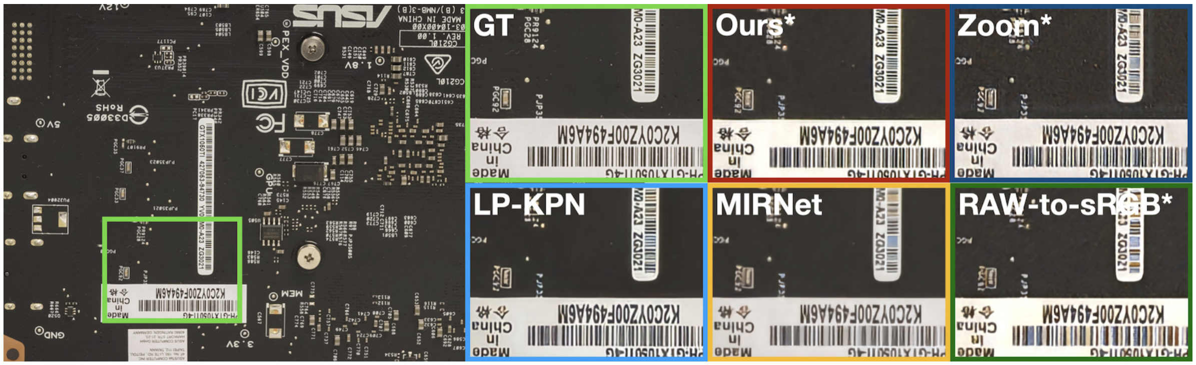


Figure 4. Qualitative zero-shot transfer comparison, where $*$ denote the model is trained on SR-RAW [51] and then zero-shot transferred to RealSR [5], otherwise the model is trained on RealSR [5].

While the models require external kernels either hallucinate textures in ZSSR [28] or give piece-wise smoothed results in SMRD [49]. Meanwhile, the models trained on real data such as LP-KPN [5] and Zoom-learn-zoom [51] achieve better results than the synthetic ones, but their results are visually less plausible than ours because they are designed either for the RGB-to-RGB domain or the RAW-to-RGB domain. Please refer to the *supplementary* for more results.

4.3. Zero-shot Transfer to Unseen Sensors

Method	PSNR↑	SSIM↑	LPIPS↓
Zoom-learn-Zoom [51]	26.57	0.832	0.393
RAW-to-sRGB [53]	26.11	0.826	0.410
Ours	**32.07**	**0.920**	**0.238**

Table 3. Quantitative comparison for models trained on SR-RAW [51] and zero-shot transferred to RealSR [5].

Different image sensors have different characteristics [51], the proposed ZIC method is robust to this difference because it explicitly minimizes the objective function defined in Eq. (8), so the output is estimated to fit the statistics of each input sample individually. To demonstrate the zero-shot transferability of ZIC, we train our model with the same settings as for Sec. 4.2, but only with the SR-RAW dataset. Then we made comparisons on the RealSR [5] with Zoom-learn-Zoom [51] and RAW-to-sRGB [53]. The additional quantitative results are listed in Tab. 3, and the qualitative results are shown in Fig. 4. The results indicate that the proposed ZIC model generalizes to unseen sensors in zero-shot while the methods designed for RAW-to-RGB domain [51, 53] are biased to the sensors of the training data. Our zero-shot transferred model even performs better than the methods [5, 46] both trained and evaluated on RealSR.

4.4. Ablation Studies

In this paper, we proposed ZIC——the first method that zooms an image into higher-resolution from RAW to RAW. We conduct ablation studies to further verify the effectiveness of each individual component and give some insights behind the designs.

Global v.s. Local Kernel: The first question is whether we can replace the basis kernel maps with a single uniform kernel? To answer this question, we estimate the kernel in the original form along with the image basis coefficients, which gives worse results quantitatively in Tab. 4. Fig. 5 shows that a global kernel can not handle the pixels around image boundaries well and produces color artifacts. It is because the degradation kernels are spatially-variant and depend on the depth at pixel as introduced in Sec. 3.2.

Joint v.s. Alternating Minimization: The second question is whether the joint minimization is necessary. The alternating minimization is a popular strategy applied by both the conventional [37] and the learning based methods [14,48,54]. To initialize the alternating minimization, we apply Bicubic up-sampling to get $\boldsymbol{x}$ from $\boldsymbol{y}$, and estimate the kernel map $\boldsymbol{k}$ with Eq. (5). Then we update $\boldsymbol{x}$ by Eq. (7) and repeat the alternating procedure between Eq. (5) and Eq. (7) until the solutions converge. As shown in Tab. 4, the alternating minimization performs worse quantitatively, and it also produces color artifacts on the letters and the numbers in Fig. 5.

RAW v.s. RGB: The third question is whether it is necessary to zoom an image in the RAW-to-RAW domain? As observed in Sec. 3.2, the Gaussian assumption about degradation kernels does not hold on real RGB images because some ISP units complicate the problem. To demonstrate the benefits brought in RAW-to-RAW domain only, we train our model on real RGB images with the same settings. The RGB-to-RGB ZIC performs qualitatively worse in Fig. 5, where the repetitive vertical textures are not recovered. Tab. 4 also shows the poorer performance of this RGB version quantitatively. However, our RGB version still performs better than

Variations	**RealSR**			**SR-RAW**		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Global Kernel	31.63	0.912	0.256	32.24	0.911	0.211
RGB	31.91	0.917	0.241	32.38	0.914	0.205
Alternating	32.10	0.923	0.215	32.65	0.924	0.195
Full	**32.58**	**0.926**	**0.212**	**32.89**	**0.927**	**0.190**

Table 4. Quantitative comparisons with different model variations on RealSR [5] and SR-RAW [51] datasets under $4\times$ scale ratio.

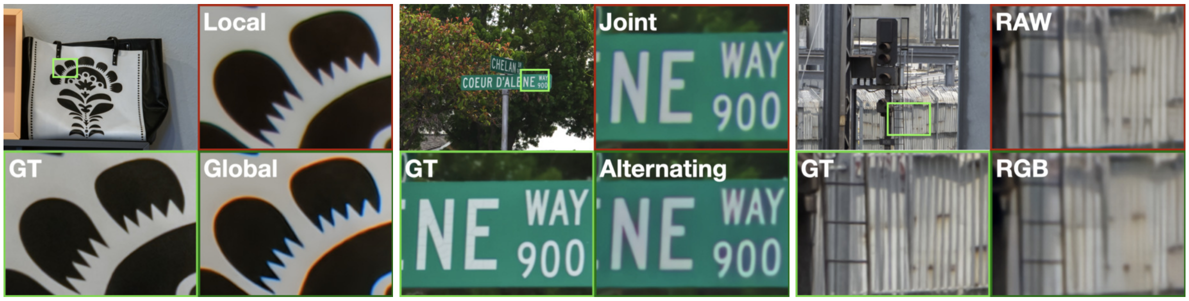


Figure 5. Qualitative comparisons with different model variations.

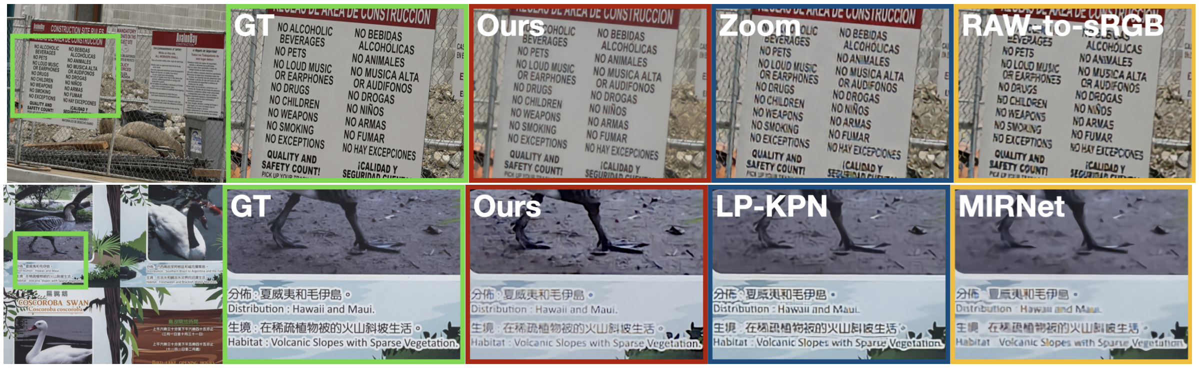


Figure 6. Qualitative comparisons with the RGB-to-RGB and the RAW-to-RGB methods when $d_{\boldsymbol{x}} = 0$ in our method.

the other comparative methods in Sec. 4.2, which demonstrates the technical strength of the algorithm itself.

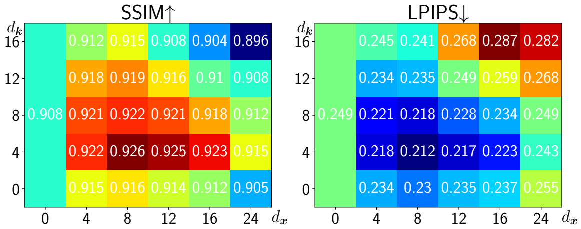


Figure 7. SSIM and LPIPS for different $d_{\boldsymbol{k}}$ and $d_{\boldsymbol{x}}$.

Subspace Dimensions: Lastly, we investigate the impact of the dimensionality of the two generated subspaces, and search for the optimal trade-off between the performance and the efficiency. We alternate the dimensional $d_{\boldsymbol{x}}$ of the image subspace from $\{0, 4, 8, 12, 16, 24\}$ and the dimensional $d_{\boldsymbol{k}}$ of the kernel subspace from $\{0, 4, 8, 12, 16\}$, where $d_{\boldsymbol{k}} = 0$ means the output of the kernel basis generator is directly the kernel which is fixed during the minimization, and $d_{\boldsymbol{x}} = 0$ means the output of the image basis generator is directly the output $\boldsymbol{x}$, and no further minimization is required. Fig. 7 shows that the best performance is achieved when $d_{\boldsymbol{k}} = 4$ and $d_{\boldsymbol{x}} = 8$. So we choose it as the default setting through the paper. When $d_{\boldsymbol{x}} = 0$, our method degenerates to a similar network to [51] but still performs better than the RGB-to-RGB and the RAW-to-RGB methods, both quantitatively in Fig. 7 and qualitatively in Fig. 6. Together with the observation in Sec. 3.2, this advantage of $d_{\boldsymbol{x}} = 0$ also validates our claims because it is *only* contributed by the RAW-to-RAW domain.

5. Conclusion and Discussion

In this work, we proposed the ZIC that zooms an image inside the camera imaging pipeline from RAW to RAW. Previous methods adopt either the RGB-to-RGB domain, or the RAW-to-RGB domain which does not fully take advantage of the RAW format. We have shown that the degradation model between HR-LR images is closer to ideal distributions in RAW, which reduces the difficulty of the kernel estimation. Based on this observation, we estimate the high-resolution output along with the degradation kernel map, which is constrained with learned subspaces. The subspace constrained minimization reformulates the ill-posed SISR problem into a well-posed one. Experiments have shown that our method outperforms previous approaches both in metrics and in visual quality, and have also verified the respective contributions of the minimization and the RAW-to-RAW data domain in ablation studies.

Limitations: Since our model is trained in a supervised manner, its final performance is still limited by the training data, especially the alignment accuracy. However, precise image alignment is difficult for general scenes, and the RAW pixel arrangement of the Bayer pattern even further complicates the problem. We reduce this difficulty in Sec. 4.1 but not fully solve it. Therefore, future works on data capturing, such as using a beam splitter and multiple image sensors, are worth exploring to solve this limitation from the root.

References

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

[2] S. Baker and T. Kanade. Hallucinating faces. In *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, pages 83–88, 2000. 2

[3] S. Baker and T. Kanade. Limits on super-resolution and how to break them. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(9):1167–1183, 2002. 1

[4] Sefi Bell-Kligler, Assaf Shocher, and Michal Irani. Blind super-resolution kernel estimation using an internal-gan. In *Proc. NeurIPS*, 2019. 2

[5] 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 *Proc. CVPR*, pages 3086–3095, 2019. 1, 2, 4, 5, 6, 7

[6] Subhasis Chaudhuri and A. N. Rajagopalan. Depth from defocus: A real aperture imaging approach. In *Springer New York*, 1999. 3

[7] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proc. CVPR*, June 2018. 2

[8] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *Proc. ECCV*, pages 184–199, 2014. 1, 2

[9] Georgios D Evangelidis and Emmanouil Z Psarakis. Parametric image alignment using enhanced correlation coefficient maximization. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30(10):1858–1865, 2008. 5

[10] W.T. Freeman, T.R. Jones, and E.C. Pasztor. Example-based super-resolution. *IEEE Computer Graphics and Applications*, 22(2):56–65, 2002. 2

[11] Michaël Gharbi, Gaurav Chaurasia, Sylvain Paris, and Frédo Durand. Deep joint demosaicking and denoising. *ACM Transactions on Graphics (TOG)*, 35(6):1–12, 2016. 2

[12] Daniel Glasner, Shai Bagon, and Michal Irani. Super-resolution from a single image. In *Proc. ICCV*, pages 349–356, 2009. 1, 2

[13] Andrew S. Glassner. Principles of digital image synthesis. 1995. 4

[14] Jinjin Gu, Hannan Lu, Wangmeng Zuo, and Chao Dong. Blind super-resolution with iterative kernel correction. In *Proc. CVPR*, pages 1604–1613, 2019. 5, 7

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, 2016. 4

[16] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proc. CVPR*, pages 5197–5206, 2015. 2

[17] Andrey Ignatov, Luc Van Gool, and Radu Timofte. Replacing mobile camera isp with a single deep learning model. In *Proc. CVPRW*, pages 536–537, 2020. 2

[18] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Proc. ECCV*, pages 694–711. Springer, 2016. 1, 2, 4

[19] Hamid Reza Vaezi Joze, Ilya Zharkov, Karlton Powell, Carl Ringler, Luming Liang, Andy Roulston, Moshe Lutz, and Vivek Pradeep. Imagepairs: Realistic super resolution dataset via beam splitter camera rig. In *Proc. CVPRW*, pages 518–519, 2020. 1

[20] Kwang In Kim and Younghee Kwon. Single-image super-resolution using sparse regression and natural image prior. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32(6):1127–1133, 2010. 2

[21] E L Kosarev. Shannon’"s superresolution limit for signal recovery. *Inverse Problems*, 6(1):55–76, February 1990. 2

[22] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proc. CVPR*, pages 624–632, 2017. 5

[23] 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 *Proc. CVPR*, pages 4681–4690, 2017. 2

[24] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proc. CVPRW*, pages 136–144, 2017. 1, 2, 5

[25] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, second edition, 2006. 4

[26] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *Proc. CVPR*, 2020. 5

[27] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948. 2

[28] Assaf Shocher, Nadav Cohen, and Michal Irani. “zero-shot” super-resolution using deep internal learning. In *Proc. CVPR*, pages 3118–3126, 2018. 2, 5, 7

[29] Mehrdad Shoeiby, Ali Armin, Sadegh Aliakbarian, Saeed Anwar, and Lars Petersson. Mosaic super-resolution via sequential feature pyramid networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020. 2

[30] Mehrdad Shoeiby, Antonio Robles-Kelly, Ran Wei, and Radu Timofte. PIRM2018 challenge on spectral image super-resolution: Dataset and study. *CoRR*, abs/1904.00540, 2019. 2

[31] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. ICLR*, 2015. 4

[32] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[33] Jian Sun, Zongben Xu, and Heung-Yeung Shum. Image super-resolution using gradient profile prior. In *Proc. CVPR*, pages 1–8. IEEE, 2008. 1

[34] Libin Sun and James Hays. Super-resolution from internet-scale scene matching. In *Proc. ICCP*, pages 1–12, 2012. 2

[35] Chengzhou Tang, Lu Yuan, and Ping Tan. Lsm: Learning subspace minimization for low-level vision. In *Proc. CVPR*, pages 6235–6246, 2020. 2, 3

[36] Radu Timofte, Vincent De Smet, and Luc Van Gool. Anchored neighborhood regression for fast example-based super-resolution. In *Proc. ICCV*, pages 1920–1927, 2013. 2

[37] Markus Unger, Thomas Pock, Manuel Werlberger, and Horst Bischof. A convex approach for variational super-resolution. In *Joint pattern recognition symposium*, pages 313–322. Springer, 2010. 1, 3, 7

[38] 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 *Proc. ECCVW*, September 2018. 1, 2, 5

[39] Zhihao Wang, Jian Chen, and Steven CH Hoi. Deep learning for image super-resolution: A survey. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2020. 1, 2

[40] Pengxu Wei, Ziwei Xie, Hannan Lu, Zongyuan Zhan, Qixiang Ye, Wangmeng Zuo, and Liang Lin. Component divide-and-conquer for real-world image super-resolution. In *Proc. ECCV*, pages 101–117, 2020. 1

[41] Yazhou Xing, Zian Qian, and Qifeng Chen. Invertible image signal processing. In *Proc. CVPR*, 2021. 2

[42] Chih-Yuan Yang, Chao Ma, and Ming-Hsuan Yang. Single-image super-resolution: A benchmark. In *Proceedings of European Conference on Computer Vision*, 2014. 2

[43] Jianchao Yang, John Wright, Thomas Huang, and Yi Ma. Image super-resolution via sparse representation. In *Proc. CVPR*, pages 1–8, 2008. 1, 2

[44] Yuan Yuan, Siyuan Liu, Jiawei Zhang, Yongbing Zhang, Chao Dong, and Liang Lin. Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. In *Proc. CVPRW*, pages 701–710, 2018. 2

[45] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Cycleisp: Real image restoration via improved data synthesis. In *Proc. CVPR*, pages 2696–2705, 2020. 2

[46] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Learning enriched features for real image restoration and enhancement. In *Proc. ECCV*, 2020. 1, 2, 5, 7

[47] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *International conference on curves and surfaces*, pages 711–730, 2010. 2

[48] Kai Zhang, Luc Van Gool, and Radu Timofte. Deep unfolding network for image super-resolution. In *Proc. CVPR*, pages 3217–3226, 2020. 2, 7

[49] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Learning a single convolutional super-resolution network for multiple degradations. In *Proc. CVPR*, pages 3262–3271, 2018. 2, 5, 7

[50] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proc. CVPR*, pages 586–595, 2018. 2, 5

[51] Xuaner Zhang, Qifeng Chen, Ren Ng, and Vladlen Koltun. Zoom to learn, learn to zoom. In *Proc. CVPR*, pages 3762–3770, 2019. 1, 2, 4, 5, 6, 7, 8

[52] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proc. ECCV*, pages 286–301, 2018. 1, 2, 5

[53] Zhilu Zhang, Haolin Wang, Ming Liu, Ruohao Wang, Jiawei Zhang, and Wangmeng Zuo. Learning raw-to-srgb mappings with inaccurately aligned supervision. In *Proc. ICCV*, pages 4348–4358, 2021. 1, 2, 5, 7

[54] Luo Zhengxiong, Yan Huang, Shang Li, Liang Wang, and Tieniu Tan. Unfolding the alternating optimization for blind super resolution. *Advances in Neural Information Processing Systems*, 33, 2020. 7