Self-Supervised Learning for Real-World Super-Resolution from Dual and Multiple Zoomed Observations

Zhilu Zhang, Ruohao Wang, Hongzhi Zhang, and Wangmeng Zuo, *Senior Member, IEEE*

Abstract—In this paper, we consider two challenging issues in reference-based super-resolution (RefSR) for smartphone, (i) how to choose a proper reference image, and (ii) how to learn RefSR in a self-supervised manner. Particularly, we propose a novel self-supervised learning approach for real-world RefSR from observations at dual and multiple camera zooms. Firstly, considering the popularity of multiple cameras in modern smartphones, the more zoomed (telephoto) image can be naturally leveraged as the reference to guide the super-resolution (SR) of the lesser zoomed (ultra-wide) image, which gives us a chance to learn a deep network that performs SR from the dual zoomed observations (DZSR). Secondly, for self-supervised learning of DZSR, we take the telephoto image instead of an additional high-resolution image as the supervision information, and select a center patch from it as the reference to super-resolve the corresponding ultra-wide image patch. To mitigate the effect of the misalignment between ultra-wide low-resolution (LR) patch and telephoto ground-truth (GT) image during training, we first adopt patch-based optical flow alignment to obtain the warped LR, then further design an auxiliary-LR to guide the deforming of the warped LR features. To generate visually pleasing results, we present local overlapped sliced Wasserstein loss to better represent the perceptual difference between GT and output in the feature space. During testing, DZSR can be directly deployed to super-solve the whole ultra-wide image with the reference of the telephoto image. In addition, we further take multiple zoomed observations to explore self-supervised RefSR, and present a progressive fusion scheme for the effective utilization of reference images. Experiments show that our methods achieve better quantitative and qualitative performance against state-of-the-arts. Codes are available at https://github.com/cszhilu1998/SelfDZSR_PlusPlus.

Index Terms—Reference-based super-resolution, self-supervised learning, real world.

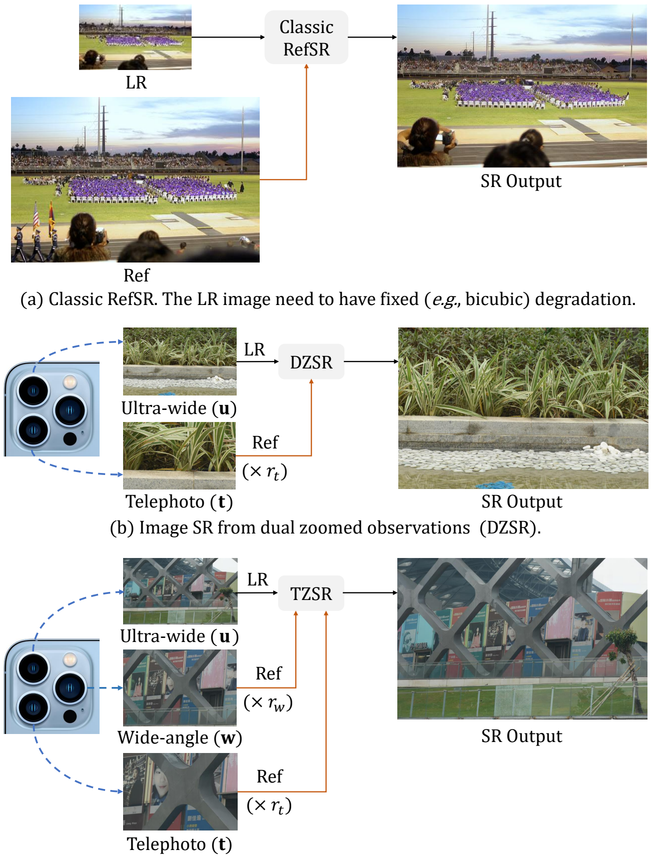
1 Introduction

IMAGE super-resolution (SR) [1]–[5] aiming to recover a high-resolution (HR) image from its low-resolution (LR) counterpart is a severely ill-posed inverse problem with many practical applications. Recently, reference-based image SR (RefSR) [6]–[15] has made progress in relaxing the ill-posedness, which suggests to super-resolve the LR image for more accurate details by leveraging a reference (Ref) image, as shown in Fig. 1(a).

For RefSR, the Ref image should contain similar content and texture with the HR image, and is generally acquired from video frames (*e.g.*, CUFED5 dataset [6]) and web image search (*e.g.*, WR-SR dataset [11]). However, the video frame with high resolution cannot be always got in realistic scenarios, while web image retrieval is time-consuming and sometimes unreliable. It remains a challenging issue to choose a proper Ref image for each LR image, especially in real-world applications. Fortunately, advances and popularity of imaging techniques make it practically feasible to collect images of a scene at different camera zooms. For example, asymmetric cameras with different fixed-focal lenses have been equipped in modern smartphones. In these practical scenarios, the more zoomed (telephoto) image can be naturally leveraged as the reference to guide the SR of the lesser zoomed (ultra-wide) image. Image SR from the dual zoomed observations (DZSR) can thus be carried out, in which Ref has the same scene as the center part of the LR image but higher resolution, as shown in Fig. 1(b).

DZSR is different from classic RefSR methods, but can be still regarded as a special case of RefSR. While conventional RefSR methods [6]–[14] usually use synthetic (*e.g.*, bicubic) degraded LR images for training and evaluation, DZSR should cope with real-world LR ultra-wide images and no ground-truth HR images are available in training. To bridge the domain gap between synthetic and real-world LR images, DCSR [15] suggests a self-supervised real-image adaptation (SRA) strategy, which involves degradation preserving and detail transfer terms. However, DCSR [15] only attains limited success, since the two loss terms in SRA cannot well address the gap between the synthetic and real-world LR degradation as well as the misalignment between ultra-wide and telephoto images. Different from DCSR [15] requiring to pre-train on synthetic images, we introduce self-supervised learning to train DZSR model from scratch directly on ultra-wide and telephoto images, without additional HR images as ground truths (GT). Specifically, we crop the center part of the ultra-wide and telephoto images respectively as the input LR and Ref images, and use the whole telephoto image as the GT during training (see Fig. 2(a)). During inference, by taking the whole ultra-wide

- Z. Zhang is with the Faculty of Computing, Harbin Institute of Technology, Harbin, China. (E-mail: cszlzhang@outlook.com)
- R. Wang is with the Faculty of Computing, Harbin Institute of Technology, Harbin, China. (E-mail: rhwangHIT@outlook.com)
- H. Zhang is with the Faculty of Computing, Harbin Institute of Technology, Harbin, China. (Corresponding author. E-mail: zhanghz0451@gmail.com)
- W. Zuo is with the Faculty of Computing, Harbin Institute of Technology, Harbin, China. (E-mail: wmzuo@hit.edu.cn)



(a) Classic RefSR. The LR image need to have fixed (*e.g.*, bicubic) degradation.

(b) Image SR from dual zoomed observations (DZSR).

(c) Image SR from triple zoomed observations (TZSR).

Fig. 1: Overall pipeline of the classic RefSR, DZSR, and TZSR during inference. r_w and r_t respectively represent the wide-angle and telephoto resolution multiple relative to the ultra-wide image.

and telephoto images respectively as LR and Ref, DZSR can be directly deployed to super-solve the whole ultra-wide image (see Fig. 2(c)).

However, when training DZSR model, the cropped ultra-wide LR image generally cannot be accurately aligned with the telephoto GT image, making the learned model prone to producing blurry SR results [16], [17]. In this case, matching the Ref to LR will also result in the warped Ref being not aligned with the GT, bringing more uncertainty to network training. To handle the misalignment issue, we propose a two-stage alignment method. Firstly, we perform patch-based alignment between LR and GT images by a pre-trained optical flow network. The warped LR is thus obtained, which is roughly aligned with GT. Secondly, we perform finer alignment in the feature space. Specifically, we hope to construct an auxiliary-LR as the target position image for deforming the warped LR features toward GT during training. The auxiliary-LR should be aligned with GT and can be replaced by LR safely during inference. Thus, we carefully design the auxiliary-LR generator network, as well as its position preserving and content preserving constraints. Then an adaptive spatial transformer network (AdaSTN) is used to deform the warped LR features for obtaining final aligned LR features, according to offsets estimated between the warped LR and auxiliary-LR features. When training is done, the optical flow network, auxiliary-LR generator and offset estimator of AdaSTN can be safely detached, bringing no extra cost in the test phase.

For the matching of the Ref image, we perform corresponding contents searching similar to most existing RefSR methods [6], [9]–[11], [15], but it is from Ref to the warped LR image rather than from Ref to the original LR image. Finally, the aligned LR and aligned Ref features can be combined and fed into the restoration module. Furthermore, we present local overlapped sliced Wasserstein (LOSW) loss to optimize the DZSR model. LOSW loss can better measure the perceptual difference between GT and output in the feature space, which is beneficial to the generation of visually pleasing results.

In addition, we bring self-supervised RefSR from multiple zoomed observations into account, and take SR from triple zoomed ones (TZSR) as an example to achieve. For TZSR, the wide-angle image can be utilized as an addition Ref, which resolution is between the resolution of ultra-wide and telephoto images, as shown in Fig 1(c). Following the self-supervised learning of DZSR, self-supervised TZSR can be succeeded. Moreover, to make better use of reference images, we present a progressive fusion scheme for TZSR, where the aligned Ref features (from wide-angle and telephoto images) successively fuse with the aligned LR ones.

Extensive experiments are conducted on the Nikon camera images from the DRealSR dataset [18] as well as the iPhone camera images from the RefVSR dataset [19]. The results demonstrate the effectiveness and practicability of the proposed method for real-world RefSR. In comparison to the state-of-the-art SR and RefSR methods, our method performs favorably in terms of both quantitative metrics and perceptual quality. We also conduct detailed ablation studies, analyzing the effectiveness of different components in the proposed method.

In comparison with the previous version SelfDZSR [20] in ECCV 2022, two main changes (patch-based optical flow alignment and LOSW loss) are introduced to improve the self-supervised learning pipeline for DZSR, while TZSR is newly proposed in this work. The proposed self-supervised learning framework for DZSR and TZSR is named Self-DZSR++ and SelfTZSR++, respectively. To sum up, the main contributions of this work include:

- To achieve real-world RefSR from dual zoomed observations without additional HR images, we propose a self-supervised framework SelfDZSR++.
- To alleviate the adverse effect of image misalignment for self-supervised learning, we propose a two-stage alignment method involving patch-based optical flow alignment and auxiliary-LR guiding alignment, while bringing no extra inference cost.
- To generate visually pleasing results, we present local overlapped sliced Wasserstein loss for measuring perceptual differences better.
- To explore self-supervised RefSR from multiple zoomed observations, SelfDZSR++ is expanded to SelfTZSR++, where we present a progressive fusion scheme for efficient restoration.
- Quantitative and qualitative results on the Nikon and iPhone camera images show that our method outperforms the state-of-the-art methods.

2 Related Work

2.1 Blind Single-Image Super-Resolution

With the development of deep networks, single image super-resolution (SISR) methods based on fixed and known degradation have achieved great success in terms of both performance [1]–[5], [21] and efficiency [22]–[26]. However, these methods perform poorly when applied to images with unknown degradations, and may cause some artifacts. Thus, blind super-resolution comes into being to bridge the gap.

On the one hand, some works estimate the blur kernel or degradation representation for LR and feed it into the SR reconstruction network. IKC [27] performs kernel estimation and SR reconstruction processes iteratively, while DAN [28] conducts it in an alternating optimization scheme. KernelGAN [29] utilizes the image patch recurrence property to estimate an image-specific kernel, and FKP [30] learns a kernel prior based on normalization flow [31] at test time. To relax the assumption that blur kernels are spatially invariant, MANet [32] estimates a spatially variant kernel by the suggested mutual affine convolution. Different from the above explicit methods of estimating kernel, DASR [33] introduces contrastive learning [34] to extract discriminative representations to distinguish different degradations. On the other hand, Hussein *et al*. [35] modify the LR to a pre-defined degradation type (*e.g*., bicubic) by a closed-form correction filter. BSRGAN [36] and Real-ESRGAN [37] design more complex degradation models to generate LR data for training the networks, making the networks generalize well to many scenarios with real-world degradation.

2.2 Real-World Single-Image Super-Resolution

Although blind SR models trained on synthetic data have shown appreciable generalization capacity, the formulated degradation assumption limits the performance on real-world images with much more complicated and changeable degradation. Thus, image SR directly toward real-world scenes has also received much attention. On the one hand, given unpaired real LR and HR, several real-world SR methods [38]–[40] attempt to approximate real degradation and generate the auxiliary-LR image from HR. Then they learn to super-resolve the auxiliary-LR in a supervised manner. On the other hand, some methods [18], [41]–[44] construct paired datasets by adjusting the focal length of a camera, in which the image with a long focal and short focal length is regarded as GT and LR, respectively. In this work, our data collection manner is similar to theirs. The main difference lies in the tasks that need to be performed. We entail training a RefSR model rather than a SISR model, where images with different focal lengths are all required to be taken as input.

In addition, spatial misalignment is a universal problem in real-world paired datasets, and it may cause blurry SR results. The above methods based on paired datasets pre-execute complex alignment or even manual selection, which are generally laborious and time-consuming. Different from them, CoBi [17] loss offers an effective way to deal with misalignment during SR training. Zhang *et al*. [16] incorporates global color mapping and optical flow [45] to explicitly align the data pairs with severe color inconsistency. Nevertheless, optical flow is limited in handling complicated misalignment. In this work, we further propose patch-based optical flow alignment and auxiliary-LR guiding deforming to handle the complicated misalignment after image-based alignment with optical flow.

2.3 Reference-Based Image Super-Resolution

RefSR aims to take advantage of a high-resolution reference image that has similar content and texture as HR image for super-resolution. It relaxes the ill-posedness of SISR and facilitates the generation of more accurate details. The features extracting and matching between LR and Ref is the research focus of most RefSR methods. Among them, Zheng *et al.* [46] proposes a correspondence network to extract features for matching, and an HR synthesis network with the input of the matched Ref. SRNTT [6] calculates the correlation between pre-trained VGG features of LR and Ref at multiple levels for matching them. Zhang *et al*. [47] extend the scaling factor of RefSR methods from $4\times$ to $16\times$. Furthermore, TTSR [9] and FRM [7] develop an end-to-end training framework and proposed learnable feature extractors. C^2-Matching [11] performs a more accurate match by the teacher-student correlation distillation. MASA-SR [10] reduces the computational cost by coarse-to-fine correspondence matching. Recently, Huang *et al*. [12] decouples the RefSR task into SISR and the texture transfer tasks for alleviating reference-underuse and reference-misuse issues, while RRSR [14] introduces the reciprocal learning strategy to improve RefSR models. Besides, CrossNet [48] and SEN [8] respectively introduce optical flow [49] and deformable convolution [50], [51] to align Ref with LR. However, optical flow is limited in handling large and complicated motions while deformable convolution is limited in modeling long-distance correspondence. In this work, we follow [11] to perform patch-wise matching.

Additionally, the RefSR methods mentioned above are all based on bicubic down-sampling. DCSR [15] explores an adaptive fine-tuning strategy on real-world images based on the pre-trained model with synthetic data. In this work, we propose a fully self-supervised learning framework directly on weakly aligned multiple zoomed observations.

3 Proposed Method

In this section, we first introduce our self-supervised learning approach of SelfDZSR++. Then we detail the solutions for alignment between LR and GT, alignment between Ref and LR, restoration module, LOSW loss, and learning objective in SelfDZSR++. Finally, we propose an extension of SelfDZSR++, which utilizes multiple zoomed observations to perform self-supervised RefSR.

3.1 Self-Supervised Learning Framework

Denote by $\mathbf{u}$ and $\mathbf{t}$ the ultra-wide image and the telephoto image, respectively. Super-resolution based on dual zoomed observations aims to super-resolve the ultra-wide image $\mathbf{u}$ with the reference telephoto image $\mathbf{t}$, which can be written as,

$$\hat{\mathbf{y}} = \mathcal{Z}(\mathbf{u}, \mathbf{t}; \Theta_{\mathcal{Z}}), \tag{1}$$

where $\hat{\mathbf{y}}$ has the same field-of-view as $\mathbf{u}$ and the same resolution as $\mathbf{t}$, $\mathcal{Z}$ denotes the zooming network with the parameter $\Theta_{\mathcal{Z}}$.

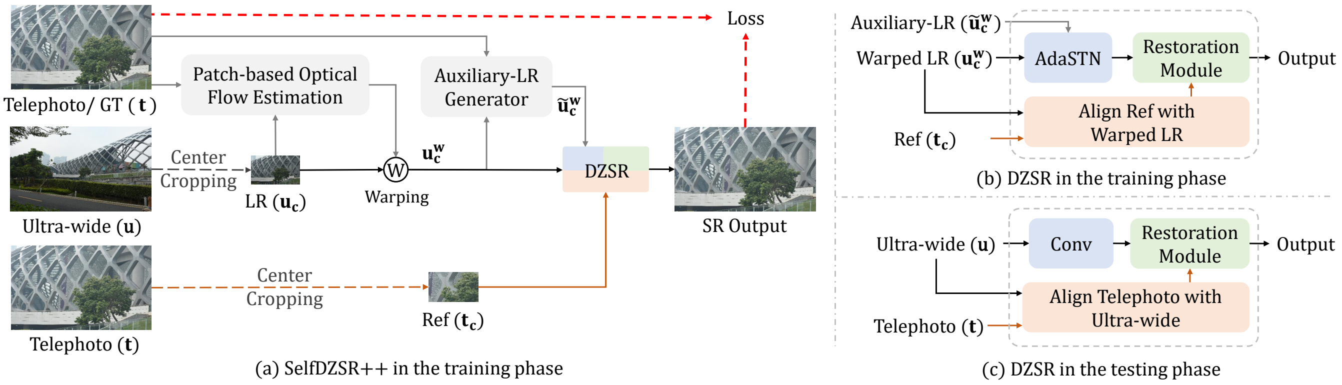

Loss
Telephoto/ GT ($\mathbf{t}$)
Patch-based Optical Flow Estimation
Auxiliary-LR Generator
$\tilde{\mathbf{u}}_\mathbf{c}^\mathbf{w}$
Center Cropping
LR ($\mathbf{u_c}$)
W
Warping
$\mathbf{u}_\mathbf{c}^\mathbf{w}$
DZSR
SR Output
Ultra-wide ($\mathbf{u}$)
Center Cropping
Ref ($\mathbf{t_c}$)
Telephoto ($\mathbf{t}$)
(a) SelfDZSR++ in the training phase
Auxiliary-LR ($\tilde{\mathbf{u}}_\mathbf{c}^\mathbf{w}$)
Warped LR ($\mathbf{u}_\mathbf{c}^\mathbf{w}$)
AdaSTN
Restoration Module
Output
Align Ref with Warped LR
Ref ($\mathbf{t_c}$)
(b) DZSR in the training phase
Ultra-wide ($\mathbf{u}$)
Conv
Restoration Module
Output
Align Telephoto with Ultra-wide
Telephoto ($\mathbf{t}$)
(c) DZSR in the testing phase

Fig. 2: Overall pipeline of proposed SelfDZSR++. (a) SelfDZSR++ in the training phase. The original telephoto image is taken as GT ($\mathbf{t}$), while the center areas of ultra-wide ($\mathbf{u}$) and telephoto ($\mathbf{t}$) images are regarded as LR ($\mathbf{u_c}$) and Ref ($\mathbf{t_c}$) images, respectively. (b) DZSR in the training phase. The auxiliary-LR ($\tilde{\mathbf{u}}_\mathbf{c}^\mathbf{w}$) is aligned with GT and used for deforming the warped LR ($\mathbf{u}_\mathbf{c}^\mathbf{w}$) towards the GT by AdaSTN. Then aligned LR and Ref features are fed into the restoration module. (c) DZSR in the testing phase. The ultra-wide ($\mathbf{u}$) and telephoto ($\mathbf{t}$) images can be regarded as LR and Ref, respectively. Patch-based optical flow alignment and auxiliary-LR generator are detached. AdaSTN is simplified to a convolution layer.

However, in real-world scenarios, the GT of $\hat{\mathbf{y}}$ is hard or almost impossible to acquire. A simple alternative solution is to leverage synthetic data for training, but the domain gaps between the degradation model of synthetic images and that of real-world images prevent it from working well. DCSR [15] tries to bridge the gaps by fine-tuning the trained model using an SRA strategy, but the huge difference in the field of view between the output and the target telephoto images limits it in achieving satisfying results.

In contrast to the above methods, we propose a novel self-supervised dual-zooms super-resolution (SelfDZSR++) framework, which can be trained from scratch solely on the ultra-wide and telephoto image (see Fig. 2(a)), and be directly deployed to the real-world dual zoomed observations (see Fig. 2(c)). During training, we first crop the central area of the ultra-wide and telephoto images,

$$\mathbf{u_c} = \mathcal{C}(\mathbf{u}; r_t), \qquad \mathbf{t_c} = \mathcal{C}(\mathbf{t}; r_t), \tag{2}$$

where $\mathcal{C}$ denotes the center cropping operator, r_t is the focal length ratio between $\mathbf{t}$ and $\mathbf{u}$. Note that $\mathbf{t_c}$ has the same scene and higher resolution with $\mathcal{C}(\mathbf{u_c}; r_t)$, *i.e.*, the central area of $\mathbf{u_c}$. Simultaneously, the resolution of $\mathbf{t}$ is r_t times that of $\mathbf{u_c}$, and their scene is the same. Thus, $\mathbf{u_c}$ and $\mathbf{t}$ can be naturally used as LR and GT respectively, while $\mathbf{t_c}$ can be regarded as the Ref during training. Then we can define DZSR as,

$$\Theta_{\mathcal{Z}}^* = \arg\min_{\Theta_{\mathcal{Z}}} \mathcal{L}\left(\mathcal{Z}(\mathbf{u_c}, \mathbf{t_c}; \Theta_{\mathcal{Z}}), \mathbf{t}\right), \tag{3}$$

where $\mathcal{L}$ denotes the self-supervised learning objective.

Nonetheless, GT $\mathbf{t}$ is not spatially aligned with LR $\mathbf{u_c}$, bringing adverse effects on self-supervised learning. To handle the misalignment issue, we want to align LR to GT as much as possible during training. And we hope such an operation won't affect the inference process. For this purpose, the elaborate design of the framework is essential for SelfDZSR++, which is introduced below.

3.2 Alignment between LR and GT

For aligning LR to GT, we propose a two-stage alignment method. First, we adopt patch-based optical flow alignment to get a warped LR image that is roughly aligned with GT. Then we construct an auxiliary-LR to guide the deformation of warped LR towards GT in the feature space, which is more refined.

3.2.1 Patch-based Optical Flow Alignment

The LR image $\mathbf{u_c}$ and GT image $\mathbf{t}$ in SelfDZSR++ are captured from the different camera lenses, and are generally misaligned in space. When training the model using these pairs, the output would be spatially misaligned with GT, thus leading to inaccurate pixel-wise loss calculation. And it has been shown in recent works [16], [17] that such misalignment will cause the network to generate blurry results. More seriously, the misalignment will result in the warped Ref features being not aligned with GT after matching Ref to LR, bringing more uncertainty to model learning.

Off-the-shelf optical flow [45] offers a probable solution to deal with this issue. But when the image resolution is large (*e.g.*, >1K), the optical flow network sometimes tends to estimate the motion globally and performs poorly on small local contents. Thus, we further adopt patch-based optical flow alignment after image-level alignment, which is carried out on the training patches cropped from the original images. Specifically, we take PWC-Net [45] to calculate the optical flow from the GT patch to the LR one. Then we back warp the LR patch to get the warped LR $\mathbf{u}_\mathbf{c}^\mathbf{w}$ according to the optical flow.

3.2.2 Generation of Auxiliary-LR for Alignment

However, limited to the offset diversity [52] of optical flow, the warped LR $\mathbf{u}_\mathbf{c}^\mathbf{w}$ and GT $\mathbf{t}$ are still slightly misaligned in some complex circumstances (*e.g.*, occlusions caused by scene parallax or moving objects) and explicit perfect alignment is impracticable. To handle the above issues, we hope to construct an auxiliary-LR $\tilde{\mathbf{u}}_\mathbf{c}^\mathbf{w}$ from the GT $\mathbf{t}$ while keeping the spatial position unchanged, and take it to guide the alignment of warped LR towards GT in the feature space (see Fig. 2(a) and (b)). Noted that the auxiliary-LR cannot be used in testing, and it should be substituted by the ultra-wide $\mathbf{u}$ (see Fig. 2(c)).

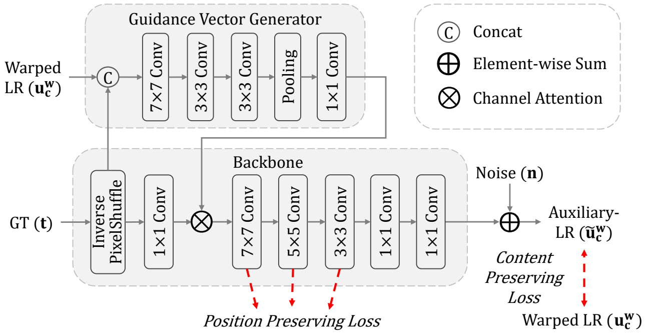


Fig. 3: Illustration of the auxiliary-LR generator. The position preserving loss constraints the kernel weight to ensure the alignment between auxiliary-LR and GT, while content preserving loss constraints that auxiliary-LR has similar contents and degradations as LR.

Thus, the auxiliary-LR $\tilde{\mathbf{u}}_\mathbf{c}^\mathbf{w}$ is required to satisfy two prerequisites. (i) $\tilde{\mathbf{u}}_\mathbf{c}^\mathbf{w}$ can be substituted by $\mathbf{u}$ during testing. (ii) The spatial position of $\tilde{\mathbf{u}}_\mathbf{c}^\mathbf{w}$ should keep the same as $\mathbf{t}$. For the first point, The auxiliary-LR should have similar contents and degradation types as LR, so that it can be substituted safely during testing. In particular, we design an auxiliary-LR generator network and constrain the contents of auxiliary-LR to be similar with these of LR, as shown in Fig. 3. For the second point, inspired by KernelGAN [29], we take advantage of the position preserving loss to constrain the centroid of the convolution kernel in the center of space. The position preserving loss $\mathcal{L}_\mathrm{p}$ can be defined as,

$$\begin{aligned}\mathcal{L}_\mathrm{p}(\mathbf{W}^l) = \|&\sum_{i=0}^{k-1}\sum_{j=0}^{k-1}(i-\frac{k}{2}+0.5)w_{i,j}^l\|_1 \\ +\|&\sum_{i=0}^{k-1}\sum_{j=0}^{k-1}(j-\frac{k}{2}+0.5)w_{i,j}^l\|_1,\end{aligned} \quad (4)$$

where $\mathbf{W}^l$ denotes the kernel weight parameters of the l-th convolution layer in the backbone of the auxiliary-LR generator, k is odd and denotes the kernel size, $w_{i,j}^l$ denotes the value in the (i,j) position of $\mathbf{W}^l$. In addition, warped LR $\mathbf{u_c^w}$ can be used to generate a conditional guidance vector for modulating features of $\mathbf{t}$ globally, which does not affect the preservation of spatial position. Denote by $\mathcal{D}$ the auxiliary-LR generator, its optimization objective can be written as,

$$\Theta_\mathcal{D}^* = \arg\min_{\Theta_\mathcal{D}} \|\mathcal{D}(\mathbf{t},\mathbf{u_c^w};\Theta_\mathcal{D}) - \mathbf{u_c^w}\|_1 + \lambda_p \sum_{l=1}^{L}\mathcal{L}_\mathrm{p}(\mathbf{W}^l), \quad (5)$$

where $\Theta_\mathcal{D}$ is generator's parameter and λ_p is set to 100.

At this time, although the auxiliary-LR already has similar content and degradation as the LR in most cases (see Fig. 11), it may struggle to cover some noise and artifacts that exist in LR image. Thus, the auxiliary-LR sometimes has clearer contents than the LR. When using these auxiliary-LR images to guide the alignment of the warped LR images, the restoration module may be overfitted to auxiliary-LR images, bringing adverse effects to the restoration of LR images. To alleviate the problem, we add some simple perturbations (*e.g.*, noise) to auxiliary-LR. Finally, the auxiliary-LR can be represented as,

$$\tilde{\mathbf{u}}_\mathbf{c} = \mathcal{D}(\mathbf{t},\mathbf{u_c^w};\Theta_\mathcal{D}^*) + \mathbf{n}, \quad (6)$$

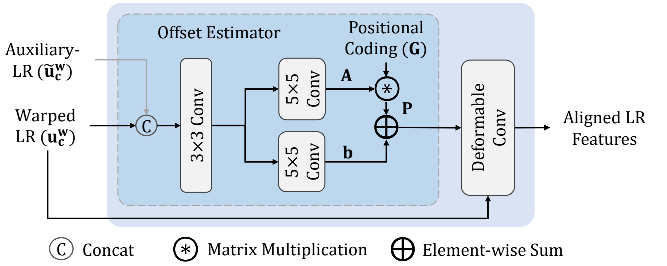


Fig. 4: Illustration of AdaSTN. The offset estimator predicts the position offsets between the warped LR and auxiliary-LR, then deformable convolution is used to deform warped LR according to the offsets.

where $\mathbf{n}$ denotes Gaussian noise and JPEG compression noise. The variance of Gaussian noise is uniformly sampled from 5/255 to 30/255, and the JPEG quality factor is uniformly chosen from 60 to 95.

3.2.3 Aligning Warped LR to Auxiliary-LR

Given warped LR and auxiliary-LR, we suggest implicitly aligning warped LR to auxiliary-LR (aligned with GT). We can estimate the offsets between them and then deform the warped LR features to align with GT. Deformable convolution [50] is a natural choice, but the direct estimation of the offsets may bring instability to the network training. Inspired by [53], we utilize adaptive spatial transformer networks (AdaSTN) that offset is obtained indirectly by estimating the pixel-level affine transformation matrix and translation vector, as shown in Fig. 4. For each pixel, the estimated offset of AdaSTN can be written as,

$$\mathbf{P} = \mathbf{A}\mathbf{G} + \mathbf{b}, \quad (7)$$

where $\mathbf{A} \in \mathbb{R}^{2\times 2}$ is a predicted affine transformation matrix and $\mathbf{b} \in \mathbb{R}^{2\times 1}$ is the translation vector. $\mathbf{G}$ is a positional coding represented by

$$\mathbf{G} = \begin{bmatrix} -1 & -1 & -1 & 0 & 0 & 0 & 1 & 1 & 1 \\ -1 & 0 & 1 & -1 & 0 & 1 & -1 & 0 & 1 \end{bmatrix}. \quad (8)$$

Thus, the deformable convolution of AdaSTN can be formulated as,

$$\mathbf{y}(\mathbf{q}) = \sum\nolimits_{k=0}^{8} \mathbf{w}_k \mathbf{x}(\mathbf{q}+\mathbf{p}_k), \quad (9)$$

where $\mathbf{x}$ and $\mathbf{y}$ represent the input and output features, respectively. $\mathbf{w}_k$ denotes kernel weight and $\mathbf{p}_k$ denotes k-th column value of $\mathbf{P}$. In experiments, we stack 3 AdaSTNs to align warped LR and auxiliary-LR progressively.

Note that auxiliary-LR is not available in the testing phase. We can set $\mathbf{P} = \mathbf{0}$ directly, which means that the deformable convolution of AdaSTN can only observe the input value at the center point of the kernel and AdaSTN degenerates into 1×1 convolution (see Fig. 2(c)). However, this way may produce some artifacts in the results due to the gap between training and testing. In order to bridge this gap, for each AdaSTN, we randomly set $\mathbf{P} = \mathbf{0}$ with probability p (*e.g.*, 0.3) during training. For each training sample, the probability p^3 (*e.g.*, 0.027) that 3 AdaSTNs are all set to $\mathbf{P} = \mathbf{0}$ is low, so it has little impact on the learning of the overall framework.

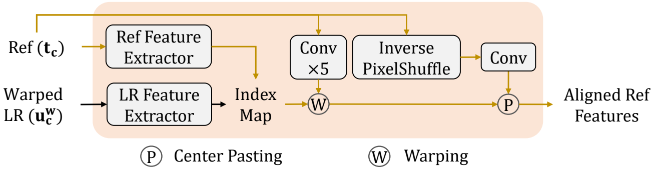


Fig. 5: Alignment between Ref and auxiliary-LR.

3.3 Alignment between Ref and LR

Previous RefSR methods generally perform matching by calculating cosine similarity between Ref and LR features. But for SelfDZSR during training, the misalignment between LR and GT will result in the warped Ref features being not aligned with GT after matching Ref to LR. Given that warped LR $\mathbf{\tilde{u}_c^w}$ is already roughly aligned with GT $\mathbf{t}$, we instead calculate the correlation between Ref and warped LR features (see Fig. 2(b)). During testing, the warped LR $\mathbf{\tilde{u}_c^w}$ can be substituted by the ultra-wide image $\mathbf{u}$ (see Fig. 2(c)).

Fig. 5 shows the detailed alignment scheme between Ref and warped LR. The index map is obtained by calculating the cosine similarity between Ref and warped LR features that are extracted by pre-trained feature extractors. Then the Ref is warped according to the index map. In addition, for SelfDZSR, the central part of LR has the same scene as Ref. Taking this property into account, we can rearrange Ref elements by an inverse PixelShuffle [54] layer, and then paste it to the center area of the warped Ref features.

3.4 Restoration Module

After getting the aligned LR features (introduced in Sec. 3.2) and aligned Ref features (introduced in Sec. 3.3), we feed them into the restoration module. Fig. 6(a) shows the detailed structure of the restoration module. First, the aligned LR and aligned Ref features are concatenated and fed into the backbone, which consists of 16 residual blocks [3]. Then the concatenated features, Ref image, and central area of warped LR image are input into an encoder to generate vectors that modulate the features of each residual block. This modulation can be regarded as channel attention on the features of the residual block. And it is beneficial to relieve the color inconsistency (see Fig. 6(b)) between the real-world ultra-wide and telephoto images during testing.

3.5 LOSW Loss and Learning Objective

The sliced Wasserstein (SW) distance has exhibited outstanding merit for training deep generative networks [55], [56]. Recently, SW loss has been successfully applied in texture synthesis [57], image enhancement [58], image quality assessment [59] and *etc*. And we also utilize SW loss to optimize our model in the earlier version SelfDZSR [20]. However, here we find that although it brings sharper results, it also leads to more artifacts. Thus, we improve SW loss and present local overlapped SW (LOSW) loss $\mathcal{L}_{\mathrm{LOSW}}$ to optimize SelfDZSR++ in this work.

Algorithm 1 Pseudocode of LOSW loss

```
Input: $\mathbf{U} \in \mathbb{R}^{C \times K \times K}$: VGG features of output image;
       $\mathbf{V} \in \mathbb{R}^{C \times K \times K}$: VGG features of target image;
       $\mathbf{M} \in \mathbb{R}^{C' \times C}$: random projection matrix;
Output: $\mathcal{L}_{\mathrm{SW}}(\mathbf{U}, \mathbf{V})$: the value of LOSW loss;
Unfold features $\mathbf{U}$ and $\mathbf{V}$ to overlapped patch $\mathbf{U_p} (\in \mathbb{R}^{P \times C \times k \times k})$ and $\mathbf{V_p} (\in \mathbb{R}^{P \times C \times k \times k})$ with kernel size $k$, respectively; $P$ denotes the number of patches;
Flatten features $\mathbf{U_p}$ and $\mathbf{V_p}$ to $\mathbf{U_f} (\in \mathbb{R}^{P \times C \times k^2})$ and $\mathbf{V_f} (\in \mathbb{R}^{P \times C \times k^2})$, respectively;
Project the features onto $C'$ directions: $\mathbf{U_d} = \mathbf{M} \mathbf{U_f}$, $\mathbf{V_d} = \mathbf{M} \mathbf{V_f}$;
Sort projections for each direction: $\mathbf{U_s} = \mathbf{Sort}(\mathbf{U_d}$, dim=2), $\mathbf{V_s} = \mathbf{Sort}(\mathbf{V_d}$, dim=2);
$\mathcal{L}_{\mathrm{LOSW}}(\mathbf{U}, \mathbf{V}) = \|\mathbf{U_s} - \mathbf{V_s}\|_1$
```

The algorithm of LOSW loss is described in Alg. 1. We first divide output and target VGG [60] features ($\mathbf{U}$ and $\mathbf{V}$) into overlapped small patches ($\mathbf{U_p}$ and $\mathbf{V_p}$), then obtain the patch representation ($\mathbf{U_d}$ and $\mathbf{V_d}$) through random linear projection. Finally, we calculate the Wasserstein distance between the output and the target patch representation, which is defined as the element-wise ℓ_1 distance over sorted patch representation ($\mathbf{U_s}$ and $\mathbf{V_s}$). LOSW loss and SW loss have different focuses in terms of feature similarity. Specifically, LOSW loss emphasizes the similarity of feature distribution in the local area, while SW loss focuses on that globally. Consequently, LOSW loss can encourage the output to be more faithful to the target image at the local level, and can also help reduce artifacts to some extent. SelfDZSR++ is jointly optimized with ℓ_1 loss and LOSW loss. The total loss term can be written as,

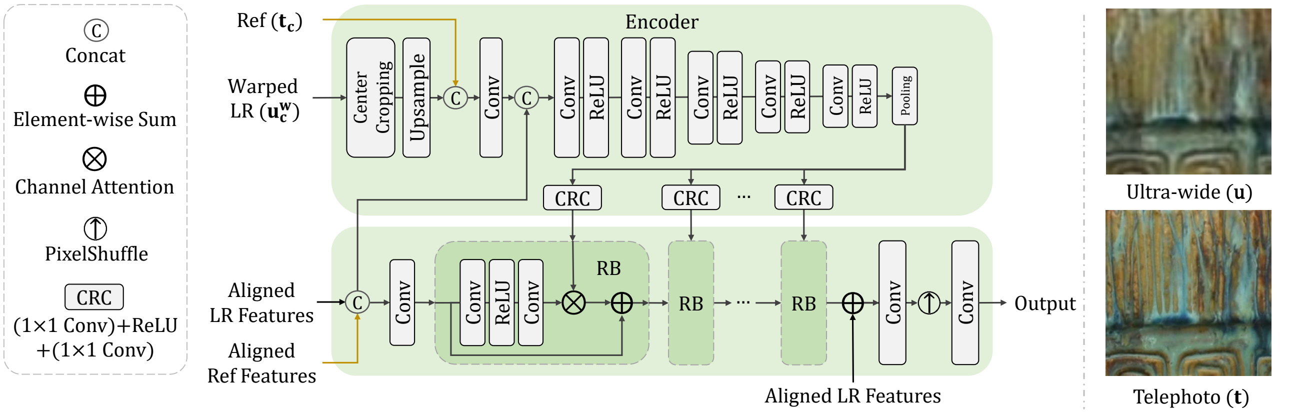
$$\mathcal{L}_{\mathrm{total}}(\mathbf{\hat{y}}, \mathbf{t}) = \|\mathbf{\hat{y}} - \mathbf{t}\|_1 + \lambda_{LOSW} \mathcal{L}_{\mathrm{LOSW}}(\phi(\mathbf{\hat{y}}), \phi(\mathbf{t})), \quad (10)$$

where ϕ denotes the pre-trained VGG-19 [60] network, and we set $\lambda_{LOSW} = 0.08$.

3.6 Extension to Multiple Zoomed Observations

With the introduction in the previous sections (Sec. 3.1 ∼ Sec. 3.5), we can utilize dual zoomed observations (ultra-wide and telephoto images) for real-world RefSR in a self-supervised manner. In recent times, modern smartphones are being outfitted with not just two, but multiple lenses with different focal lengths. This enables us to capture multiple images simultaneously with varying focal lengths. Consequently, it is natural and significant to extend our method to multiple zoomed observations. In this subsection, we take image SR from triple zoomed observations (TZSR) as an example and introduce SelfTZSR++. Especially, we emphatically explore the fusion restoration scheme with Ref images at different focal lengths.

Self-Supervised Learning for TZSR. Some modern smartphones come equipped with three lenses that allow users to capture images at different focal lengths, including ultra-wide, wide-angle, and telephoto shots. As the focal length increases, the resolution of the image increases, while its field of view (FOV) gradually narrows. The proposed DZSR utilizes telephoto image $\mathbf{t}$ as a reference to super-resolve ultra-wide image $\mathbf{u}$. It's worth noting that the wide-angle image still has a higher resolution than the ultra-angle image, and potentially makes up for the lack of the telephoto image with a narrow FOV. To further improve SR results,



(a) Structure of the restoration model for DZSR

(b) Color inconsistency

Fig. 6: (a) The detailed structure of the restoration model for DZSR. ‘RB’ denotes the residual block [3]. (b) An example of color inconsistency between ultra-wide and telephoto images.

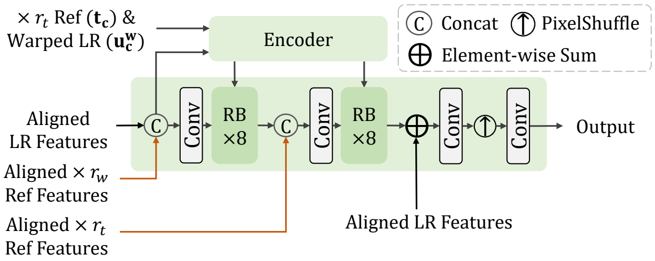

$\times r_t$ Ref ($\mathbf{t_c}$) &
Warped LR ($\mathbf{u_c^w}$)
Encoder
Concat
PixelShuffle
Element-wise Sum
Aligned LR Features
Aligned $\times r_w$ Ref Features
Aligned $\times r_t$ Ref Features
Conv
RB ×8
Conv
RB ×8
Conv
Conv
Output
Aligned LR Features

Fig. 7: Structure of the restoration model for TZSR. The aligned $\times r_w$ and $\times r_t$ features are from the central areas of the wide-angle $\mathbf{w}$ and telephoto $\mathbf{t}$ image, respectively.

TZSR aims to introduce the wide-angle image $\mathbf{w}$ as an additional reference, as shown in Fig 1(c). For self-supervised training of TZSR, we introduce SelfTZSR++ based on SelfDZSR++. Specifically, SelfTZSR first crops the central area of the wide-angle image $\mathbf{w}$,

$$\mathbf{w_c} = \mathcal{C}(\mathbf{w}; r_w), \tag{11}$$

where r_w is the focal length ratio between $\mathbf{w}$ and $\mathbf{u}$. Compared to the low-resolution image $\mathbf{u_c}$, the telephoto image $\mathbf{t_c}$ can be treated as a reference with a higher resolution of $\times r_t$, while the wide-angle image $\mathbf{w_c}$ can serve as another reference with a resolution of $\times r_w$. Then we can define TZSR as,

$$\Theta_{\mathcal{Z}} = \arg\min_{\Theta_{\mathcal{Z}}} \mathcal{L}\left(\mathcal{Z}(\mathbf{u_c}, \mathbf{w_c}, \mathbf{t_c}; \Theta_{\mathcal{Z}}), \mathbf{t}\right), \tag{12}$$

which is modified from Eqn. (3). SelfTZSR++ is also jointly optimized with ℓ_1 loss and LOSW loss. The total loss term is the same as Eqn. (10).

Progressive Fusion Restoration. The aligned LR features, as well as the aligned $\times r_w$ Ref and $\times r_t$ Ref features can be obtained following the methods introduced in Sec. 3.2 and Sec. 3.3, respectively. Here, we focus on how these features can be processed for better restoration. Perhaps due to the information bottleneck, it can not achieve the best performance when directly concatenating these features to feed the restoration network. Instead, we propose a progressive fusion scheme in which aligned LR features are sequentially fused with two Ref features. Specifically, as shown in Fig. 7, we first concatenate $\times r_w$ Ref features (which have a lower resolution than the $\times r_t$ Ref ones) with the LR ones, and process them. Then we merge $\times r_t$ Ref features with the processed features for further modulation. Through the progressive utilization of Refs with different resolutions, the effect of image SR can be gradually improved.

4 Experiments

4.1 Experimental Setup

Datasets. Experiments are conducted on Nikon camera images from DRealSR dataset [18] and iPhone camera images from RefVSR dataset [19]. The training patches of DRealSR have been manually and carefully selected for mitigating the alignment issue, which is laborious and time-consuming. Instead, we take the originally captured images for training, making the whole process fully automated. In particular, each scene of the original data contains four different focal-length images. We adopt the shortest focal-length image, the second shortest focal-length one, and the longest focal-length one as the ultra-wide, wide-angle, and telephoto images, respectively. There are 163 image pairs for training and 20 for evaluation. RefVSR [15] dataset is collected by iPhone 12 Pro Max, which provides three videos with different fixed focal lengths (ultra-wide, wide-angle, and telephoto) for each scene. We remove blurry frames, and treat the video frames as single images. There are 13893 image pairs for training and 1024 for evaluation. For simplicity, we resize the above images so that r_w and r_t are 2 and 4, respectively.

Training Configurations. We augment the training data with random horizontal flip, vertical flip and 90° rotation. The batch size is 16, and the patch size for LR is 48×48. The model is trained with the Adam optimizer [61] by setting $\beta_1 = 0.9$ and $\beta_2 = 0.999$ for 400 epochs. The learning rate is initially set to 1×10^{-4} and is decayed to 5×10^{-5} after 200 epochs. The experiments are conducted with PyTorch [62] on an Nvidia GeForce RTX 3090 GPU.

Evaluation Configurations. No ground-truths can be utilized when inputting the ultra-wide and telephoto images directly, leading to quantitative evaluation difficult. As a result, we still use the center areas of ultra-wide and telephoto images as LR and Ref, respectively. Then we align the whole telephoto image with the output by optical flow network [45]. Quantitative metrics (*i.e.*, PSNR, SSIM [63] and

TABLE 1: Quantitative results of SR models trained only with ℓ_1 (or ℓ_2) loss. The best and second-best results are masked by red and blue colors, respectively. RefSR† represents that the RefSR methods are trained in our self-supervised learning manner.

		PSNR↑ / SSIM↑ / LPIPS↓			
	Method	**Nikon Camera Images**		**Iphone Camera Images**	
		Full-Image	*Corner-Image*	*Full-Image*	*Corner-Image*
SISR	EDSR [3]	27.26 / 0.8364 / 0.362	27.29 / 0.8345 / 0.363	23.54 / 0.7224 / 0.376	23.54 / 0.7236 / 0.361
	RCAN [4]	27.30 / 0.8344 / 0.383	27.33 / 0.8323 / 0.383	23.51 / 0.7249 / 0.376	23.50 / 0.7261 / 0.361
	CDC [18]	27.20 / 0.8306 / 0.412	27.24 / 0.8283 / 0.412	23.49 / 0.7167 / 0.433	23.49 / 0.7177 / 0.417
RefSR†	SRNTT-ℓ_2 [6]	27.30 / 0.8387 / 0.359	27.33 / 0.8366 / 0.359	23.60 / 0.7194 / 0.419	23.59 / 0.7205 / 0.403
	TTSR-ℓ_1 [9]	25.83 / 0.8272 / 0.369	25.80 / 0.8259 / 0.369	23.48 / 0.7199 / 0.370	23.47 / 0.7210 / 0.355
	C^2-Matching-ℓ_1 [11]	27.19 / 0.8402 / 0.362	27.23 / 0.8381 / 0.362	23.51 / 0.7175 / 0.391	23.51 / 0.7185 / 0.375
	MASA-ℓ_1 [10]	27.27 / 0.8372 / 0.339	27.30 / 0.8352 / 0.339	23.53 / 0.7196 / 0.391	23.51 / 0.7208 / 0.375
	DCSR-ℓ_1 [15]	27.73 / 0.8274 / 0.355	27.72 / 0.8275 / 0.349	23.23 / 0.7173 / 0.383	23.22 / 0.7186 / 0.367
Ours	SelfDZSR-ℓ_1 [20]	28.93 / 0.8572 / 0.308	28.67 / 0.8457 / 0.328	23.79 / 0.7396 / 0.320	23.52 / 0.7243 / 0.325
	SelfDZSR++-ℓ_1	29.63 / 0.8663 / 0.290	29.36 / 0.8555 / 0.309	24.08 / 0.7502 / 0.332	23.81 / 0.7347 / 0.338
	SelfTZSR++-ℓ_1	29.74 / 0.8708 / 0.280	29.47 / 0.8617 / 0.295	24.36 / 0.7680 / 0.303	24.10 / 0.7576 / 0.305

TABLE 2: Quantitative results of SR models trained with their all loss terms. The best and second-best results are masked by red and blue colors, respectively. RefSR† represents that the RefSR methods are trained in our self-supervised learning manner.

		PSNR↑ / SSIM↑ / LPIPS↓			
	Method	**Nikon Camera Images**		**Iphone Camera Images**	
		Full-Image	*Corner-Image*	*Full-Image*	*Corner-Image*
SISR	BSRGAN [36]	26.91 / 0.8151 / 0.279	26.96 / 0.8135 / 0.278	22.15 / 0.6833 / 0.313	22.14 / 0.6844 / 0.298
	Real-ESRGAN [37]	25.96 / 0.8076 / 0.272	26.00 / 0.8063 / 0.271	21.78 / 0.6847 / 0.311	21.77 / 0.6859 / 0.296
RefSR†	SRNTT [6]	27.31 / 0.8242 / 0.286	27.35 / 0.8223 / 0.283	23.29 / 0.6833 / 0.349	23.28 / 0.6847 / 0.334
	TTSR [9]	25.31 / 0.7719 / 0.282	25.27 / 0.7708 / 0.282	22.40 / 0.6514 / 0.342	22.39 / 0.6528 / 0.325
	C^2-Matching [11]	26.79 / 0.8141 / 0.327	26.81 / 0.8123 / 0.325	22.57 / 0.6725 / 0.349	22.56 / 0.6734 / 0.331
	MASA [10]	27.32 / 0.7640 / 0.273	27.37 / 0.7615 / 0.274	21.75 / 0.6277 / 0.324	21.75 / 0.6291 / 0.308
	DCSR [15]	27.69 / 0.8232 / 0.276	27.68 / 0.8232 / 0.272	23.08 / 0.7014 / 0.308	23.07 / 0.7029 / 0.294
Ours	SelfDZSR [20]	28.67 / 0.8356 / 0.219	28.42 / 0.8238 / 0.231	23.37 / 0.7128 / 0.252	23.12 / 0.6987 / 0.250
	SelfDZSR++	29.30 / 0.8511 / 0.201	29.03 / 0.8401 / 0.213	23.64 / 0.7266 / 0.244	23.38 / 0.7105 / 0.247
	SelfTZSR++	29.62 / 0.8582 / 0.187	29.37 / 0.8490 / 0.196	24.00 / 0.7466 / 0.215	23.76 / 0.7358 / 0.213

TABLE 3: Model #parameters and #FLOPs comparison of SISR and RefSR methods. The #FLOPs is measured when ×4 super-resolving LR image to 1280 × 720 resolution. For RefSR methods, the Ref image has the same size with LR.

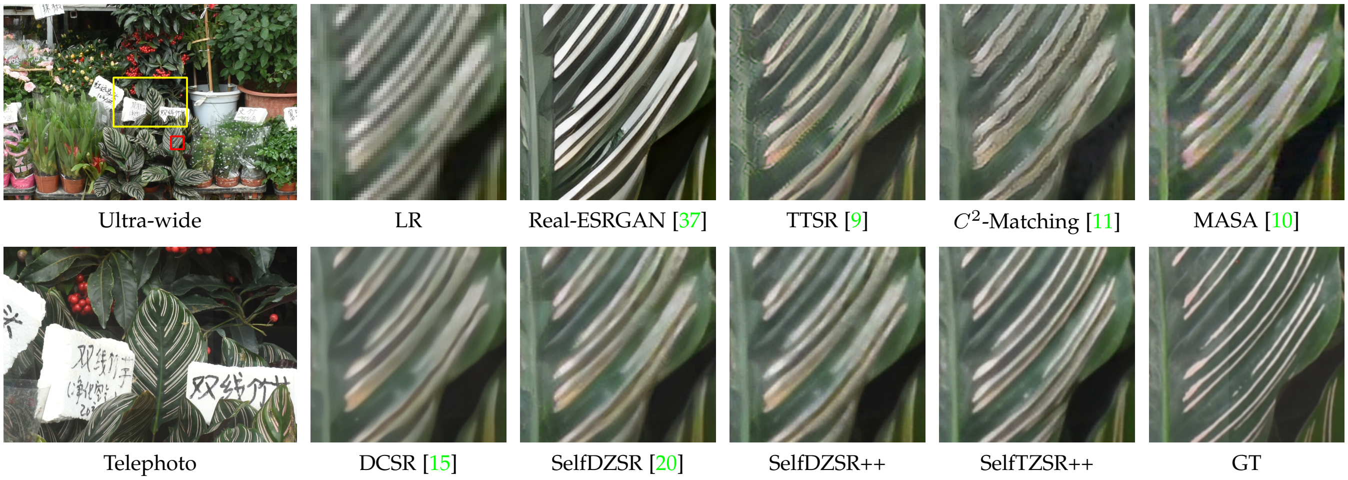
	Method	# Params (M)	#FLOPs (G)
SISR	EDSR [3]	43.1	5792
	RCAN [4]	15.6	1838
	CDC [18]	39.9	1626
	BSRGAN [36]	16.7	2068
	Real-ESRGAN [37]	16.7	2068
RefSR	SRNTT [6]	5.5	3568
	TTSR [9]	7.3	2468
	C^2-Matching [11]	8.9	1968
	MASA [10]	4.0	1984
	DCSR [15]	3.2	836
Ours	SelfDZSR [20]	3.1	454
	SelfDZSR++	3.1	454
	SelfTZSR++	3.3	538

LPIPS [64]) can be calculated between the output and the aligned telephoto image. Noted that the scene of the Ref is the center area of the telephoto image. In addition to calculating the metrics on the full image (marked as *Full-Image*), we also calculate the metrics of the areas excluding the center (marked as *Corner-Image*). And all patches for visual comparison are selected from the areas excluding the center of the output.

4.2 Quantitative and Qualitative Results

We compare results with SISR (*i.e.*, EDSR [3], RCAN [4], CDC [18], BSRGAN [36] and Real-ESRGAN [37]) and RefSR (*i.e.*, SRNTT [6], TTSR [9], MASA [10], C^2-Matching [11], DCSR [15] and our SelfDZSR [20]) methods. The results of BSRGAN and Real-ESRGAN are generated via the officially released model, other methods are retrained using our images for a fair comparison. Among them, RefSR methods are trained in our self-supervised learning manner and each method has two models, obtained by minimizing ℓ_1 (or ℓ_2) loss and all loss terms that are used in their papers.

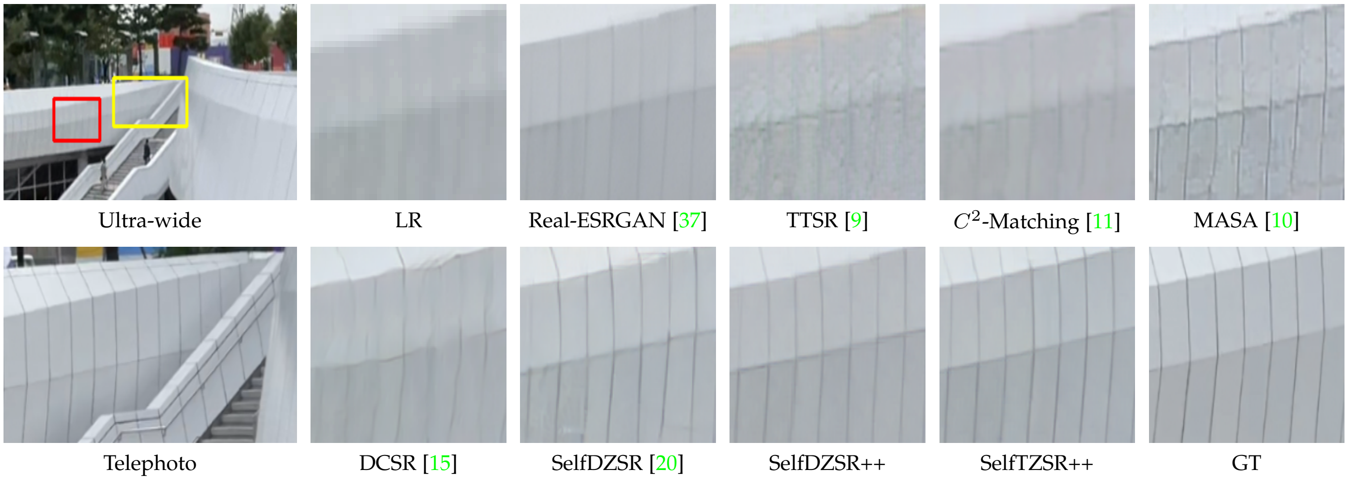
Table 1 and Table 2 show the quantitative results of SR models trained with ℓ_1 (or ℓ_2) loss and their all loss terms, respectively. From the tables, SelfDZSR [20] has exceeded most previous SISR and RefSR methods, benefiting from the alignment of data pairs and effective utilization of Ref information. Due to the better handling of the alignment problem and the proposal of LOSW loss, SelfDZSR++ outperforms



Ultra-wide LR Real-ESRGAN [37] TTSR [9] C^2-Matching [11] MASA [10]

Telephoto DCSR [15] SelfDZSR [20] SelfDZSR++ SelfTZSR++ GT

Fig. 8: Visual comparison on Nikon camera images. In the ultra-wide image, the yellow box indicates the overlapped scene with the telephoto image, while the red box represents the selected LR patch.



Ultra-wide LR Real-ESRGAN [37] TTSR [9] C^2-Matching [11] MASA [10]

Telephoto DCSR [15] SelfDZSR [20] SelfDZSR++ SelfTZSR++ GT

Fig. 9: Visual comparison on iPhone camera images. In the ultra-wide image, the yellow box indicates the overlapped scene with the telephoto image, while the red box represents the selected LR patch.

SelfDZSR on most metrics. Moreover, with the further introduction of additional Ref and progressive fusion scheme, SelfTZSR++ exceeds all competing methods both in terms of fidelity and perception.

The visual comparison on Nikon and iPhone camera images can be seen in Fig. 8 and Fig. 9, respectively. Our results usually restore more fine-scale textures, and are clearer and more photo-realistic.

4.3 Comparison of #Parameters and #FLOPs

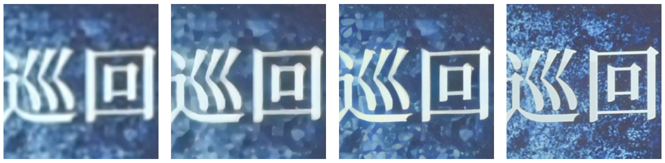
We also compare the number of parameters and FLOPs of different models, as shown in Table 3. For RefSR methods, the cost of calculating the similarity between LR and Ref occupies a large part of the computational cost. In this work, we calculate cosine similarity between ×4 down-sampled Ref and ×4 down-sampled LR features, and find that its performance is close to that of computing similarity at the original image size. By virtue of the lightweight restoration model and the fast similarity calculation, our method has low #parameters and #FLOPs in comparison to both SISR and RefSR methods.

5 Ablation Study

In this section, we conduct ablation experiments for assessing the effect of self-supervised learning, alignment between LR and GT, LOSW loss, different Refs, and fusion scheme. Unless otherwise stated, experiments are carried out on the Nikon camera images [18] with SelfTZSR++, and the metrics are evaluated on full images.

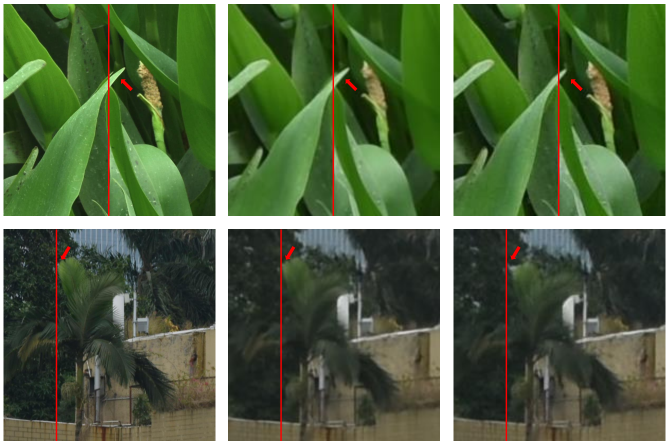
5.1 Effect of Self-Supervised Learning

In order to verify the effectiveness of our proposed self-supervised approach (see Sec. 3.1), we conduct experiments on different training strategies. First, we remove the two-stage alignment components and AdaSTN in SelfDZSR++. Then we replace the real-world LR image with the bicubic downsampling GT image, and retrain the network. Finally, for a fair comparison, we take the self-supervised real-image adaptation (SRA) [15] strategy and our self-supervised method to fine-tune the above model, respectively. As can be seen from Table 4, when evaluating on real-world images, our proposed self-supervised method achieves better results. The PSNR metric is 1.03 dB higher than the model



Bicubic SRA [15] Ours GT

Fig. 10: Visual result comparison when using different training strategies. Our result is sharper and clearer.



(a) GT (b) Auxiliary-LR with no noise (c) LR

Fig. 11: Visual results of noisy-free auxiliary-LR image. The auxiliary-LR has similar contents as LR and is aligned with GT. The red lines and arrows in the same row are in the same position relative to the image.

based on SRA fine-tuning. From Fig. 10, our visual result is sharper and clearer. In a word, it can be seen that even if the misalignment between LR and GT is not handled, our self-supervised method is still better than SRA [15] strategy.

5.2 Effect of Alignment between LR and GT

Effect of Two-stage Alignment. In order to evaluate the effect of our two-stage alignment method (see Sec. 3.2), we first remove patch-based optical flow alignment and auxiliary-LR guiding alignment to train a baseline model in our self-supervised manner. Then we add the alignment method of these two stages in turn to experiment. When taking patch-based optical flow alignment only, the PSNR increases by 0.33 dB against the baseline, as shown in Table 5. Coupled with auxiliary-LR guiding alignment, better quantitative results can be further attained.

Effect of Auxiliary-LR Generator. We show the auxiliary-LR images before adding synthetic noise $\mathbf{n}$ in Fig. 11(b). And the corresponding LR and GT images are shown in Fig. 11(a) and Fig. 11(c), respectively. The red lines and arrows in the same row are in the same position relative to the image. It can be seen that the auxiliary-LR has similar contents as LR and is aligned with GT. It indicates that the function of the auxiliary-LR generator is guaranteed. In addition, we conduct an experiment that adds noise $\mathbf{n}$ to bicubic downsampling GT and replaces auxiliary-LR with it. In this case, PSNR drops by 0.88 dB, and LPIPS gets worse by 0.078. The result shows the auxiliary-LR generator is necessary and effective. We also conduct experiments on different coefficients (*i.e.*, λ_p) of position preserving loss, as shown in Table 6. In order to bring auxiliary-LR into play better on alignment and obtain better SR performance, we take a trade-off between content preserving loss and position preserving loss, and set λ_p to 100.

TABLE 4: Ablation study on training strategies.

Training Strategy	PSNR↑ / SSIM↑ / LPIPS↓
Bicubic Degradation Pre-training	28.08 / 0.8357 / 0.397
SRA Fine-tuning [15]	28.11 / 0.8109 / 0.268
Our Fine-tuning	29.14 / 0.8511 / 0.185

TABLE 5: Ablation study on two-stage alignment methods.

Patch-based Optical Flow	Auxiliary-LR Guiding	PSNR↑ / SSIM↑ / LPIPS↓
×	×	29.19 / 0.8495 / 0.199
✓	×	29.52 / 0.8533 / 0.186
✓	✓	29.62 / 0.8582 / 0.187

TABLE 6: Ablation study on loss terms of auxiliary-LR generator. λ_p denotes the coefficient of position preserving loss in Eqn. (5).

λ_p	0	1	100	10000
PSNR↑	29.37	29.43	29.62	29.61

TABLE 7: Ablation study on the noise of auxiliary-LR.

Noise	PSNR↑ / SSIM↑ / LPIPS↓
None	29.39 / 0.8550 / 0.233
JPEG	29.26 / 0.8523 / 0.198
Gaussian	29.61 / 0.8560 / 0.187
Gaussian and JPEG	29.62 / 0.8582 / 0.187

TABLE 8: Ablation study on AdaSTN.

Method	PSNR↑ / SSIM↑ / LPIPS↓
Baseline	29.52 / 0.8533 / 0.186
Baseline + Deformable Conv [50]	29.60 / 0.8582 / 0.191
Baseline + AdaSTN	29.62 / 0.8582 / 0.187

Effect of Noise in Auxiliary-LR. Some noise is added to auxiliary-LR to prevent overfitting problems of the restoration module. Gaussian noise is a natural choice, and we empirically find it is sufficient to achieve the goal, as shown in Table 7. Moreover, additional JPEG compression noise can provide further slight improvement, which can be regarded as simulated artifacts.

Effect of AdaSTN. We regard the model only using patch-based optical flow alignment as the baseline. And we modify AdaSTN to deformable convolution [50] to verify its effect. Specifically, instead of calculating the offset by estimating the affine transformation matrix and vector according to Eqn. (7), we directly estimate the offset for deformable convolution. The results in Table 8 indicate that AdaSTN has superior performance than deformable convolution.

5.3 Effect of LOSW Loss

Most RefSR methods [6], [9]–[11] adopt VGG-based [60] perceptual loss and adversarial loss [66] for more realistic results. Here we follow C^2-Matching [11] to train SelfTZSR++

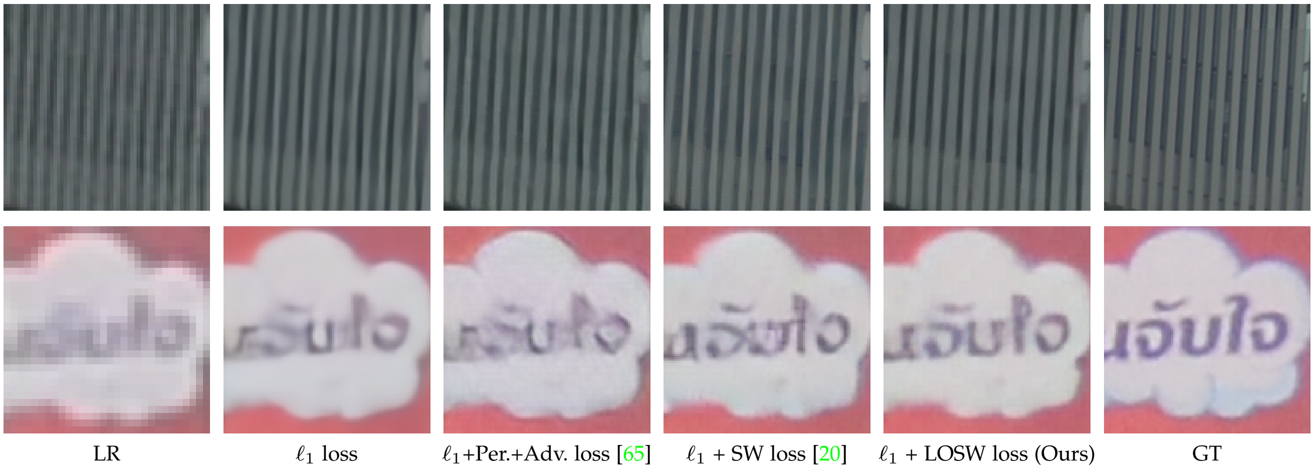


Fig. 12: Visual results comparison when using different loss terms. 'Per.' and 'Adv.' represent perceptual and adversarial loss terms, respectively. The textures and details in our results are more realistic.

TABLE 9: Quantitative results comparison while using different loss terms.

Loss Terms	PSNR↑ / SSIM↑ / LPIPS↓
ℓ_1	29.74 / 0.8708 / 0.280
ℓ_1 + Perceptual + Adversarial [65]	29.27 / 0.8463 / 0.213
ℓ_1 + SW [20]	29.20 / 0.8479 / 0.188
ℓ_1 + LOSW	29.62 / 0.8582 / 0.187

using a combination of ℓ_1 reconstruction loss, perceptual loss, and adversarial loss based on Relativistic GAN [65]. The quantitative results are shown in Table 9. It can be seen that the model trained by LOSW loss obtains a 0.35 dB PSNR and 0.026 LPIPS gain than that by adversarial loss. We also take SW [20] loss to train a model for comparison. The model trained by LOSW loss also has a higher PSNR metric than that by SW loss, while the gap of the LPIPS metric is small. In comparison with the model only using ℓ_1 loss, the PSNR of the model using LOSW loss is only 0.12 dB worse, while LPIPS has an advantage of 0.093. Unlike other loss terms that drop fidelity metrics when obtaining better perceptual metrics, LOSW loss has a superior ability to measure perceptual differences while maintaining fidelity.

Fig. 12 shows a visual result comparison when using different loss terms. When only ℓ_1 loss is taken, the result is over-smooth. Although adversarial loss and SW loss bring sharper content, they lead to some unrealistic artifacts. LOSW can help generate more satisfactory images which are more faithful to the high-resolution ground-truth and has fewer artifacts. In short, LOSW loss can achieve a better trade-off in fidelity and perception.

5.4 Effect of Different Refs and Fusion Schemes

Effect of Different Refs. We conduct experiments with different reference images ($\times r_w$ Ref from ultra-wide image and $\times r_t$ Ref from telephoto image). As shown in Table 10, using $\times r_t$ Ref leads to higher PSNR compared to using $\times r_w$ Ref. And it further improves fidelity and perceptual metrics when combining both Refs.

Effect of Refs Fusion Scheme. Using both reference images, we investigate various fusion schemes between aligned Ref features and aligned LR ones in our experiments. From Table 11, it can not achieve satisfactory results when directly concatenating features from Refs and LR image together for restoration. The proposed progressive fusion scheme first concatenates $\times r_w$ Ref features and the LR ones, and processes them. Then, the processed features are fused with $\times r_t$ Ref features for further modulation. It can achieve a 0.49 dB PSNR improvement over the strategy of directly concatenating. In addition, we also conduct an experiment by reversing the fusion order of the two Ref features, showing a 0.22 dB PSNR drop over the proposed scheme. The experiment illustrates that it is more suitable to fuse LR features first with lower resolution ($\times r_w$) Ref features, and then with higher resolution ($\times r_t$) Ref ones.

TABLE 10: Quantitative results comparison while using different Refs.

Ref Images	PSNR↑ / SSIM↑ / LPIPS↓
$\times r_w$ Ref	29.04 / 0.8481 / 0.201
$\times r_t$ Ref	29.30 / 0.8511 / 0.201
$\times r_w$ Ref and $\times r_t$ Ref	29.62 / 0.8582 / 0.187

TABLE 11: Quantitative results of different fusion schemes between aligned Ref features and aligned LR ones.

Fusion Schemes	PSNR↑ / SSIM↑ / LPIPS↓
Concat $\times r_w$ Ref and $\times r_t$ Ref	29.13 / 0.8517 / 0.186
$\times r_t$ Ref firstly, then $\times r_w$ Ref	29.40 / 0.8572 / 0.189
$\times r_w$ Ref firstly, then $\times r_t$ Ref	29.62 / 0.8582 / 0.187

5.5 Effect of Scaling up Models

Here we scale up our models to conduct experiments. To achieve a comparable computational cost with competitive SISR methods, we double the number of channels and triple the depth in the restoration module, named 'SelfDZSR++ (Large)' and 'SelfTZSR++ (Large)'. The quantitative results are shown in Table 12. It can be seen that scaling up models generally brings about performance improvements, especially on the LPIPS metric.

TABLE 12: Effect of scaling up models. The models are trained with their all loss terms. The PNSR, SSIM, and LPIPS metrics are calculated on the full image. The #FLOPs is measured when ×4 super-resolving LR image to 1280 × 720 resolution.

Method	# Params (M)	#FLOPs (G)	PSNR↑ / SSIM↑ / LPIPS↓	
			Nikon Camera Images	**Iphone Camera Images**
SelfDZSR++	3.1	454	29.30 / 0.8511 / 0.201	23.64 / 0.7266 / 0.244
SelfTZSR++	3.3	538	29.62 / 0.8582 / 0.187	24.00 / 0.7466 / 0.215
SelfDZSR++ (Large)	16.9	1981	29.42 / 0.8508 / 0.186	23.49 / 0.7191 / 0.232
SelfTZSR++ (Large)	17.2	2084	29.72 / 0.8594 / 0.173	24.13 / 0.7525 / 0.204

6 Conclusion

Real-world image super-resolution from dual zoomed observations (DZSR) is an emerging topic, which aims to super-resolve the ultra-wide image with the reference of telephoto image. To circumvent the problem that ground-truth is unavailable, we propose an effective self-supervised learning method. To mitigate the adverse effect of image misalignment during training, we propose a two-stage alignment method consisting of patch-based optical flow and auxiliary-LR guiding alignment. To obtain visually pleasing results, we present local overlapped sliced Wasserstein loss. Moreover, we extend DZSR to multiple zoomed observations, where we present a progressive fusion scheme for better restoration. Experiments show that our proposed method can achieve better performance against the state-of-the-art methods both quantitatively and qualitatively.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (NSFC) under Grants No.s 61872118 and U19A2073.

References

[1] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE TPAMI*, vol. 38, no. 2, pp. 295–307, 2015.

[2] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *CVPR*, 2017, pp. 4681–4690.

[3] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *CVPR Workshops*, 2017, pp. 136–144.

[4] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *ECCV*, 2018, pp. 286–301.

[5] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, "Swinir: Image restoration using swin transformer," in *ICCV*, 2021, pp. 1833–1844.

[6] Z. Zhang, Z. Wang, Z. Lin, and H. Qi, "Image super-resolution by neural texture transfer," in *CVPR*, 2019, pp. 7982–7991.

[7] Y. Xie, J. Xiao, M. Sun, C. Yao, and K. Huang, "Feature representation matters: End-to-end learning for reference-based image super-resolution," in *ECCV*, 2020, pp. 230–245.

[8] G. Shim, J. Park, and I. S. Kweon, "Robust reference-based super-resolution with similarity-aware deformable convolution," in *CVPR*, 2020, pp. 8425–8434.

[9] F. Yang, H. Yang, J. Fu, H. Lu, and B. Guo, "Learning texture transformer network for image super-resolution," in *CVPR*, 2020, pp. 5791–5800.

[10] L. Lu, W. Li, X. Tao, J. Lu, and J. Jia, "Masa-sr: Matching acceleration and spatial adaptation for reference-based image super-resolution," in *CVPR*, 2021, pp. 6368–6377.

[11] Y. Jiang, K. C. Chan, X. Wang, C. C. Loy, and Z. Liu, "Robust reference-based super-resolution via c2-matching," in *CVPR*, 2021, pp. 2103–2112.

[12] Y. Huang, X. Zhang, Y. Fu, S. Chen, Y. Zhang, Y.-F. Wang, and D. He, "Task decoupled framework for reference-based super-resolution," in *CVPR*, 2022, pp. 5931–5940.

[13] J. Cao, J. Liang, K. Zhang, Y. Li, Y. Zhang, W. Wang, and L. V. Gool, "Reference-based image super-resolution with deformable attention transformer," in *ECCV*, 2022, pp. 325–342.

[14] L. Zhang, X. Li, D. He, F. Li, Y. Wang, and Z. Zhang, "Rrsr: Reciprocal reference-based image super-resolution with progressive feature alignment and selection," in *ECCV*, 2022, pp. 648–664.

[15] T. Wang, J. Xie, W. Sun, Q. Yan, and Q. Chen, "Dual-camera super-resolution with aligned attention modules," in *ICCV*, 2021, pp. 2001–2010.

[16] Z. Zhang, H. Wang, M. Liu, R. Wang, J. Zhang, and W. Zuo, "Learning raw-to-srgb mappings with inaccurately aligned supervision," in *ICCV*, 2021, pp. 4348–4358.

[17] X. Zhang, Q. Chen, R. Ng, and V. Koltun, "Zoom to learn, learn to zoom," in *CVPR*, 2019, pp. 3762–3770.

[18] P. Wei, Z. Xie, H. Lu, Z. Zhan, Q. Ye, W. Zuo, and L. Lin, "Component divide-and-conquer for real-world image super-resolution," in *ECCV*, 2020, pp. 101–117.

[19] J. Lee, M. Lee, S. Cho, and S. Lee, "Reference-based video super-resolution using multi-camera video triplets," in *CVPR*, 2022, pp. 17 824–17 833.

[20] Z. Zhang, R. Wang, H. Zhang, Y. Chen, and W. Zuo, "Self-supervised learning for real-world super-resolution from dual zoomed observations," in *ECCV*, 2022, pp. 610–627.

[21] K. Zhang, W. Zuo, and L. Zhang, "Learning a single convolutional super-resolution network for multiple degradations," in *CVPR*, 2018, pp. 3262–3271.

[22] Z. Hui, X. Gao, Y. Yang, and X. Wang, "Lightweight image super-resolution with information multi-distillation network," in *ACM MM*, 2019, pp. 2024–2032.

[23] M. Liu, Z. Zhang, L. Hou, W. Zuo, and L. Zhang, "Deep adaptive inference networks for single image super-resolution," in *ECCV Workshops*, 2020, pp. 131–148.

[24] L. Wang, X. Dong, Y. Wang, X. Ying, Z. Lin, W. An, and Y. Guo, "Exploring sparsity in image super-resolution for efficient inference," in *CVPR*, 2021, pp. 4917–4926.

[25] X. Kong, H. Zhao, Y. Qiao, and C. Dong, "Classsr: A general framework to accelerate super-resolution networks by data characteristic," in *CVPR*, 2021, pp. 12 016–12 025.

[26] W. Xie, D. Song, C. Xu, C. Xu, H. Zhang, and Y. Wang, "Learning frequency-aware dynamic network for efficient super-resolution," in *ICCV*, 2021, pp. 4308–4317.

[27] J. Gu, H. Lu, W. Zuo, and C. Dong, "Blind super-resolution with iterative kernel correction," in *CVPR*, 2019, pp. 1604–1613.

[28] Z. Luo, Y. Huang, S. Li, L. Wang, and T. Tan, "Unfolding the alternating optimization for blind super resolution," in *NeurIPS*, 2020.

[29] S. Bell-Kligler, A. Shocher, and M. Irani, "Blind super-resolution kernel estimation using an internal-gan," in *NeurIPS*, 2019, pp. 284–293.

[30] J. Liang, K. Zhang, S. Gu, L. Van Gool, and R. Timofte, "Flow-based kernel prior with application to blind super-resolution," in *CVPR*, 2021, pp. 10 601–10 610.

[31] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," in *ICLR*, 2017.

[32] J. Liang, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, "Mutual affine network for spatially variant kernel estimation in blind image super-resolution," in *ICCV*, 2021, pp. 4096–4105.

[33] L. Wang, Y. Wang, X. Dong, Q. Xu, J. Yang, W. An, and Y. Guo, "Unsupervised degradation representation learning for blind super-resolution," in *CVPR*, 2021, pp. 10 581–10 590.

[34] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, “Momentum contrast for unsupervised visual representation learning,” in *CVPR*, 2020, pp. 9729–9738.

[35] S. A. Hussein, T. Tirer, and R. Giryes, “Correction filter for single image super-resolution: Robustifying off-the-shelf deep super-resolvers,” in *CVPR*, 2020, pp. 1428–1437.

[36] K. Zhang, J. Liang, L. Van Gool, and R. Timofte, “Designing a practical degradation model for deep blind image super-resolution,” in *ICCV*, 2021, pp. 4791–4800.

[37] X. Wang, L. Xie, C. Dong, and Y. Shan, “Real-esrgan: Training real-world blind super-resolution with pure synthetic data,” in *ICCV Workshops*, 2021, pp. 1905–1914.

[38] A. Lugmayr, M. Danelljan, and R. Timofte, “Ntire 2020 challenge on real-world image super-resolution: Methods and results,” in *CVPR Workshops*, 2020, pp. 494–495.

[39] A. Lugmayr, M. Danelljan, R. Timofte, M. Fritsche *et al.*, “Aim 2019 challenge on real-world image super-resolution: Methods and results,” in *ICCV Workshops*. IEEE, 2019, pp. 3575–3583.

[40] Y. Wei, S. Gu, Y. Li, R. Timofte, L. Jin, and H. Song, “Unsupervised real-world image super resolution via domain-distance aware training,” in *CVPR*, 2021, pp. 13 385–13 394.

[41] C. Chen, Z. Xiong, X. Tian, Z.-J. Zha, and F. Wu, “Camera lens super-resolution,” in *CVPR*, 2019, pp. 1652–1660.

[42] J. Cai, H. Zeng, H. Yong, Z. Cao, and L. Zhang, “Toward real-world single image super-resolution: A new benchmark and a new model,” in *ICCV*, 2019, pp. 3086–3095.

[43] P. Wei, H. Lu, R. Timofte, L. Lin, W. Zuo *et al.*, “Aim 2020 challenge on real image super-resolution: methods and results,” in *ECCV Workshops*, 2020, pp. 392–422.

[44] J. Cai, S. Gu, R. Timofte, and L. Zhang, “Ntire 2019 challenge on real image super-resolution: Methods and results,” in *CVPR Workshops*, 2019, pp. 0–0.

[45] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, “Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume,” in *CVPR*, 2018, pp. 8934–8943.

[46] H. Zheng, M. Ji, L. Han, Z. Xu, H. Wang, Y. Liu, and L. Fang, “Learning cross-scale correspondence and patch-based synthesis for reference-based super-resolution.” in *BMVC*, vol. 1, 2017, p. 2.

[47] Y. Zhang, Z. Zhang, S. DiVerdi, Z. Wang, J. Echevarria, and Y. Fu, “Texture hallucination for large-factor painting super-resolution,” in *ECCV*, 2020, pp. 209–225.

[48] H. Zheng, M. Ji, H. Wang, Y. Liu, and L. Fang, “Crossnet: An end-to-end reference-based super resolution network using cross-scale warping,” in *ECCV*, 2018, pp. 88–104.

[49] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, “Flownet: Learning optical flow with convolutional networks,” in *ICCV*, 2015, pp. 2758–2766.

[50] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, “Deformable convolutional networks,” in *ICCV*, 2017, pp. 764–773.

[51] X. Zhu, H. Hu, S. Lin, and J. Dai, “Deformable convnets v2: More deformable, better results,” in *CVPR*, 2019, pp. 9308–9316.

[52] K. C. Chan, X. Wang, K. Yu, C. Dong, and C. C. Loy, “Understanding deformable alignment in video super-resolution,” in *AAAI*, 2021, pp. 973–981.

[53] Z. Geng, K. Sun, B. Xiao, Z. Zhang, and J. Wang, “Bottom-up human pose estimation via disentangled keypoint regression,” in *CVPR*, 2021, pp. 14 676–14 686.

[54] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, “Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network,” in *CVPR*, 2016, pp. 1874–1883.

[55] I. Deshpande, Z. Zhang, and A. G. Schwing, “Generative modeling using the sliced wasserstein distance,” in *CVPR*, 2018, pp. 3483–3491.

[56] J. Wu, Z. Huang, D. Acharya, W. Li, J. Thoma, D. P. Paudel, and L. V. Gool, “Sliced wasserstein generative models,” in *CVPR*, 2019, pp. 3713–3722.

[57] E. Heitz, K. Vanhoey, T. Chambon, and L. Belcour, “A sliced wasserstein loss for neural texture synthesis,” in *CVPR*, 2021, pp. 9412–9420.

[58] M. Delbracio, H. Talebi, and P. Milanfar, “Projected distribution loss for image enhancement,” *arXiv preprint arXiv:2012.09289*, 2020.

[59] Y. Cao, Z. Wan, D. Ren, Z. Yan, and W. Zuo, “Incorporating semi-supervised and positive-unlabeled learning for boosting full reference image quality assessment,” in *CVPR*, 2022.

[60] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *ICLR*, 2014.

[61] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *ICLR*, 2015.

[62] A. Paszke, S. Gross, F. Massa, A. Lerer, and et al., “Pytorch: An imperative style, high-performance deep learning library,” in *NeurIPS*, 2019, pp. 8024–8035.

[63] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE TIP*, vol. 13, no. 4, pp. 600–612, 2004.

[64] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *CVPR*, 2018, pp. 586–595.

[65] A. Jolicoeur-Martineau, “The relativistic discriminator: a key element missing from standard gan,” *arXiv preprint arXiv:1807.00734*, 2018.

[66] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, “Generative adversarial nets,” in *NeurIPS*, 2014.