A Simple Plugin for Transforming Images to Arbitrary Scales

Qinye Zhou*[1]
zhouqinye@sjtu.edu.cn
Ziyi Li*[1]
599lzy@sjtu.edu.cn
Weidi Xie†[1,2]
weidi@sjtu.edu.cn
Xiaoyun Zhang†[1]
xiaoyun.zhang@sjtu.edu.cn
Yanfeng Wang[1,2]
wangyanfeng@sjtu.edu.cn
Ya Zhang[1,2]
ya_zhang@sjtu.edu.cn

[1] Coop. Medianet Innovation Center, Shanghai Jiao Tong University, China
[2] Shanghai AI Laboratory

Abstract

Existing models on super-resolution often specialized for one scale, fundamentally limiting their use in practical scenarios. In this paper, we aim to develop a general *plugin* that can be inserted into existing super-resolution models, conveniently augmenting their ability towards **A**rbitrary **R**esolution **I**mage **S**caling, thus termed **ARIS**. We make the following contributions: (i) we propose a transformer-based plugin module, which uses spatial coordinates as query, iteratively attend the low-resolution image feature through cross-attention, and output visual feature for the queried spatial location, resembling an implicit representation for images; (ii) we introduce a novel self-supervised training scheme, that exploits consistency constraints to effectively augment the model's ability for upsampling images towards unseen scales, *i.e.* ground-truth high-resolution images are not available; (iii) without loss of generality, we inject the proposed **ARIS** plugin module into several existing models, namely, IPT, SwinIR, and HAT, showing that the resulting models can not only maintain their original performance on fixed scale factor but also extrapolate to unseen scales, substantially outperforming existing any-scale super-resolution models on standard benchmarks, *e.g.* Urban100, DIV2K, etc. Project page: https://lipurple.github.io/ARIS_Webpage/

1 Introduction

Image super-resolution (SR) aims to reconstruct high-resolution (HR) images from corresponding degraded low-resolution (LR) images. In the literature, existing research [8, 9, 10, 19, 26, 38, 39, 46] predominately focuses on training specific models that work well for

*Both the authors have contributed equally to this project. † denote corresponding authors.

a few scaling factors, thus, different models have to be trained for different factors, limiting their practical use in real-world applications, when one may want to scale the image into arbitrary-resolution for viewing purpose. To address this limitation, some recent approaches, *e.g.* MetaSR [15], LIIF [6] and LTE [23] have considered designing specific architectures for arbitrary-scale super-resolution with a single model. Despite being promising, these models still fall behind the existing SR models on low-scale super-resolution, as we have experimentally shown in Table 1.

In this paper, our goal is to develop a general **plugin** module that can be inserted into any existing SR models, conveniently augmenting their ability to **A**rbitrary **R**esolution **I**mage **S**caling, thus termed **ARIS**. Specifically, we adopt a transformer-based architecture, with spatial coordinates naturally treated as the queries, that iteratively attend visual feature of the low-resolution image through an attention mechanism, and output the visual representation for desired high-resolution image, to be decoded into RGB intensity value at last. We can continuously scale the image to arbitrary resolution by simply changing the granularity of the spatial coordinates, resembling an implicit representation of the images.

In contrast to LIIF [6] and LTE [23], which also represent an image as a continuous function by MLPs that maps coordinates and the corresponding local latent codes to RGB values, thus achieving arbitrary-scale super-resolution, our proposed idea poses two critical differences: i) we represent the image continuously at the feature level, mapping the low-resolution image feature into the high-resolution image feature. Thus our module can be inserted into any network without replacing other components and the pre-trained parameters can be re-used directly, while LIIF [6] and LTE [23] need to retrain the whole network; (ii) we use spatial coordinates as query, iteratively attend the low-resolution image feature through cross-attention, and make full use of the global dependency in images, for example, self-similarity, while LIIF [6] and LTE [23] take the local latent code as input and have a limited receptive field.

Additionally, for the arbitrary-scale super-resolution task, it is often impractical to collect the paired LR-HR images for each scale with high quality, which prevents model from training towards unseen scales. To this end, we formulate this problem of reconstruction with incomplete measurements and introduce a self-supervised training scheme, that scales the image to a target resolution in the absence of paired data, by exploiting consistency constraints. Specifically, for the scales whose high-resolution images are not available, the model is trained by either upsampling the HR images of seen scales or downsampling the reconstructed images towards seen scales. As a result, we show this self-supervised training scheme can significantly improve the performance on unseen scales.

To summarise, we consider the problem of arbitrary-scale image super-resolution, and make the following contributions: (i) we propose a transformer-based plugin module, called ARIS, which resembles an implicit representation for images and can be inserted into any existing super-resolution models, conveniently augmenting their ability to upsample the image with arbitrary scale; (ii) we introduce a novel self-supervised training scheme, that exploits consistency constraints to train our ARIS plugin module towards out-of-distribution scales, *i.e.*, LR-HR image pairs are unavailable; (iii) the ARIS plugin module is orthogonal to the development of new super-resolution architectures, we insert it into several strong models published recently, namely, IPT [4], SwinIR [24], HAT [5], the resulting models outperform the any-scale super-resolution models on various benchmarks, *e.g.* Urban100, DIV2K, *etc*.

2 Related Work

Image Super-resolution. Image super-resolution (SR) is probably one of the most widely researched problems in computer vision history, reviewing all the work would be prohibitively impossible, we thus only discuss some of the most relevant work. Early super-resolution approaches are exemplar [3, 16] or dictionary [37, 42] based super-resolution. These methods generate high-resolution images by using the similarities within and between images. And the performance is limited by the size of the dataset. Since SRCNN [9], ConvNets were adopted for solving the image super-resolution task, afterwards, numerous deep learning based methods [8, 9, 10, 18, 19, 21, 25, 26, 38, 39, 45, 46] have been proposed to improve the image quality. Specifically, some works innovate the architectural design of the ConvNets, for example, the residual block [25, 45, 46], skip-connection [18, 19, 27] and recursive network [19, 34]; Other works explore different training objectives, for example, using adversarial learning [22, 38]. Recently, a series of Transformer-based approaches [4, 5, 24, 26] have shown superior performance. Generally speaking, these models are often specialised for single-scale super-resolution, which fundamentally limits their use in practical scenarios, where we may want to scale the image to arbitrary scales.

Arbitrary-scale Super-resolution. To overcome the above limitation, MDSR [25] proposed to integrate a collection of modules that are trained for different scale factors (*i.e.* x2, x3, x4). Hu *et al*. [15] proposed MetaSR to solve the arbitrary-scale upsampling problem with meta-learning, which directly predicts the filter weights for different scale factors. Building on MetaSR, ArbSR [36] designs a scale-aware convolution layer to make better use of the scale information and can handle the problem of asymmetric SR. Recently, LIIF [6] introduces the idea of implicit neural representation for images, which treats images as a function of coordinates, thus allowing to scale the image at continuous scales by simply manipulating the spatial grid of the image. LTE [23] introduces a dominant-frequency estimator to allow an implicit function to learn fine details while restoring images in arbitrary resolution. In this paper, we continue the vein of research on using implicit neural representation for image super-resolution, we adopt the transformer-based architecture, where the spatial grid can be used as the query in the transformer decoder, allowing to iteratively attend the low-resolution image feature through cross-attention.

Implicit Neural Representation. Recent work has demonstrated the great potential of using neural networks as a continuous representation of the signals, for example, for shapes [1, 12, 13], objects [1, 14, 32], or scenes [17, 30, 33]. Theoretically, such continuous parameterization enables to represent the signal to any level of fine details, with significantly less memory than using a discrete lookup table. In these representations, an object or scene is usually represented as a multilayer perceptron that maps coordinates to signed distance [1, 17, 32], occupancy [7, 29, 31] or RGB values [30, 33]. In this paper, we focus on learning implicit image representation at the feature level by a transformer-based architecture.

3 Methods

In this paper, our goal is to develop a general plugin module for any existing super-resolution (SR) model that can augment its ability to arbitrary resolution image scaling (ARIS). The baseline SR network, which refers to the pre-trained scale-specific SR network can be simplified as an encoder-decoder network, where the encoder extracts the feature map for the

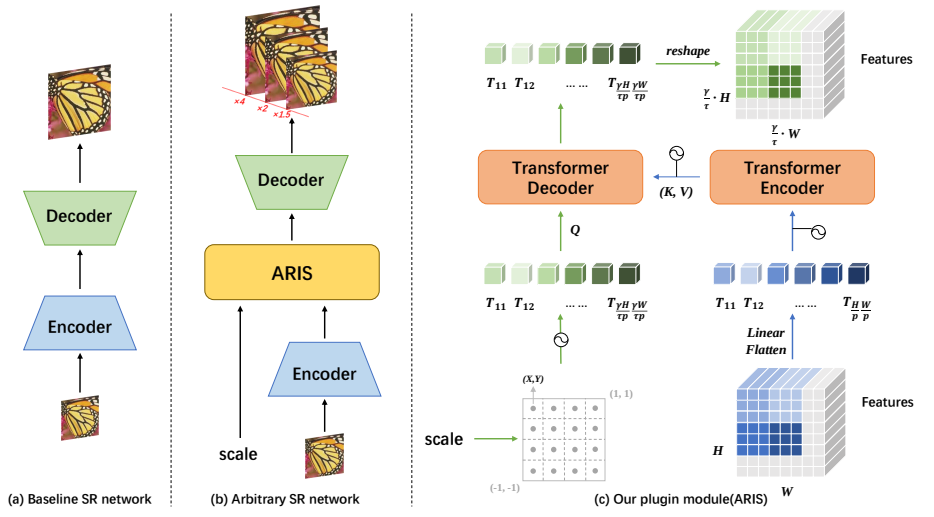

Figure 1. An overview of our ARIS plugin module. Our ARIS plugin module can be inserted into baseline SR network (a) to obtain arbitrary-scale SR network (b). We show the details of the ARIS plugin module in (c). The ARIS module utilizes the coordinate map (regarded as QUERY) and low-resolution image feature as input and outputs the desired super-resolution image feature.

low-resolution input image and the decoder outputs the super-resolution image as shown in Figure 1(a). We can obtain the arbitrary-scale SR network by inserting our ARIS plugin module into the baseline SR network as shown in Figure 1(b).

Overview. Assuming we have a training set with N paired low- and high-resolution images, $\mathcal{D}_{\text{train}} = \{(\mathcal{X}_{\text{LR}}, \mathcal{X}^2_{\text{HR}}, \mathcal{X}^3_{\text{HR}}, \mathcal{X}^4_{\text{HR}})_n, n \in [1,N]\}$, where $\mathcal{X}_{\text{LR}} \in \mathbb{R}^{H\times W\times 3}$ refers to the low-resolution image, and $\mathcal{X}^k_{\text{HR}} \in \mathbb{R}^{kH\times kW\times 3}, \forall k \in [2,3,4]$ refers to its $k\times$ upsampled high-resolution image. The goal is thus to obtain a model that can transform a low-resolution image into arbitrary scales:

$$\mathcal{Y}^{\gamma}_{\text{SR}} = \Phi(\mathcal{X}_{\text{LR}}, \gamma) = \Phi_{\text{DEC}}(\Phi_{\text{ARIS}}(\Phi_{\text{ENC}}(\mathcal{X}_{\text{LR}}), \gamma)) \tag{1}$$

where $\Phi(\cdot)$ denotes the trainable function, parameterized by the encoder (Φ_{E}), decoder (Φ_{D}) of the baseline SR network and our ARIS plugin module (Φ_{ARIS}), that maps a low-resolution image ($\mathcal{X}_{\text{LR}}$), to the desired super-resolution image ($\mathcal{Y}^{\gamma}_{\text{SR}} \in \mathbb{R}^{\gamma H\times \gamma W\times 3}$), with the scale γ denoting continuous values, *e.g.* $\gamma \in [1,8]$.

In the following sections, we first describe the details of the proposed ARIS plugin module in Section 3.1; we then introduce a novel training regime that allows training the model for arbitrary resolution scaling, even without LR-HR image pairs in Section 3.2.

3.1 ARIS Plugin Module

Unlike conventional representation that regards an image as a look-up table of intensity values, we use an implicit representation that treats an image as a function mapping from spatial coordinates to intensity, thus allows to continuously scale the image to arbitrary resolution by simply changing the granularity of its spatial coordinates. Specifically, we adopt a variant of transformer architecture for our ARIS plugin, with the normalised spatial coordinates

as `query`, iteratively attending the visual feature of the low-resolution image, to aggregate both local and global information, and eventually decode to the image of desired resolution.

As shown in Figure 1(c), the ARIS plugin module has two components, consisting of transformer encoder and transformer decoder respectively. Specifically, the transformer encoder aims to globally aggregate the local visual features from low-resolution images, while the transformer decoder resembles the implicit representation for image, mapping coordinates to visual features for decoding later.

3.1.1 Transformer Encoder

Given the visual feature from the encoder of a baseline SR network, *i.e.* $\Phi_{\text{ENC}}(\mathcal{X}_{\text{LR}})$, we use transformer encoder with L layers to aggregate information globally:

$$\mathcal{F}_{\text{LR}} = \Phi_{\text{TRANSFORMER-E}}(\Phi_{\text{ENC}}(\mathcal{X}_{\text{LR}}) + \text{PE}), \tag{2}$$

where $\Phi_{\text{TRANSFORMER-E}}(\cdot)$ refers to the transformer encoder. To maintain the spatial information, learnable position encodings (PE) are added to the visual features, and then passed into the transformer as a sequence of tokens. As a consequence, features computed from the transformer encoder is denoted as $\mathcal{F}_{\text{LR}} \in \mathbb{R}^{\frac{HW}{p^2} \times C}$, with p, C referring to the patch size used to generate tokens, and feature channels respectively.

3.1.2 Transformer Decoder for Implicit Image Representation

Here, we parametrize the image as a mapping from image coordinates to visual features, by adopting a module with multiple transformer decoder layers. In detail, we start by constructing a normalised spatial grid based on the desired scaling factor, and project them into high-dimensional vectors with the Fourier encoding [40], $\mathcal{Q}_{\text{SR}} = \Phi_{\text{FOURIER}}([\mathbf{x}, \mathbf{y}])$, where $[\cdot,\cdot]$ indicates concatenation of spatial coordinates, $\mathbf{x} = [-1, \alpha - 1, 2\alpha - 1, \ldots, 1]$, $\mathbf{y} = [-1, \beta - 1, 2\beta - 1, \ldots, 1]$ refer to the spatial coordinates respectively, with gaps computed as $\alpha = \frac{\gamma H - 1}{2p\tau}$, $\beta = \frac{\gamma W - 1}{2p\tau}$, where τ refers to the upsampling scale of baseline SR network. As a result, $\mathcal{Q}_{\text{SR}} \in \mathbb{R}^{\frac{\gamma H}{p\tau} \times \frac{\gamma W}{p\tau} \times C}$ denotes the Fourier encoded spatial coordinates for the desired super-resolution image. **Note that**, as γ can be any continuous value, the granularity of the spatial coordinates can thus be varying accordingly.

Next, we convert the spatial coordinates map into a sequence of vectors and used it as `Query` into a stack of transformer decoder layers ($\Phi_{\text{TRANSFORMER-D}}(\cdot)$),

$$\mathcal{F}_{\text{SR}} = \Phi_{\text{TRANSFORMER-D}}(W^Q \cdot \mathcal{Q}_{\text{SR}},\ W^K \cdot \mathcal{F}_{\text{LR}},\ W^V \cdot \mathcal{F}_{\text{LR}}) \tag{3}$$

where W^K, W^V refer to the learnable parameters that project the visual features to `Key` and `Value`, $\mathcal{F}_{\text{SR}}$ refers to the enriched visual feature map that can be decoded into desired super-resolution image with decoder, *i.e.*, $\mathcal{Y}^{\gamma}_{\text{SR}} = \Phi_{\text{DEC}}(\mathcal{F}_{\text{SR}})$.

Discussion. To summarise, the transformer-based ARIS plugin module can generally adapt to any existing SR models, enabling them to achieve arbitrary resolution image scaling. ARIS can globally aggregate the visual feature extracted by the baseline SR network using transformer encoder and further map the feature and spatial coordinates to a visual representation of desired super-resolution image using transformer decoder, similar to implicit representation for images.

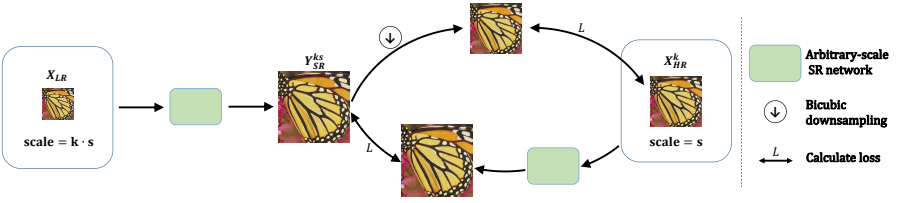

Figure 2. Self-supervised training strategy with consistency constraints. The first training setting is to downsample the SR image ($\mathcal{Y}^{ks}_{\text{SR}}$) to the same resolution as available HR image ($\mathcal{X}^{k}_{\text{HR}}$), and thus can supervise, called down-consistency training. The second training setting is to upsample the HR image ($\mathcal{X}^{k}_{\text{HR}}$) to the same resolution as the SR image ($\mathcal{Y}^{ks}_{\text{SR}}$) and then supervise using $L1$ loss, called up-consistency training.

3.2 Self-supervised Training Strategy with Consistency Constraints

It is the common practise in arbitrary scale super-resolution (*e.g.*, LIIF [8]), where the scales are divided into in-distribution and out-of-distribution. In our case, the $\times 2$, $\times 3$ and $\times 4$ are in-distribution (groundtruth LR-HR pairs are available in the given dataset $\mathcal{D}_{\text{train}}$), $\times 6$ and $\times 8$ are considered as out-of-distribution (no groundtruth LR-HR pairs). For in-distribution scales, we can train our SR model using the traditional supervision method, *i.e.*, $\mathcal{L}_{\text{pair}} = L_1(\mathcal{Y}^{k}_{\text{SR}}, \mathcal{X}^{k}_{\text{HR}})$ where $k \in \{2,3,4\}$ refers to the scale factor, $\mathcal{Y}^{k}_{\text{SR}} = \Phi(\mathcal{X}_{\text{LR}}, k)$ refers to the generated super-resolution image. For the out-of-distribution scales, in order to train the model beyond the resolution limitation, *i.e.* the resolution of dataset images might be infeasible for generating LR images for large scales, for example, if an HR image is only of resolution 128×128, the size of the generated LR image for $\times 8$ is 16×16 at maximum, thus it will be infeasible to train the model for $\times 8$ scaling 32×32 image, we adopt a self-supervised training scheme that exploits consistency constraints.

As shown in Figure 2, our proposed training scheme includes two settings, *i.e.*, down-consistency training and up-consistency training. Specifically, we first use our arbitrary-scale SR network to scale the low-resolution image by $k \cdot s$ times, *i.e.*, $\mathcal{Y}^{ks}_{\text{SR}} = \Phi(\mathcal{X}_{\text{LR}}, k \cdot s)$. For down-consistency training, we downsample the generated super-resolution image to the same resolution as available high-resolution image ($\mathcal{X}^{k}_{\text{HR}}$), and use $L1$ loss as the objective for optimisation. The down-consistency training method can be formulated as:

$$\mathcal{L}_{\text{down-consistency}} = |\Phi_{\text{BICUBIC}}(\mathcal{Y}^{ks}_{\text{SR}}, s) - \mathcal{X}^{k}_{\text{HR}}|_1 \tag{4}$$

where $\Phi_{\text{BICUBIC}}(\cdot, s)$ refers to the simple bicubic downsampling, with a factor of s.

For up-consistency training, we use our arbitrary-scale SR network to upsample high-resolution image ($\mathcal{X}^{k}_{\text{HR}}$) to the same resolution as the generated super-resolution image ($\mathcal{Y}^{ks}_{\text{SR}}$) so that we can supervise. It can be formulated as:

$$\mathcal{L}_{\text{up-consistency}} = |\mathcal{Y}^{ks}_{\text{SR}} - \Phi(\mathcal{X}^{k}_{\text{HR}}, s)|_1 \tag{5}$$

Note that, we use $\mathcal{L}_{\text{pair}}$, $\mathcal{L}_{\text{down-consistency}}$ and $\mathcal{L}_{\text{up-consistency}}$ together to train the arbitrary-scale SR network for both seen and unseen scales.

4 Experiments

4.1 Datasets and Metrics

We train all models on DIV2K, and then evaluate them on five standard benchmark datasets:

Training Set. DIV2K [35] contains over 1000 images in 2K resolution, with 800 images for training, and 100 images for validation and testing, respectively. All of our models are trained with DIV2K training set.

Testing Set. Following previous work, we report the performance of our model on 4 benchmark datasets, namely, Set5 [2], Set14 [44], B100 [28], Urban100 [16] and the DIV2K validation set. Note that all the degradation images are generated by the Matlab function *imresize* with the default setting of bicubic interpolation.

Evaluation Metrics. In accordance with [4, 6, 15, 23, 24, 41], we report peak signal-to-noise ratio (PSNR) on the Y channel of the transformed YCbCr color space for the 4 benchmark datasets and the PSNR on the RGB channel for the DIV2K validation set.

4.2 Training Details

Baseline SR Network. In our experiments, we choose IPT [4], SwinIR[24] and HAT[5] as our baseline SR network, and only use their pre-trained model on scale ×2. As shown in Fig. 1, we inject the plugin module between their feature extractor and upsampling layers, the resulting models are termed IPT-ARIS, SwinIR-ARIS, and HAT-ARIS respectively. Each resulting model is trained individually. Following the design in the original baseline networks, we randomly crop 48×48 patches to form the LR images and feed them into the feature extractor and train the model to reconstruct their corresponding HR patches for all models. Additionally, we perform data augmentation by randomly rotating 90°, 180°, 270°, and horizontal flipping.

Implementation Details. The setting of training scale factors follows the training strategy described in Section 3.2. Specifically, we use ×2, ×3, ×4, ×6, and ×8 as our training scale factors. For ×2, ×3, and ×4, we have LR-HR pairs, while for ×6 and ×8, we adopt self-supervised training scheme with consistency constraints. For the Transformer Encoder and Transformer Decoder, we both use multi-head attention with 6 heads and 6 layers. For training, we use four NVIDIA Tesla V100 GPUs to train our model for 300 epochs with batch size 8, ADAM [20] optimiser with $\beta_1 = 0.9$, $\beta_2 = 0.999$. The initial learning rate is set to 5e-5 and decayed by one-half at epoch 150, 200, and 250.

5 Results

As shown in Table. 1, we provide experimental results for our proposed plugin injected into different baseline networks, then compare with other state-of-the-art approaches for arbitrary-scale super-resolution. After that, we conduct a series of ablation studies on different architectural design choices and training strategies.

5.1 Quantitative Results

Comparison to State-of-the-art. When comparing to the existing approaches for arbitrary-scale super-resolution, due to the architectural difference, we only include the best numbers.

			arbitrary-scale SR method							single-scale SR method		
Datasets	scale	Bicubic [25]	MetaSR [15]	ArbSR [36]	LIIF [6]	LTE [23]	IPT-ARIS	SwinIR-ARIS	HAT-ARIS	IPT [4]	SwinIR [24]	HAT [5]
Set5	×2	33.97	38.22	38.26	38.17	38.33	38.20	38.25	**38.50**	38.37	38.35	38.63
	×3	30.63	34.63	34.75	34.68	34.89	34.69	34.82	**35.00**	34.81	34.89	35.06
	×4	28.63	32.38	32.50	32.50	32.81	32.58	32.66	**32.94**	32.64	32.72	33.04
	×6	26.09	29.04	28.45	29.15	29.50	29.09	29.38	**29.63**	-	-	-
	×8	24.52	26.96	26.21	27.14	27.35	27.18	27.31	**27.50**	-	-	-
Set14	×2	30.55	33.98	34.07	33.97	34.25	33.94	34.21	**34.81**	34.43	34.14	34.86
	×3	27.79	30.54	30.64	30.53	30.80	30.64	30.75	**31.05**	30.85	30.77	31.08
	×4	26.21	28.78	28.84	28.80	29.06	28.92	29.01	**29.22**	29.01	28.94	29.23
	×6	24.44	26.51	26.22	26.64	26.86	26.61	26.79	**26.96**	-	-	-
	×8	23.28	24.97	24.55	25.15	25.42	25.11	25.27	**25.47**	-	-	-
B100	×2	29.73	32.33	32.38	32.32	32.44	32.35	32.42	**32.59**	32.48	32.44	32.62
	×3	27.31	29.26	29.31	29.26	29.39	29.30	29.36	**29.49**	29.38	29.37	29.54
	×4	26.04	27.71	27.74	27.74	27.86	27.78	27.84	**27.98**	27.82	27.83	28.00
	×6	24.61	25.90	25.74	25.98	26.09	25.97	26.04	**26.16**	-	-	-
	×8	23.73	24.83	24.55	24.91	25.03	24.92	24.98	**25.09**	-	-	-
Urban100	×2	27.07	32.92	33.07	32.87	33.50	33.17	33.35	**34.28**	33.76	33.40	34.45
	×3	24.58	28.82	28.97	28.82	29.41	29.21	29.31	**30.01**	29.49	29.29	30.23
	×4	23.24	26.55	26.63	26.68	27.24	27.13	27.21	**27.84**	27.26	27.07	27.97
	×6	21.71	23.99	23.70	24.20	24.62	24.43	24.50	**25.00**	-	-	-
	×8	20.80	22.59	22.13	22.79	23.17	23.16	23.08	**23.54**	-	-	-
DIV2K	×2	31.24	35.00	34.97	34.99	35.24	34.99	35.13	**35.41**	-	-	-
	×3	28.37	31.27	31.28	31.26	31.50	31.32	31.44	**31.67**	-	-	-
	×4	26.78	29.25	29.23	29.27	29.51	29.37	29.47	**29.70**	-	-	-
	×6	24.93	26.88	26.61	26.99	27.20	27.03	27.10	**27.31**	-	-	-
	×8	23.78	25.57	24.99	25.61	25.81	25.64	25.71	**25.91**	-	-	-

Table 1. Quantitative comparison with single-scale SR methods and other state-of-the-art methods for arbitrary-scale super-resolution on five benchmark datasets. The bold numbers indicate the best results in arbitrary-scale SR methods. Arbitrary-scale SR methods train a single model for all scales. Single-scale SR methods train a specific model for each scale.

As shown in Table 1, adding our proposed ARIS plugin to any baseline SR network can achieve competitive performance on all benchmarks, and specifically, HAT-ARIS can outperform all existing arbitrary-scale super-resolution models, for both in- and out-distribution scales, validating the effectiveness of ARIS.

Compare to Baseline SR Network. We compare the SR results on baseline networks before and after injecting the ARIS plugin. The IPT-ARIS, SwinIR-ARIS, and HAT-ARIS achieve comparable performance to their corresponding baseline networks on in-distribution (seen) scale factors ($\times 2$, $\times 3$ and $\times 4$) and are able to extrapolate to out-distribution scales ($\times 6$, $\times 8$). Notably, the baseline SR networks are trained for a specific scale, thus they may have more advantages on a specific task than our method. Our plugin module inherits the original ability of baseline SR networks and fits more scales effectively.

5.2 Ablation Studies

In this section, we perform ablation studies on the necessity of self-supervised training scheme with consistency constraints and the number of transformer layers and heads, due to the space limitation, we leave the investigation to the patch size and the training scale in the supplementary material.

Figure 3. Qualitative comparison of the same patch under different upsampling scale factors. Our method recovers clearer edges of printed texts.

Self-supervised Training Scheme with Consistency Constraints. In Table. 2, we use the standard DIV2K training dataset as our training set, which contains paired images in different scaling factors (×2, ×3 and ×4). Without self-supervised training, we can achieve good performance on training scales, however, the model performs poorly on the unseen scale factors, *e.g.*, ARIS-A. While with our proposed self-supervised training, the models can effectively improve the performance on unseen scales, *e.g.*, ARIS-(B, C, D), showing it enables to train SR models even with incomplete measurements to some extent.

	training scale							testing scale				
	×2	×3	×4	×6 ↓	×8↓	×6↑	×8↑	×2	×3	×4	×6	×8
ARIS-A	✓	✓	✓					35.48	31.72	29.72	25.57	23.84
ARIS-B	✓	✓	✓	✓	✓			35.45	31.69	29.70	27.02	25.59
ARIS-C	✓	✓	✓			✓	✓	35.38	31.66	29.69	**27.31**	25.89
ARIS-D	✓	✓	✓	✓	✓	✓	✓	35.41	31.67	29.70	**27.31**	**25.91**

Table 2. Ablation study on training strategy. Performance is measured by PSNR on DIV2K validation set. Specifically, ↓ and ↑ denote the down-consistency training and up-consistency training.

Transformer Layers. We ablate the number of heads and layers of our Transformer Encoder and Decoder on the Set5 and Set14 dataset with ×2, ×3 and ×4 scales based on HAT-ARIS in Table 3. As more layers and heads are added, the performance consistently improves, achieving the best performance with 6 layers and 6 heads.

layer	head	parameters	Set5			Set14		
			×2	×3	×4	×2	×3	×4
3	3	200M	38.51	34.81	32.07	34.71	30.64	28.84
	6	200M	38.55	34.94	32.87	34.84	31.05	29.19
6	3	307M	38.50	34.76	31.99	34.62	30.53	28.82
	6	307M	**38.57**	**35.04**	**32.98**	**34.87**	**31.10**	**29.23**

Table 3. Ablation study on Transformer. Performance increases with layers and heads.

5.3 Qualitative Results

We demonstrate the qualitative results in Figure 3 and Figure 4, making the following observations: *First*, compared with other state-of-the-art methods, the image generated by our

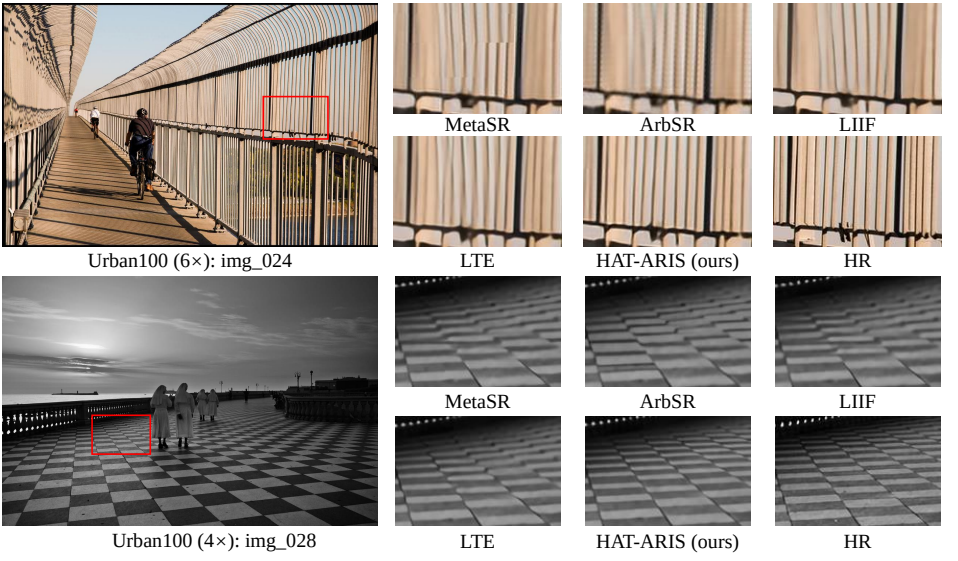

Figure 4. Visual results with bicubic downsampling from Urban100. The patches for comparison are marked with red boxes in the original images. Our method recovers more details and achieves a better visual effect.

model recovers more details and has higher fidelity, while previous methods can not recover the original images and generate some irregular shapes; *Second*, our model can learn a better continuous image representation. For example, in Figure 4, LIIF and LTE can not restore clear edges of the texts. In contrast, our HAT-ARIS is capable of maintaining the shape information better. This becomes more evident with the increase of scale factors.

6 Limitations

As ARIS adopts a Transformer architecture, it incurs relatively high memory consumption, and poses poses limitations for extending the model to finer granularity, *i.e.*, decimal scale factor with a small stride. As future work, we will investigate more efficient transformer architectures, for example, using local attention to replace the full attention, consider to sample effective tokens while computing attentions [11, 43].

7 Conclusion

In this paper, we propose ARIS, a transformer-based plugin module that can be injected into any super-resolution models and augment them towards arbitrary super-resolution. Specifically, we represent the image continuously by using spatial coordinates as query and mapping the low-resolution features into high-resolution features. We introduce a self-supervised training scheme with consistency constraints that can effectively augment the model's ability on unseen scales. Extensive experiments show that the proposed plugin module outperforms existing state-of-the-art arbitrary-SR methods on five benchmark datasets for all scale factors, showing the great potential of learning image implicit representation.

Acknowledgments

This work is supported by the National Key R&D Program of China (No. 2019YFB1804304), National Natural Science Foundation of China (62271308), 111 plan (No. BP0719010), and STCSM (No. 18DZ2270700, No. 22511105700), and State Key Laboratory of UHD Video and Audio Production and Presentation.

References

[1] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *Proc. CVPR*, 2020.

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

[3] Hong Chang, Dit-Yan Yeung, and Yimin Xiong. Super-resolution through neighbor embedding. In *Proc. CVPR*, 2004.

[4] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *Proc. CVPR*, 2021.

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

[6] Yinbo Chen, Sifei Liu, and Xiaolong Wang. Learning continuous image representation with local implicit image function. In *Proc. CVPR*, 2021.

[7] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proc. CVPR*, 2019.

[8] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *Proc. CVPR*, 2019.

[9] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence(TPAMI)*, 2015.

[10] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *Proc. ECCV*, 2016.

[11] Mohsen Fayyaz, Soroush Abbasi Kouhpayegani, Farnoush Rezaei Jafari, Eric Sommerlade, Hamid Reza Vaezi Joze, Hamed Pirsiavash, and Juergen Gall. ATS: Adaptive token sampling for efficient vision transformers. *arXiv preprint arXiv:2111.15667*, 2021.

[12] Kyle Genova, Forrester Cole, Avneesh Sud, Aaron Sarna, and Thomas A Funkhouser. Deep structured implicit functions. *arXiv preprint arXiv:1912.06126*, 2019.

[13] Kyle Genova, Forrester Cole, Daniel Vlasic, Aaron Sarna, William T Freeman, and Thomas Funkhouser. Learning shape templates with structured implicit functions. In *Proc. ICCV*, 2019.

[14] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099*, 2020.

[15] Xuecai Hu, Haoyuan Mu, Xiangyu Zhang, Zilei Wang, Tieniu Tan, and Jian Sun. Meta-sr: A magnification-arbitrary network for super-resolution. In *Proc. CVPR*, 2019.

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

[17] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, Thomas Funkhouser, et al. Local implicit grid representations for 3d scenes. In *Proc. CVPR*, 2020.

[18] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proc. CVPR*, 2016.

[19] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proc. CVPR*, 2016.

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

[21] 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*, 2017.

[22] 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*, 2017.

[23] Jaewon Lee and Kyong Hwan Jin. Local texture estimator for implicit representation function. In *Proc. CVPR*, 2022.

[24] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proc. ICCVW*, 2021.

[25] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proc. CVPRW*, 2017.

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

[27] Xiaojiao Mao, Chunhua Shen, and Yu-Bin Yang. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. *NIPS*, 2016.

[28] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. ICCV*, 2001.

[29] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proc. CVPR*, 2019.

[30] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Proc. ECCV*, 2020.

[31] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proc. CVPR*, 2020.

[32] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proc. CVPR*, 2019.

[33] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *NIPS*, 2019.

[34] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *Proc. CVPR*, 2017.

[35] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In *Proc. CVPR*, 2017.

[36] Longguang Wang, Yingqian Wang, Zaiping Lin, Jungang Yang, Wei An, and Yulan Guo. Learning a single network for scale-arbitrary super-resolution. In *Proc. ICCV*, 2021.

[37] Shenlong Wang, Lei Zhang, Yan Liang, and Quan Pan. Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis. In *Proc. CVPR*, 2012.

[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. ECCV*, 2018.

[39] Zhihao Wang, Jian Chen, and Steven CH Hoi. Deep learning for image super-resolution: A survey. *IEEE transactions on pattern analysis and machine intelligence(TPAMI)*, 2020.

[40] Yuanbo Xiangli, Linning Xu, Xingang Pan, Nanxuan Zhao, Anyi Rao, Christian Theobalt, Bo Dai, and Dahua Lin. Citynerf: Building nerf at city scale. *arXiv preprint arXiv:2112.05504*, 2021.

[41] Xingqian Xu, Zhangyang Wang, and Humphrey Shi. Ultrasr: Spatial encoding is a missing key for implicit image function-based arbitrary-scale super-resolution. *arXiv preprint arXiv:2103.12716*, 2021.

[42] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE transactions on image processing (TIP)*, 2010.

[43] Hongxu Yin, Arash Vahdat, Jose M Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-ViT: Adaptive tokens for efficient vision transformer. In *CVPR*, 2022.

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

[45] 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*, 2018.

[46] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proc. CVPR*, 2018.