Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

A feature reuse framework with texture-adaptive aggregation for reference-based super-resolution

Xiaoyong Mei [a], Yi Yang [a], Ming Li [b,a,*], Changqin Huang [c,a], Kai Zhang [d], Fudan Zheng [e]

[a] *Zhejiang Key Laboratory of Intelligent Education Technology and Application, Zhejiang Normal University, Jinhua, China*
[b] *Zhejiang Institute of Optoelectronics, Jinhua, China*
[c] *College of Education, Zhejiang University, Hangzhou, China*
[d] *School of Intelligence Science and Technology, Nanjing University, Suzhou, China*
[e] *Sun Yat-Sen University, Guangzhou Higher Education Mega Center, Guangzhou, China*

ARTICLE INFO

Keywords:
Reference-based image super-resolution
Texture adaptive
Feature reuse
Dynamic residual block

ABSTRACT

Reference-based super-resolution (RefSR), significant success has been achieved in the field of super-resolution. It reconstructs low-resolution (LR) inputs using high-resolution reference images, obtaining more high-frequency details and alleviating the ill-posed problem of single-image super-resolution (SISR). Previous research in the RefSR has mainly focused on finding correlations, transferring, and aggregating similar texture information from LR reference (Ref) the LR. However, an essential detail of perceptual loss and adversarial loss has been underestimated, impacting texture transfer and reconstruction negatively. In this paper, we propose a feature reuse framework, FRFSR, which divides the model training into two steps. Firstly, the first model is trained using reconstruction loss to enhance its texture transfer and aggregation abilities. Secondly, using all losses for training, the feature output of the first model is reintroduced into the training process to supplement texture, generating visually appealing images. The feature reuse framework is applicable to any RefSR model, and experiments show that several RefSR methods exhibit improved performance when retrained with our reuse framework. Considering that the textures in the reference are not entirely consistent with those in the LR, this naturally leads to the problem of texture misuse. Therefore, we design a Dynamic Residual Block (DRB). The DRB utilizes the feature perception capability of decoupled dynamic filters to dynamically aggregate texture information between LR input and Ref images, reducing instances of texture misuse. The source code can be obtained from https://github.com/Yi-Yang355/FRFSR.

1. Introduction

Single Image Super-Resolution (SISR) involves generating a high-resolution image with high-frequency information from a low-resolution (LR) input. The practical significance of SISR in various contexts such as medical imaging and surveillance is notable. Based on the optimization criteria, the approaches of SISR can be divided into two categories. One approach optimizes pixel-level errors such as mean squared error (MSE) and mean absolute error (MAE), potentially resulting in images that are too smooth, and the other approach involves visual perception-based errors such as perceptual loss and adversarial loss. The latter results in images with better visual effects and greater alignment to human visual perception but may produce artifacts and unrealistic textures. These approaches face the inherent problem of SISR the ill-posed nature of the problem because different high-resolution images can be degraded to the same low-resolution image [1,2]. Reference-based super-resolution (RefSR) alleviates the inherent problem of SISR to a certain extent by using an additional high-resolution reference (Ref) image to transfer relevant textures and achieve super-resolution. Methods of obtaining relevant Ref images are varied and include web search and video frames. RefSR has two primary limitations that compromise its performance. The first one is accurately finding the correspondence between the LR and Ref. Some existing methods address this through spatial alignment, such as CrossNet [3], which utilizes optical flow estimation to align LR and Ref, and SSEN [4], which employs deformable convolutions to learn adaptive LR and Ref alignment. Other methods, such as SRNTT [5], TTSR [6] adopt dense patch matching algorithms for patch matching to find corresponding matches, whereas MASA [7] employs a coarse-to-fine matching approach for reducing computational requirements. However, obtaining accurate matching is challenging due to differences

* Corresponding author at: Zhejiang Institute of Optoelectronics, Jinhua, China.
E-mail addresses: cdmxy@zjnu.edu.cn (X. Mei), yangyi@zjnu.edu.cn (Y. Yang), mingli@zjnu.edu.cn (M. Li), cqhuang@zju.edu.cn (C. Huang), cskaizhang@gmail.com (K. Zhang), zhengfd3@mail2.sysu.edu.cn (F. Zheng).

https://doi.org/10.1016/j.knosys.2025.113201
Received 5 March 2024; Received in revised form 20 May 2024; Accepted 18 February 2025
Available online 1 March 2025

in resolution and texture distribution. C^2-Matching [8] uses knowledge distillation and contrastive learning to train a feature extractor, and a combination of patch matching and deformable convolution to improve the accuracy of correspondence matching. The second challenge is effectively transferring texture features. TTSR proposes a cross-scale feature integration module that conveys texture information using multiple texture transformers in a stacked manner, whereas MASA uses a spatial adaptive module to remap the aligned Ref feature distribution, ensuring robustness to different color and brightness distributions. Additionally, DATSR [9] replaces the traditional ResBlock with the Swin-Transformer [10], resulting in considerable improvements in model performance.

Although deformable convolution [11,12] is capable of learning implicit alignment between feature maps LR and Ref, it still faces challenges in aligning distant features. Furthermore, existing RefSR methods effectively prioritize aggregating textures over reconstructing their own textures. It is also important to note that during the feature aggregation process, the ResBlock treats all pixel features equally, resulting in the introduction of irrelevant textures from the Ref image. Even with DATSR replacing ResBlock with Swin-Transformer, the window self-attention calculation will noticeably increase the parameters and runtime.

To address these three issues, we first do not make any modifications to the deformable convolution, but instead shuffle the reference image, thereby indirectly increasing the distance between similar features, increasing the training difficulty and improving performance; secondly, inspired by TADE [13], we use single-image feature embedding to assist the LR inputs to self-reconstruct their features while mitigating the introduction of irrelevant textures. Finally, we introduce a new feature aggregation module, namely Dynamic ResBlock (DRB). Specifically, the DRB module adds a group of decoupled filters to the residual block, which can aware texture information in both the spatial and channel domains, and then adaptively aggregate relevant textures, further reducing the introduction of irrelevant information such as noise, wrong textures, etc. In addition, we employ residual blocks with an Enhanced Spatial Attention (ESA) after the decoupled filters to enhance the relevant texture information.

In addition to the aforementioned points, most previous works overlook a crucial fact: the increase in perceptual loss and adversarial loss adversely affects the texture transfer and reconstruction effects. To fully utilize the texture transfer and reconstruction abilities of the reconstruction loss-trained model, we propose a feature reuse framework FRFSR. In the training and testing process of $RefSR_2$ with three types losses $\mathcal{L}_{rec}+\mathcal{L}_{per}+\mathcal{L}_{adv}$, we will provide feature feedback to the feature aggregation process from the $RefSR_1$ trained with only one loss $\mathcal{L}_{rec}$. This maneuver effectively diminishes the impact of perceptual and adversarial losses on texture transfer and reconstruction. In summary, this paper's primary contributions are:

1. We introduce a feature reuse framework that effectively mitigates the degradation of texture reconstruction from the application of perceptual loss and adversarial loss. We apply this framework to various RefSR methods, which have shown consistent improvements in performance.
2. To enhance the reconstruction of LR's self-texture and maintain texture relevance, we utilize a single-image feature embedding module. Unlike the approach used by [13], we exclude feature upsampling and final image reconstruction processes in this module, and focus solely on embedding the LR's own reconstructed features into the aggregation process.
3. We designed a dynamic residual block and introduced it into the texture adaptive module. This block applies decoupled dynamic filters and enhanced spatial attention to selectively perceive and transfer textures from the Ref image. This approach adaptively reduces the likelihood of introducing incorrect textures.
4. Our method achieved state-of-the-art (SOTA) performance in multiple benchmarks, demonstrating significant improvements in robustness to unrelated reference images and long-range feature alignment. Notably, even without the single image feature embedding module, our method still achieved SOTA performance in CUFED5.

2. Related work

2.1. Single image super-resolution

Single image super-resolution (SISR) aims to input a single LR image and reconstruct it to an image with high-frequency details. Before the emergence of deep learning, traditional methods such as various interpolation methods were usually used. With SRCNN [14] first using deep learning methods to perform super-resolution, deep learning-based super-resolution began to appear in large numbers. Later, ResNet [15] appeared, which deepened the network layers. EDSR [16], CARN [17] and other methods added residual structure in super-resolution models, thus improving the performance of super-resolution. After this, the attention mechanism merged, which can make the network selectively focus on some features and appropriately ignore unnecessary features. RCAN [18] was the first to apply the attention mechanism to super-resolution. Additionally, the game theory approach used by GAN [19] has enabled GAN-based super-resolution models, such as SRGAN [20], ESRGAN [21], RankSRGAN [22], AMPRN [23], and Real-ESRGAN [24] to deliver enhanced perceptual quality in produced images. Recently, SRGAT [25] used the graph attention network to help LR recover additional textures from neighboring patches. TDPN [26] utilizes a texture and detail-preserving network that preserves texture and detail while the features are reconstructed. However, the SISR problem is ill-posed, with low-resolution (LR) and super-resolution (SR) having a one-to-many relationship.

2.2. Reference-based image super-resolution

The biggest difference between RefSR and SISR is that the former has an additional high-resolution Ref image. The RefSR can transfer texture details from the Ref image to LR to help LR reconstruction, and these texture details should be similar to the ground truth (GT). CrossNet [3] twists the reference image and LR to align them through the flow estimation network. SSEN [4] uses deformable convolution [11, 12] to align LR and Ref images. RRSGAN [27] utilizes deformable convolutions to align the Ref and LR features. It also employs a Correlation Attention Module (RAM) to enhance the model's robustness in different scenarios. Both of these methods are implicit alignment, and some work performs feature matching between LR and Ref to achieve explicit alignment. SRNTT [5] enumerates patches to transfer multi-scale reference features. CIMR-SR [28] employs a content independent searching the local matched patterns. E2ENT [29] constructs a match and swap module to obtain similar texture and high-frequency information. TTSR [6] introduces the Transformer architecture to more reasonably transfer reference features by combining soft and hard attention. MASA [7] uses a matching method from coarse to fine to reduce the computational complexity and a spatial adaptive module is used to make the transferred texture closer to GT. However, due to the resolution gap between the LR and Ref image, the matching performance is affected. C^2-Matching [8] introduces knowledge distillation and contrastive learning methods, which greatly improve the matching robustness between LR and Ref. WTRN [30] utilizes the benefits of wavelet transformation to categorize features into high-frequency and low-frequency sub-bands, which facilitates the transfer of texture patterns with more effectiveness. TADE [13] uses a decoupling framework, which divides RefSR into two parts: super-resolution and texture migration, which alleviates the two problems of reference-underuse and reference-misuse. However, it does not take into consideration the lack of detailed textures in the super-resolution image, which results in inaccurate matching between SR and Ref. DATSR [9] uses the Swin-Transformer [10] to replace the traditional ResBlock for feature aggregation. ERVSR [31] introduces an attention-based feature align module and an aggregation upsampling module for video super-resolution that attends LR features using the correlation between the reference and LR frames. Recently, RRSR [32] implemented a reciprocal learning strategy, thereby strengthening the learning of the model. [33,34] enhance the detail quality of input images by transferring texture details from multiple reference images. Reviewing the existing research findings, it can be seen that first, the existing methods do not fully take into consideration the textural dissimilarities between LR and Ref, so it is still inevitable that irrelevant textures are introduced in the texture transfer process. Second, existing studies have focused on improving the accuracy of matching and the ability of texture transfer, but few studies have focused on the texture detail reconstruction of LR itself. Third, no one has noticed that adding perceptual loss and adversarial loss will lead to a decline in the texture reconstruction effect. To address the aforementioned issues, we propose a dynamic residual block (DRB) to perceive texture information, adaptively transfer and aggregate relevant textures and suppress irrelevant textures and reconstruct their own features by embedding single-image feature reconstruction LR features. In addition, we propose a feature reuse framework to improve the texture reconstruction effect under perceptual loss and adversarial loss supervision.

2.3. Dynamic weights

Unlike the weight sharing in conventional convolutions, dynamic filters [35–39] have content-aware characteristics and are capable of dynamically adjusting and predicting filter weights based on input features. The dynamic weights approach has been successfully applied in various works, such as super-resolution [40–42], image deblurring [43], image denoising [44], adaptive modulation [45,46] and style transfer [47], because of its powerful representation and content-awareness capabilities. The work in [32], which introduces a set of reference-aware filters for selecting reference features to identify the most suitable texture, is strongly related to our study. However, the generation of these filters is computationally expensive due to their deep separable and spatially changing nature, leading to high time consumption. Inspired by [38], we propose to decouple the spatial and channel domains and use spatial and channel attention to dynamically filter each pixel, extending this to texture-adaptive aggregation.

3. Methodology

3.1. Feature reuse reconstruction framework

Firstly, we discovered that RefSR struggles to reconstruct high-frequency details from the LR image (I_{LR}) itself. To address this issue, we utilize an SISR method without upsampling called SIFE to reconstruct fine texture features F_{sife} from I_{LR}. These reconstructed features are then integrated into the reconstruction process of the reference-based super-resolution. This approach not only supplements the difficult to reconstruct texture details in RefSR but also helps to limit the introduction of irrelevant textures to some extent.

$$F_{sife} = SIFE(I_{LR}). \tag{1}$$

We chose the same SISR baseline used in [13] to ensure a more equitable comparison. Nevertheless, we removed the last upsampling stage which is present in SISR.

Previous work has shown that feature reuse [15,48–51] prevents the vanishing gradient issue in deep networks to enhance network learning and parameter efficiency by inputting previous layers' features into subsequent layers. Various computer vision tasks, such as super-resolution [52], image compression [53], and image restoration [54], utilize the characteristic of feature reuse to enhance the efficiency and effectiveness of their models. Prior studies have shown that SR images which are produced using only reconstruction loss are much more detailed in texture compared to those generated by models that use perceptual and adversarial losses. To address this issue, we propose to utilize a pre-trained model $RefSR_1$ that generates SR feature maps with fine textures through reconstruction loss only, and integrate them into the second model trained with three types losses to supplement texture reconstruction and accelerate convergence of the second model $RefSR_2$, as shown in Fig. 1. Therefore, we extend feature reuse to the training process of the second models. In summary, first, we input the LR image I_{LR}, Ref image I_{Ref} and the features F_{sife} extracted by a pre-trained SIFE network into the network to obtain the reconstructed high-resolution features F_{SR}^{rec}, which is then convolved to generate the output image I_{SR}^{rec}. In this process, we only train the RefSR model with reconstruction loss, consistent with previous $RefSR_1$ methods.

$$F_{SR}^{rec} = RefSR_1\left(I_{LR}, I_{Ref}, F_{sife}\right), \tag{2}$$

$$I_{SR}^{rec} = Conv(F_{SR}^{rec}). \tag{3}$$

At this stage, we have obtained a super-resolution network that exhibits impressive texture transfer and reconstruction capabilities. However, to produce high-quality perceptual images, supervision using perceptual and adversarial losses is typically required. Finally, to further enhance $RefSR_2$'s texture transfer and reconstruction capabilities, we generate F_{SR}^{rec} with refined texture details using $RefSR_1$, and then incorporate this feature map back into the training of the $RefSR_2$. Note that in this process, the $RefSR_1$ and SIFE are only responsible for inference and does not participate in weight updating. The aforementioned process can be represented as follows:

$$I_{SR}^{all} = Conv\big(RefSR_2(I_{LR}, I_{Ref}, F_{SR}^{rec}, F_{sife})\big). \tag{4}$$

Utilizing this framework, we have built $RefSR_1$ with efficient texture transfer and reconstruction performance, $RefSR_2$ with efficient perceptual reconstruction performance. Through feature reuse, the FRFSR model was collaboratively constructed, achieving a significant improvement in both qualitative and quantitative experiments with reference super-resolution. In the ablation study, we apply this framework to MASA [7] and C^2-Matching [8], demonstrating a significant improvement in their performance.

3.2. Correlation-based texture warp

For the RefSR task, a large part of the work is focused on accurately finding the matching correspondence between the LR image and the Ref image. This is crucial for subsequent texture transfer. We use the correlation based texture warp, also known as the CTW block, for matching correspondence, as shown in Fig. 2. Then, we use a parameter-sharing texture encoder to extract the texture features of LR and Ref images and generate $F_{LR\uparrow}^{tex} \in \mathbb{R}^{C\times H_{LR\uparrow}\times W_{LR\uparrow}}$, $F_{Ref}^{tex} \in \mathbb{R}^{C\times H_{Ref}\times W_{Ref}}$. We keep the texture encoder consistent with [8] because its training method of knowledge distillation and contrastive learning alleviates the problem of inaccurate matching between LR and the reference image due to different resolutions, and enhances the robustness of matching. Then, the texture features $F_{LR\uparrow}^{tex}$ and F_{Ref}^{tex} are respectively unfolded into $l\left(H_{Ref} \times W_{Ref}\right)$ patches to obtain $\left\{Q_1, Q_2, Q_3, \ldots, Q_l\right\}$, $\left\{K_1, K_2, K_3, \ldots, K_l\right\}$. The cosine similarity between Q_m and each patch K_n is calculated using the inner product formula to form the similarity matrix $\mathcal{M}_{m,n} \in \mathbb{R}^l$.

$$\hat{F}_{Ref}^{tex}, \hat{F}_{LR\uparrow}^{tex} = unfold\left(F_{Ref}^{tex}, F_{LR\uparrow}^{tex}\right), \tag{5}$$

$$\mathcal{M}_{m,n} = S_p\left({\hat{F}_{LR\uparrow}^{tex}}^{\mathrm{T}} \cdot \hat{F}_{Ref}^{tex}\right) = \left\langle \frac{Q_m}{\|Q_m\|}, \frac{K_n}{\|K_n\|} \right\rangle, \tag{6}$$

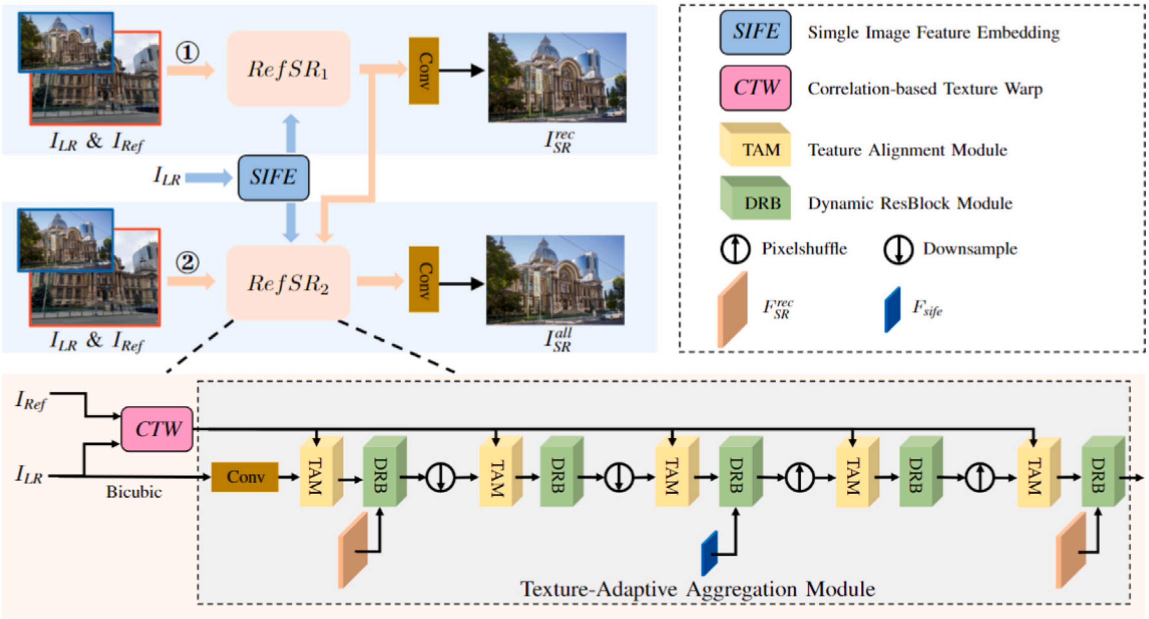


Fig. 1. The architecture of our FRFSR. We first utilized SIFE to reconstruct the features of the I_{LR}, obtaining F_{sife}, which was then embedded into two RefSR models. We eliminated the upsampling and image reconstruction process in SIFE. Next, $RefSR_1$ was trained solely using the reconstruction loss (*-rec*) and then all loss was utilized in training $RefSR_2$. We feed back F_{SR}^{rec}, which $RefSR_1$ reconstructed, during the process into the feature aggregation process to guide $RefSR_2$ in retaining more texture features.

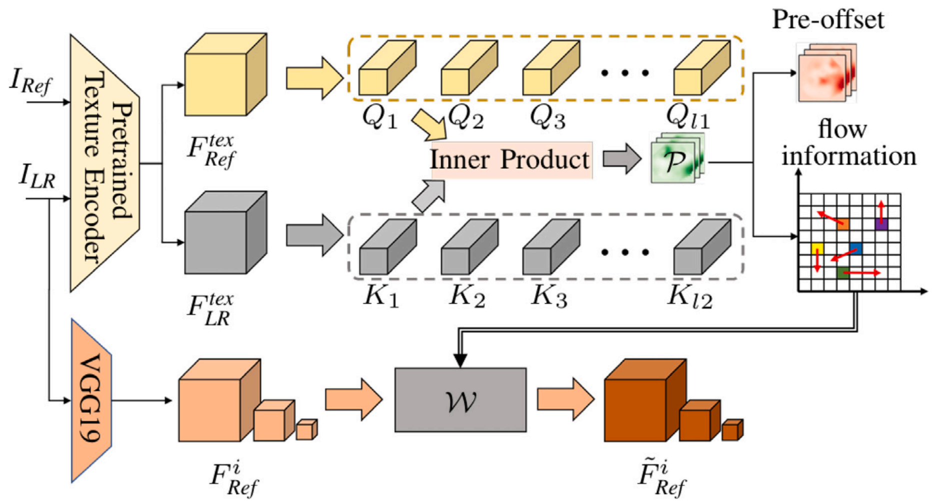


Fig. 2. The architecture of the correlation-based texture warp (CTW).

where $\hat{F}_{LR\uparrow}^{tex}$ and $\hat{F}_{Ref}^{tex}$ respectively represent the patch features of $F_{LR\uparrow}^{tex}$ and F_{Ref}^{tex} after being split into patches. $\frac{Q_m}{\|Q_m\|}$ and $\frac{K_n}{\|K_n\|}$ respectively represent the normalized features of the mth patch in $\hat{F}_{LR\uparrow}^{tex}$ and the nth patch in $\hat{F}_{Ref}^{tex}$, and $\langle\cdot,\cdot\rangle$ represents the inner product operation. ${\hat{F}_{LR\uparrow}^{tex}}^{\mathrm{T}}$ denotes the transpose of $\hat{F}_{LR\uparrow}^{tex}$. For a given patch Q_m in $\hat{F}_{LR\uparrow}^{tex}$, the most similar patch K_n in $\hat{F}_{Ref}^{tex}$ can be found and recorded as P_{max}^m. The index matrix $P = \left\{P_{max}^1, P_{max}^2, \dots, P_{max}^l\right\} \in \mathbb{R}^l$ is formed by recording the indices of these most similar patches.

$$P_{max}^m = \underset{n}{\operatorname{argmax}}\, \mathcal{M}_{m,n}, \tag{7}$$

where $\mathcal{M}_{m,n}$ represents the confidence score of patch K_n corresponding to patch Q_m which is most similar to it. All P_{max}^m form the index matrix P. To use optical flow to initially warp the reference features, we need to convert the index matrix P into flow information. The process is shown below:

$$\left(\mathcal{G}_y, \mathcal{G}_x\right) = \mathbb{G}\left(W_{LR}, H_{LR}\right), \tag{8}$$

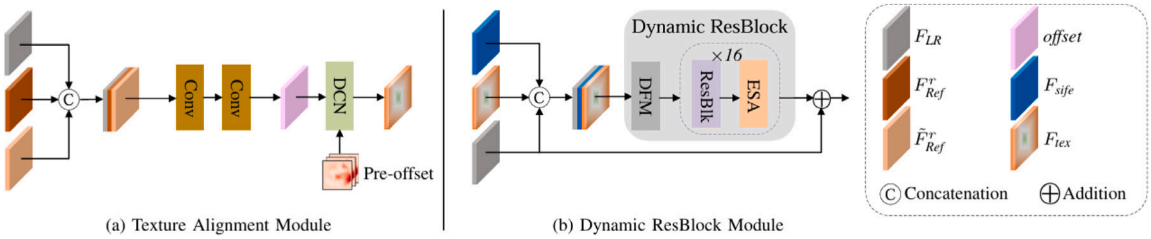
$$\mathcal{F} = \left[P \bmod W_{LR\uparrow}; \lfloor P, W_{LR} \rfloor\right] - \left[\mathcal{G}_x; \mathcal{G}_y\right]. \tag{9}$$

where $[;]$ represents the concatenation of two vectors, $\mathbb{G}(\cdot)$ represents the grid function, which generates a grid with a width of W_{LR} and a height of H_{LR}, x and y denote the coordinate values along the height and width of the grid, The symbols represent the mathematical operations of module and floor division, respectively, and $\mathcal{F}$ represents the flow information.

Finally, we select three different scales of feature maps F_{Ref} extracted by the pre-trained VGG19 [55]. The reason for choosing VGG19 is that it has a strong feature extraction ability and does not require training of additional feature extraction modules. Furthermore, by utilizing flow information at various scales, we distort three reference features F_{Ref}^r at respective scales using optical flow, obtaining three features The detailed procedure is as follows:

$$\left(G_{h_y^r}, G_{h_x^r}\right) = \mathbb{G}\left(W_r, H_r\right), \tag{10}$$

$$h_x^r, h_y^r = split\left(\left[G_{h_y^r}, G_{h_x^r}\right] - \mathcal{F}\right), \tag{11}$$



(a) Texture Alignment Module

(b) Dynamic ResBlock Module

Fig. 3. The structure of the feature alignment module (FAM) and Dynamic ResBlock Module (DRB).

$$G_r = \left[\frac{2 \times h^r_x}{\max\left(W_r - 1, 1\right)} - 1; \frac{2 \times h^r_y}{\max\left(H_r - 1, 1\right)} - 1\right], \tag{12}$$

$$\tilde{F}^r_{Ref} = \mathcal{W}(\left(F^r_{Ref}, G_r\right)), \tag{13}$$

where, H_r and W_r represent the height and width of the rth scale F^r_{Ref}, respectively, $split(\cdot)$ represents the separation of two vectors according to the concatenated channels and $\mathcal{W}(\cdot,\cdot)$ represents the optical flow warping function.

3.3. Texture-adaptive aggregation module

Using an effective texture transfer based on a corresponding matching relationship is another important goal of the RefSR task. To more effectively transfer and aggregate the textures in reference images, we propose a multi-scale dynamic texture transfer module, as shown in the gray background in Fig. 1. In our module, we utilize the multi-scale characteristics to progressively aggregate texture features from multi-scale reference images and learn to generate richer textures. Unlike the direct texture transfer methods used in [5,6], we use specific deformable convolutions [11,12] for texture alignment between F_{LR} and F^r_{Ref} for RefSR tasks, and finally use several Dynamic ResBlock modules to complete texture transfer and aggregation.

3.3.1. Texture alignment module

The image texture offset is typically calculated as the data distribution difference between the source-domain block and the target-domain block. However, due to the lack of positional constraints during the distortion process, the characteristic textures of F^r_{Ref} and $\tilde{F}^r_{Ref}$ may differ from those of F_{LR}. to obtain the offset required for deformable convolution, we concatenate F_{LR}, F^r_{Ref}, and F_{LR} Ref to obtain the offset ΔP_k.

To more accurately transfer the texture features in the multi-scale reference feature F^r_{Ref}, We use specific deformable convolution designed for RefSR to achieve multiple domain-specific mappings $F_{LR} \rightarrow F^r_{Ref}; \tilde{F}^r_{Ref}$, accurately mapping to the corresponding domain for texture alignment. As shown in the flowchart in Fig. 3(a), To achieve accurate alignment, the texture features of F_{LR} are mapped to $F^r_{Ref}; \tilde{F}^r_{Ref}$. The required offset for the mapping $F_{LR} \rightarrow F^r_{Ref}; \tilde{F}^r_{Ref}$ is obtained by aligning the feature distributions of $F^r_{Ref}; \tilde{F}^r_{Ref}$ using deformable convolutions. We concatenate F_{LR}, F^r_{Ref}, and $\tilde{F}^r_{Ref}$, to obtain the stable offset ΔP_k. This is because using the optically distorted reference feature to guide deformable convolution [56] training can make the training process more stable.

$$\Delta P_k = Conv\left(Conv\left(\left(\left[F_{LR}; F^r_{Ref}; \tilde{F}^r_{Ref}\right]\right)\right)\right), \tag{14}$$

where $Conv(\cdot)$ represents the convolution layer. After this, for each patch P_{LR} in LR, we used the previously obtained index matrix P to find the corresponding most similar patch P_{Ref} in F^r_{Ref}. We use ΔP to represent the spatial difference between P_{LR} and P_{Ref}, that is, $\Delta P = P_{LR} - P_{Ref}$, which is the pre-offset output by CTW. Finally, the improved deformable convolution is used to aggregate P_{LR} and its surrounding textures. The specific process is shown below:

$$F^p_{tex} = \sum_{g=1}^{G} \sum_{k=1}^{K} \omega_g \cdot x_g \left(P_{LR} + \Delta P + P_k + \Delta P^k_g\right) \cdot \Delta m^k_g, \tag{15}$$

where G represents the number of groups, K denotes the total number of sampling patches, and k enumerates the sampling patch. ω_g denotes the shared patch irrelevant projection weight of each group, and Δm^k_g denotes the normalized modulation scalar of kth the sampling patch in the gth group, x_g represents the sliced input feature map. ΔP^k_g is the offset corresponding to the grid sampling patch P_k in the g-th group.

By using deformable convolutions to guide the learning of offsets, it becomes possible to calculate the offsets more accurately and achieve texture alignment between P_{LR} and P_{Ref}, the surrounding textures of the most similar patches in each corresponding reference feature can be aggregated, fully utilizing the contextual information in each patch, thus providing a guarantee for subsequent texture transfer.

3.3.2. Dynamic ResBlock module

To effectively aggregate the features of F_{LR}, F_{tex}, and F_{sife}. We propose DRM for self-adapting transfer and aggregating related texture features. Furthermore, to address the challenge of RefSR difficulty in reconstructing high-frequency information from LR alone, we incorporate the output feature F_{sife} of SIFE during aggregation. This feature represents the reconstructed features of LR itself, as shown in the flowchart in Fig. 3(b). Specifically, we concatenate the aligned texture feature F_{tex} with F_{LR} and F_{sife}, and input them into a convolution layer. Then, we use the dynamic residual block to transfer and aggregate the related textures in the reference feature to obtain the output F_{agg}. It is worth noting that we only embed F_{sife} in the DRB module corresponding to the smallest scale, that is, only the feature mapping of the smallest scale is used. The other DRB modules at other scales only aggregate F_{LR} and F_{tex} features.

$$F^{rec}_{agg} = DRB\left(Conv\left(\left[F_{tex}; F_{LR}; F_{sife}\right]\right)\right) + F_{LR}. \tag{16}$$

To train the second model, we reused the feature map F^{rec}_{agg} created by the first model. As a result, we added this feature map to the feature aggregation process to enhance the texture features. Eq. (16) can be expressed in the following form:

$$F^{all}_{agg} = DRB\left(Conv\left(\left[F_{tex}; F_{LR}; F_{sife}; F^{rec}_{agg}\right]\right)\right) + F_{LR}. \tag{17}$$

The DRB module consists of two decoupled dynamic filters and a ResBlock with an ESA (Enhanced Spatial Attention) [57]. In the DRB module, we do not utilize standard convolutions or dynamic filters. The main reason is that standard convolutions lack content awareness and have high computational complexity. Dynamic filters address content adaptation but significantly increase computational complexity. To overcome these limitations, we introduces lightweight decoupled dynamic filter that enhances content-aware perception through lightweight spatial and channel attention branches. Moreover, The parameter count of the dynamic decoupled filters is consistent with that of traditional convolution. The decoupled dynamic filter are shown in

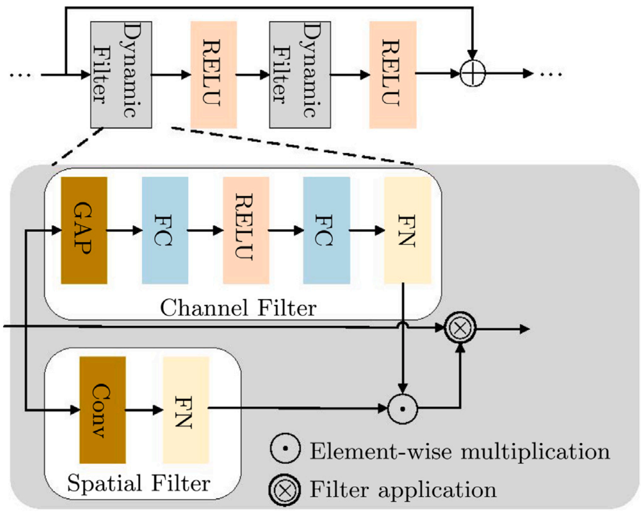


Fig. 4. The structure of dynamic filter module (DFM). 'FC' denotes the fully connected layer and 'GAP' denotes the global average pooling. 'FN' denotes filter normalization.

Fig. 4, inspired by the significant advancements brought by attention mechanisms, by decoupling the dynamic filters into channel filters and spatial filters, we can effectively perceive the related texture content between F_{LR}, F_{tex} and F_{sife}, while addressing the issue of computational intensity. The spatial dynamic filter and channel dynamic filter can be represented by the following equations:

$$H^{sf} = \sum_j \sum_c f_{sf}(i,j) \times F_{(i,j,c)} + b_i, \tag{18}$$

$$H^{cf} = \sum_c \sum_j f_{cf}(j,c) \times F_{(i,j,c)} + b, \tag{19}$$

where $H^{sf}(\cdot)$ denotes to the spatial dynamic filter, $H^{cf}(\cdot)$ denotes the channel dynamic filter at cth channel, b_i is the bias vector, c represents the number of channels. the bias vector b remains unchanged at the channel. Then the routing weight and the final aggregated features can be generated:

$$\mathcal{W}_k = \left(\gamma^{sf}\left(H_i^{sf}\right) + \beta^{sf}\right) \odot \left(\gamma^{cf}\left(H_c^{cf}\right) + \beta^{cf}\right), \tag{20}$$

$$F'_{tex} = \mathcal{W}_k * F_{tex} \tag{21}$$

where, H_i^{sf} and H_i^{cf} represent the values obtained from the spatial and channel filter branches, respectively, after normalization is applied, while $\mathcal{W}_k = \left(\mathcal{W}_1, \mathcal{W}_2, \dots, \mathcal{W}_n\right)$ denotes the routing weights. γ^{sf}, γ^{cf}, β^{sf}, and β^{cf} are similar to BN [58] and specify the learnable mean and standard deviation of the two branches. '$\odot$' and '$*$' are used to denote element-wise multiplication and the filter application, respectively.

After the decoupled dynamic filtering, we have effectively perceived the relevant texture content between F_{LR}, F_{tex} and F_{sife}. However, as shown in the feature visualization in Fig. 13, although the dynamic filter can effectively perceive and aggregate relevant reference textures, it also leads to texture misuse. Therefore, we embedded ESA into the residual block, as shown in Fig. 5, to enhance the related texture features of F_{LR}, aggregate reference features with high relevance while suppressing interference features with low relevance. We also conducted experiments to demonstrate its effectiveness, and the feature visualization of the ESA module in Fig. 14 shows that ESA can sharpen features and weaken the introduction of irrelevant textures. It is worth noting that this attention module is lightweight and only adds a small number of parameters.

ESA has been proven to be efficient and effective in previous work [57,59]. This is because it uses 1 × 1 convolution and 3 × 3 convolution with a stride of 2 to compress the channel size and spatial size respectively, and further reduces the feature size using max pooling.

This attention-based texture-adaptive aggregation method not only transfers and fuses effective textures from reference images and reduces interference from irrelevant textures, it also ensures that the features F_{sife} reconstructed by the SISR method are well integrated into F_{LR}. By aggregating F_{sife} features, it not only makes up for the defect that reference-based super-resolution is difficult to reconstruct its own texture, it also suppresses the generation of irrelevant textures to a large extent.

3.4. Loss functions

Reconstruction loss. To ensure the model has an excellent texture transfer ability and image reconstruction ability, we use the following reconstruction loss to train the model.

$$\mathcal{L}^{rec} = \left\|I_{HR} - I_{SR}\right\|_1, \tag{22}$$

where I_{HR} represents the ground truth image, I_{SR} represents the super-resolved image. $\|\cdot\|_1$ represents l_1 norm. Only using reconstruction loss to train the model will cause the image to be too smooth.

Perceptual loss. By calculating perceptual loss [60] in the feature domain, the generated image can be more semantically similar to GT. Perceptual loss is shown as follows:

$$\mathcal{L}^{per} = \frac{1}{V}\sum_{i=1}^{C}\left\|\phi_i\left(I_{HR}\right) - \phi_i\left(I_{SR}\right)\right\|_F, \tag{23}$$

where $\phi_i(\cdot)$ represents the ith intermediate layer of VGG19 [55]. $\|\cdot\|_F$ represents Frobenius norm, C and V represent the number of channels and volume of feature maps respectively.

Adversarial loss. The generator G and discriminator $\mathcal{D}$ improve together in a game against each other, ensuring the model is able to generate output images with pleasing visual effects. The adversarial loss we choose is WGAN [61], which is shown as follows:

$$\mathcal{L}^{adv} = -\mathcal{D}\left(I_{SR}\right). \tag{24}$$

During the training process, the loss of discriminator $\mathcal{D}$ is shown as follows:

$$\mathcal{L}^{D} = \mathcal{D}\left(I_{SR}\right) - \mathcal{D}\left(I_{GT}\right) + \lambda\left(\left\|\nabla_{\hat{I}}\mathcal{D}(\hat{I})\right\|_2 - 1\right)^2, \tag{25}$$

where $\nabla_{\hat{I}}$ represents the random convex combination of I_{HR} and I_{SR}.

Finally, the total loss function is shown as follows:

$$\mathcal{L}^{all} = \lambda_1\mathcal{L}^{rec} + \lambda_2\mathcal{L}^{per} + \lambda_3\mathcal{L}^{adv}, \tag{26}$$

where λ_1, λ_2, and λ_3 are respectively the weight coefficients for each loss.

4. Experiments

This section commences by presenting the datasets essential to the training and testing of the models. We utilized PSNR (Peak Signal to Noise Ratio), SSIM (Structural Similarity), and LPIPS (Learned Perceptual Image Patch Similarity) as quantitative comparison metrics. PSNR serves as a metric for evaluating image quality, while SSIM quantifies the structural similarity between two images. Their formulas are shown as (27) and (28) respectively. LPIPS is also commonly employed to measure perceptual similarity between two images, making it more aligned with human perception. Its formula is consistent with the perceptual loss. Subsequently, we comparatively analyze several super-resolution methods along various aspects for our approach. Ablation studies are conducted on the SIFE and DRB components, along with the feature reuse framework. Lastly, we evaluate the efficacy of our proposed approach against other super-resolution methods in a practical implementation.

$$PSNR(I_{HR}, I_{SR}) = 10 \cdot \log_{10}\left(\frac{255^2}{MSE(I_{HR}, I_{SR})}\right) \tag{27}$$

$$SSIM(I_{HR}, I_{SR}) = \frac{(2\mu_{I_{HR}}\mu_{I_{SR}} + c_1)(2\sigma_{I_{HR}}\sigma_{I_{SR}} + c_2)}{(\mu_{I_{HR}}^2 + \mu_{I_{SR}}^2 + c_1)(\sigma_{I_{HR}}^2 + \sigma_{I_{SR}}^2 + c_2)} \tag{28}$$

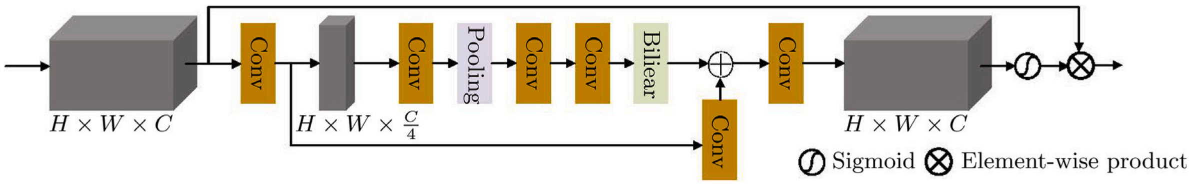


Fig. 5. The structure of Enhanced Spatial Attention (ESA).

Table 1
PSNR/SSIM are the evaluation metrics uesd to compare the other methods quantitatively. The model is trained using only reconstruction loss (-*rec*). Text highlighted in bold indicates the most favorable outcome.

Method	CUFED [5] PSNR/SSIM	SUN80 [64] PSNR/SSIM	Urban100 [62] PSNR/SSIM	Manga [63] PSNR/SSIM	WR-SR [8] PSNR/SSIM
SRCNN [14]	25.33/0.745	28.26/0.781	24.41/0.738	27.12/0.850	27.27/0.767
EDSR [16]	25.93/0.777	28.52/0.792	25.51/0.783	28.93/0.891	28.07/0.793
ENet [65]	24.24/0.695	26.24/0.702	23.63/0.711	25.25/0.802	25.47/0.699
RCAN [18]	26.33/0.781	29.97/0.814	25.99/0.787	30.11/0.908	27.91/0.793
RRDB [21]	26.41/0.783	29.99/0.814	25.98/0.788	29.87/0.907	27.96/0.793
Cross-Net [3]	25.48/0.764	28.52/0.793	25.11/0.764	23.36/0.741	–
SSEN-*rec* [4]	26.78/0.791	–	–	–	–
SRNTT-*rec* [5]	26.24/0.784	28.54/0.793	25.50/0.783	28.95/0.885	27.59/0.780
TTSR-*rec* [6]	27.09/0.804	30.02/0.814	25.87/0.784	30.09/0.907	27.97/0.792
MASA-*rec* [7]	27.54/0.814	30.15/0.815	26.09/0.786	30.28/0.909	28.19/0.796
C^2-Matching-*rec* [8]	28.24/0.841	30.18/0.817	26.03/0.785	30.47/0.911	28.32/0.801
WTRN-*rec* [30]	27.33/0.810	30.11/0.816	26.00/0.787	30.37/0.909	–
TADE-*rec* [13]	28.64/0.850	30.31/0.820	26.71/0.807	**31.23/0.917**	28.34/0.805
DATSR-*rec* [9]	28.72/0.856	30.20/0.818	26.52/0.798	30.49/0.912	28.52/0.807
RRSR-*rec* [32]	28.83/0.856	30.13/0.816	26.21/0.790	30.91/0.913	28.41/0.804
FRFSR-*rec* (Ours)	**29.18/0.865**	**30.35/0.822**	**26.84/0.811**	31.15/**0.917**	**28.67/0.811**

4.1. Datasets and metrics

4.1.1. Training dataset

We use CUFED [5] to train our model, which consists of a total of 11,871 image pairs, comprising 11,871 input images along with their corresponding reference images, each with a resolution of 160 × 160.

4.1.2. Testing dataset

Our study evaluates the efficiency of our model across five benchmark datasets: CUFED5 [5], Urban100 [62], Manga109 [63], Sun80 [64], and WR-SR [8]. CUFED5 contains a total of 126 input images, each accompanied by five reference images of decreasing relevance, denoted as L1-L4. Urban100 comprises primarily 100 urban images known for their strong self-similarity. During testing, we utilized the low-resolution input images themselves as reference images. Manga109 consists of 109 manga images commonly used in super-resolution tasks. We randomly select one image from the remaining 108 images as the reference image. For Sun80 consists of 80 natural images, with each input image having 20 corresponding reference images. During testing, we randomly selected one reference image for evaluation. WR-SR, proposed in [8], offers a richer variety of scenes and categories compared to CUFED5. It consists of 80 pairs of input images and reference images, providing a more comprehensive evaluation of RefSR performance. Our metrics for evaluation consisted of PSNR and SSIM calculated on the Y channel in the YCbCr color space.

4.1.3. Implementation details

To obtain the LR inputs, we downsample the HR images by a scale factor of 4. For data augmentation, we apply horizontal flip, vertical flip, and random rotation. To increase the training difficulty and improve the performance of long-distance feature alignment, we divide the reference images into patches and shuffle them randomly. We use the official RRDB [21] parameters as the pre-trained model for the single image feature embedding module, which we train in two stages. First, we use $\mathcal{L}^{rec}$ as the only loss function. Second, we use $\mathcal{L}^{rec}$, $\mathcal{L}^{per}$, and $\mathcal{L}^{adv}$ for joint supervision. During the training process, we choose the Adam optimizer and set the β_1 and β_2 parameters to 0.99 and 0.999, respectively. We set the initial learning rate of the model to 1e–4 and the batch size to 9. The weights λ_1, λ_2, and λ_3 of $\mathcal{L}^{rec}$, $\mathcal{L}^{per}$, and $\mathcal{L}^{adv}$ are set to 1.0, 10^{-4}, and 10^{-6}, respectively. Our model has a floating-point operation complexity of approximately 116 GFLOPs and took approximately 56 h to train on two NVIDIA GeForce RTX 3090 GPUs.

4.2. Comparison with state-of-the-art methods

We conduct quantitative and qualitative comparisons between our proposed method and some existing SISR and RefSR methods. The SISR methods are SRCNN [14], EDSR [16], RCAN [18], Enet [65], SRGAN [20], ESRGAN [21], RankSRGAN [22]. The RefSR methods are CrossNet [3], SSEN [4], SRNTT [5], TTSR [6], MASA [7], C^2-Matching [8], TADE [13], DATSR [9], and RRSR [32]. We train two sets of parameters, one using only the reconstruction loss (denoted by $-rec$), and the other using all losses.

4.2.1. Quantitative comparison

As shown in Table 1, our method achieves state-of-the-art results on five benchmark datasets using only the reconstruction loss. Our method leverages effective texture matching, dynamic texture transfer, and complementary SISR features in the reconstruction process, which enables it to transfer similar textures from the high-resolution reference images in CUFED5 and WR-SR datasets to the LR images, enhancing their high-frequency information, and to transfer self-features to assist LR reconstruction on the self-similar dataset Urban100. As shown in Table 3, our model outperforms all the other methods on all datasets under the joint supervision of losses, although its performance slightly degrades compared to the results obtained when only using reconstruction loss. Interestingly, our method still maintains a significant advantage (+0.8 dB) over the other RefSR methods, even with the presence of perceptual loss and adversarial loss. Furthermore, our method consistently achieves lower values for both Perceptual Index (PI) and Fréchet Inception Distance (FID) compared to the other three methods (C^2-Matching, DATSR, RRSR) across the majority of datasets, as illustrated in Table 2. The quantitative comparison under the two

Table 2
Three different methods, including C^2-Matching, DATSR, and RRSR, are quantitatively compared in terms of Perceptual Index (PI) and Fréchet Inception Distance (FID). It is noteworthy that the loss weights for these three methods are set consistently with our approach.

Method	CUFED5		Sun80		Urban		Manga	
	PI↓	FID↓	PI↓	FID↓	PI↓	FID↓	PI↓	FID↓
C^2-Matching [8]	2.6758	45.73	4.2813	16.61	4.0452	25.49	3.9057	13.36
DATSR [9]	2.6234	43.48	4.2801	16.60	4.0936	24.53	3.9127	13.12
RRSR [32]	2.4791	41.23	4.2752	16.57	**4.0628**	22.41	3.8149	12.95
FRFSR (Ours)	**2.4453**	**39.76**	**4.2704**	**15.69**	4.0645	**20.96**	**3.7854**	**11.67**

Table 3
The model is trained using all losses and the results are compared with PSNR/SSIM. Text highlighted in bold indicates the most favorable outcome.

Method	CUFED [5]	SUN80 [64]	Urban100 [62]	Manga [63]	WR-SR [8]
	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
SRGAN [20]	24.40/0.702	26.76/0.725	24.07/0.729	25.12/0.802	26.21/0.728
ESRGAN [21]	21.90/0.633	24.18/0.651	20.91/0.620	23.53/0.797	26.07/0.726
RankSRGAN [22]	22.31/0.635	25.60/0.667	21.47/0.624	25.04/0.803	26.15/0.719
SRNTT [5]	25.61/0.764	27.59/0.756	25.09/0.774	27.54/0.862	26.53/0.745
TTSR [6]	25.53/0.765	28.59/0.774	24.62/0.747	28.70/0.886	26.83/0.762
MASA [7]	24.92/0.729	27.12/0.708	23.78/0.712	27.34/0.848	25.76/0.717
C^2-Matching [8]	27.16/0.805	29.75/0.799	25.52/0.764	29.73/0.893	27.80/0.780
WTRN [30]	25.98/0.761	28.46/0.756	24.88/0.747	29.18/0.878	–
TADE [13]	27.37/0.816	28.85/0.768	25.80/0.776	30.12/0.889	27.40/0.769
DATSR [9]	27.95/0.835	29.77/0.800	25.92/0.775	29.75/0.893	27.87/0.787
RRSR [32]	28.09/0.835	29.57/0.793	25.68/0.767	29.82/0.893	27.89/0.784
FRFSR (Ours)	**28.71/0.852**	**29.89/0.804**	**26.65/0.802**	**30.89/0.906**	**28.27/0.793**

Table 4
Performance comparison under different relevance levels on CUFED5.

Method	L1		L2		L3		L4	
	PSNR/SSIM	LPIPS	PSNR/SSIM	LPIPS	PSNR/SSIM	LPIPS	PSNR/SSIM	LPIPS
Cross-Net [3]	25.48/0.764	–	25.48/0.764	–	25.47/0.763	–	25.46/0.763	–
SRNTT-*rec* [5]	26.15/0.781	0.248	26.04/0.776	0.252	25.98/0.775	0.258	25.95/0.774	0.261
SSEN-*rec* [4]	26.78/0.791	–	26.52/0.783	–	26.48/0.782	–	26.42/0.781	–
TTSR-*rec* [6]	26.99/0.800	0.230	26.74/0.791	0.239	26.64/0.788	0.244	26.58/0.787	0.251
MASA-*rec* [7]	27.35/0.814	0.205	26.92/0.796	0.232	26.82/0.793	0.238	26.74/0.790	0.242
C^2-Matching-*rec* [8]	28.24/0.841	0.170	27.39/0.813	0.193	27.17/0.806	0.204	26.94/0.799	0.230
WTRN-*rec* [30]	27.23/0.807	0.236	26.90/0.794	0.236	26.79/0.792	0.240	26.71/0.789	0.245
TADE-*rec* [13]	28.64/0.850	–	27.77/0.821	–	27.46/0.815	–	27.23/0.807	–
DATSR-*rec* [9]	28.50/0.850	0.166	27.47/0.820	0.209	27.22/0.811	0.218	26.96/0.803	0.2281
RRSR-*rec* [32]	28.64/0.850	0.161	27.77/0.821	0.201	27.46/0.815	0.211	27.23/0.807	0.223
FRFSR-*rec* (Ours)	**29.01/0.860**	**0.152**	**28.01/0.831**	**0.189**	**27.77/0.824**	**0.198**	**27.49/0.815**	**0.209**

paradigms demonstrates that our model exhibits a strong generalization ability and achieves optimal performance.

4.2.2. Qualitative evaluation

Figs. 6 and 7 shows the visual comparison of our model and the existing SISR and RefSR methods. It can be clearly seen that RCAN and RRDB have difficulty in reconstructing texture information due to the severe degradation of high-frequency information, especially in text and texture-dense areas. Compared with SISR, RefSR can transfer similar textures from the reference images, thus producing more texture details. Compared with some existing RefSR methods, the adaptive nature of FRFSR allows for the perception and transferal of texture information from the Ref images. Thus, the model is capable of compensating for missing high-frequency details in LR, leading to the reconstruction of images with texture details more closely resembling the ground truth. For example, in the third pair of local details in Fig. 6, RCAN and RRDB fail to reconstruct any window blind texture, and the existing RefSR methods generate some texture details, but the images are very unrealistic and far from the ground truth. Our proposed method can generate a sharper, clearer blind texture that is very close to the ground truth. This demonstrates the effectiveness of our texture search and texture-adaptive aggregation methods. Due to the feature reuse framework, FRFSR can preserve increasingly more realistic texture information when trained with $\mathcal{L}^{rec} + \mathcal{L}^{per} + \mathcal{L}^{adv}$, such as the text on the clothes in the fourth pair of images in Fig. 7, and the stone pillar texture in the second pair of images. Compared with the other RefSR methods, our method can generate complete text texture and stone pillar texture, reflecting the advantages of the feature reuse framework and our method.

4.2.3. Comparison of robustness of texture transformations

Texture transfer robustness is an important criterion for evaluating the performance of RefSR models. As shown on the left of Fig. 8, SOTA methods suffer from texture mis-transfer. Moreover, even if the texture of the Ref image is irrelevant, the model should exhibit good adaptive texture transfer robustness. CUFED5 provides four reference images with different levels of relevance (L1-L4). Table 4 shows the results of different models under different relevance settings. The results demonstrate that our model surpasses several existing RefSR models in terms of texture transfer and robustness. Notably, especially when the reference image is least relevant, our model achieves a performance gain of 0.26 dB over other SOTA models. To further validate the superiority of our model, we created the CUFEDR dataset, which extended all HR images from Urban100, Manga109 and Sun80 into a reference set, consisting of 289 images. During testing, we randomly selected one HR from CUFEDR as a reference image for testing, and the test results are shown in Table 5. Even if the reference image is irrelevant, our model outperforms RRSR by 0.3 dB, as shown on the right of Fig. 8. These experiment results indicate that our model can match and transfer similar textures from relevant reference images, and also possesses adaptive texture transfer robustness in low-relevance scenarios.

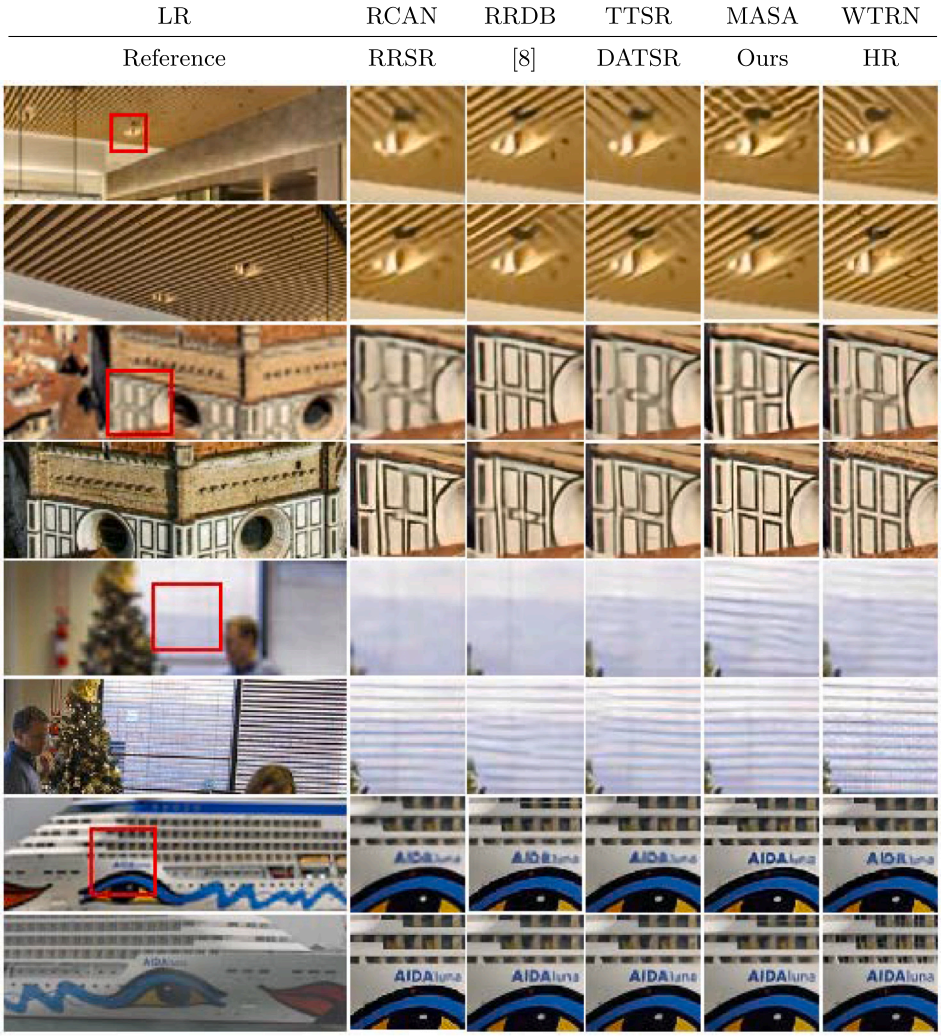


Fig. 6. Qualitative comparison of the SISR and RefSR methods. All these methods are trained with the L1 loss. It can be seen from the figure that our method can transfer and reconstruct more texture details from the reference images.

Table 5
Robustness comparison for irrelevant texture transfer on CUFEDR testing set.

Method	CUFEDR	
	PSNR/SSIM	LPIPS
TTSR-*rec* [6]	26.40/0.778	0.273
MASA-*rec* [7]	26.59/0.784	0.260
C^2-Matching-*rec* [8]	26.50/0.784	0.265
DATSR-*rec* [9]	26.43/0.784	0.267
RRSR-*rec* [32]	26.58/0.785	0.259
Ours-*rec*	**26.88/0.795**	**0.241**

4.2.4. Comparison of robustness of long-range alignment

To enhance the robustness of our model with respect to long-distance feature alignment, we integrated training with long-distance alignment and context perturbation samples. Specifically, we divide the reference image into multiple n × n patches, then randomly shuffle their positions, and finally reassemble them into a new complete sample image, which disturbs the contextual dependency of the image and enlarges the misalignment distance between relevant patches. During testing, we perform three different levels of random shuffling on the reference image, namely easy, medium, and hard, which divide the image into 2×2, 4×4, and 8 × 8 patches respectively, as shown in Fig. 9. Fig. 10 shows our model and the other RefSR method for different levels of randomly shuffled reference images. It is worth noting that we refrained from retraining the model with the strategy of random shuffling. We observed that, although this strategy did not lead to improvements in PSNR, it maintained a certain level of robustness under varying degrees of confusion in reference images. By using cross-layer semantic regularization to fuse and enhance texture features with similar semantics at different granularities, we show that our model is more robust than the C^2-Matching method. It is worth noting that we only use medium-level data augmentation during training.

4.3. Discussion of model size and computation cost

In this section, we compare our method with other methods in terms of parameter size and running time, as shown in Tables 7 and 6.

Our model improves the running time by 33.5% and reduces the parameters by 25% compared to DATSR [9], which is based on Swin-Transformer [10] as the basic module. However, compared to MASA's [7] coarse-to-fine matching method, our method increases both running time and parameters, but our method greatly improves the performance. our model consists of two parts: relevant texture search and transfer with SIFE, the model parameters are relatively large. The SIFE module can reduce the introduction of some texture error, so it plays an auxiliary role. It is worth noting that after removing the SIFE module, the parameters are only 13.5M, but the performance can still reach SOTA.

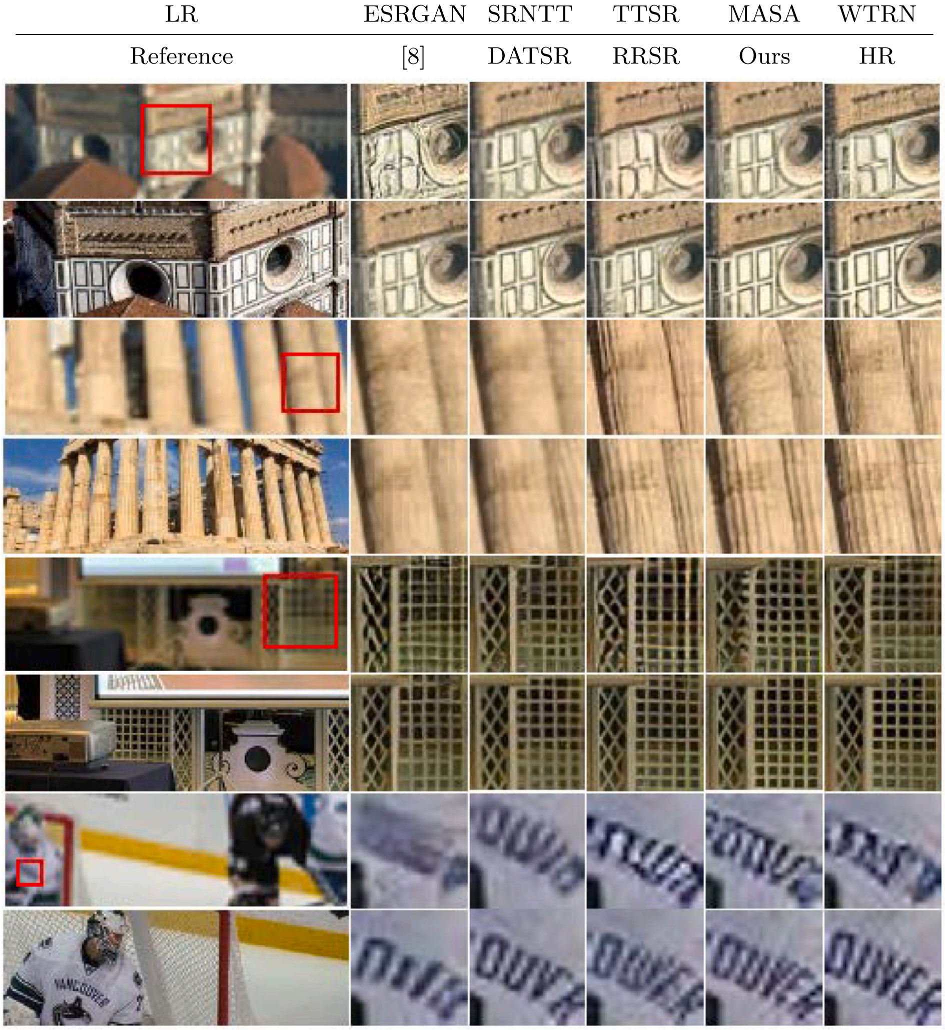


Fig. 7. Qualitative comparison of the SISR and RefSR methods. In this category of methods, perceptual loss and generative adversarial loss have been incorporated.

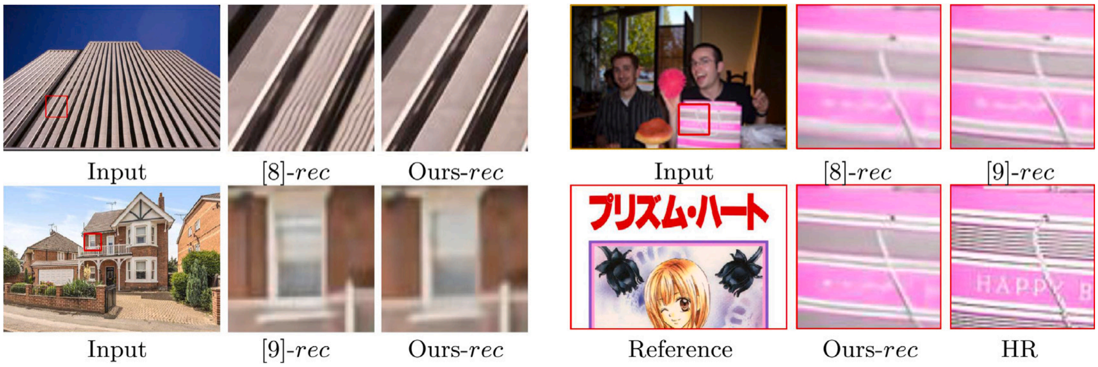


Fig. 8. To the left is an exemplification of reference misuse in present methods, whereas to the right stands a texture reconstruction image with a reference unrelated to the original.

4.4. Ablation studies

In this section, we evaluate our method's dynamic residual block component and single image feature embedding using CUFED5. Table 8 shows the evaluation results. We also apply the feature reuse framework to other RefSR methods to demonstrate its effectiveness (see Fig. 15).

4.4.1. Single image feature embedding

The reconstructed features from SISR can effectively compensate for the remaining features other than the texture features in the Ref image. To verify the effectiveness of single-image feature embedding, we do not consider feature embedding when transferring texture, and we find that both the performance of transferring matching texture from the Ref image and reconstructing a similar texture that does not exist in

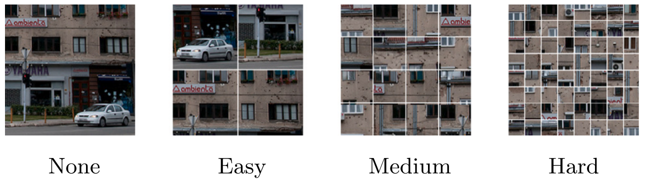


Fig. 9. Diagram of different degrees of random disruption.

Table 6
Running time of FRFSR compared with other RefSR methods on CUFED5.

Model	Runtime (ms)
SRNTT [5]	13 256
TTSR [6]	505
MASA [7]	336
C^2-Matching [8]	361
DATSR [9]	1214
FRFSR (Ours)	807

Table 7
Comparison between our FRFSR and other RefSR methods in the number of parameters.

Model	Params	PSNR↑/SSIM↑
CorssNet [3]	33.18M	25.48/0.764
C^2-Matching-*rec* [8]	8.9M	28.24/0.841
TADE-*rec* [13]	10.9M+15.9M	28.64/0.850
RRSR-*rec* [32]	22.6M	28.83/0.856
Ours-*rec* (w/o SIFE)	13.5M	28.93/0.865
Ours-*rec*	13.5M+15.9M	29.16/0.865

Table 8
Quantitative evaluation of the ablation study on the single-image feature embedding module and dynamic residual ResBlock component on the CUFED5.

Model	SIFE	DRB	ESA	PSNR↑/SSIM↑	LPIPS↓
Baseline(DATSR)				28.72/0.856	0.1630
Baseline+DRB		✓		28.88/0.860	0.1592
Baseline+DRB+ESA		✓	✓	28.93/0.865	0.1563
Baseline+SIFE	✓			29.01/0.861	0.1462
Baseline+DRB+SIFE+ESA	✓	✓	✓	29.16/0.865	0.1431

the Ref image are affected. Table 9 shows that with the help of the SIFE module, our model not only improved by 0.37 dB on CUFED5, but also achieved corresponding improvements on other datasets. In addition, we also compared our model with the decoupled and coupled frameworks in TADE [13] at the same time, further demonstrating the effectiveness of SIFE. It can be observed that the coupled model achieves better metrics on high-resolution reference datasets with similar textures, such as CUFED5 and WR-SR. However, its performance is relatively poorer on datasets like Urban100 and Manga109, which lack similar high-resolution textures. This phenomenon suggests that the coupled model possesses stronger texture aggregation capabilities. In contrast, the decoupled model, leveraging features reconstructed from upsampled SISR methods, has richer inherent feature information. Therefore, it is more suitable when the reference image lacks relevant textures. However, RefSR methods prioritize matching and aggregating reference textures. Hence, we consider that having stronger texture transfer and aggregation capabilities is crucial for RefSR. Consequently, we ultimately opt for the coupled framework. On the other hand, as shown in Fig. 11, adding the SIFE module can facilitate the model's learning, alleviate detail loss, and not only transfer richer and finer texture details from the Ref images in CUFED5, it can also make texture features more prominent in the SR images reconstructed on other datasets. It is worth noting that adding the SIFE module can suppress irrelevant texture transfer to some extent, as shown in the third row. Through quantitative and qualitative evaluation, the SIFE module improves the model's ability to transfer texture and recover texture details that do not exist in the Ref image.

Table 9
Quantitative ablation experiments of SIFE on multiple benchmark datasets. In addition, we also chose the decoupled and non-decoupled frameworks in TADE for comparison.

Method	CUFED5	Sun80	Urban100	WR-SR
	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM	PSNR/SSIM
Decouple	28.93/0.859	**30.38/0.821**	**26.89/0.811**	28.41/0.796
w/o SIFE	28.68/0.853	30.07/0.814	25.95/0.779	28.24/0.794
w/ SIFE (Couple)	**29.01/0.861**	30.35/0.821	26.80/0.810	**28.52/0.806**

Table 10
Quantitative ablation study on adding FRF on multiple methods.

Model	FRF	PSNR↑/SSIM↑	LPIPS↓
MASA [7]		24.92/0.729	0.0987
MASA+FRF	✓	25.16/0.744	0.0954
C^2-Matching [8]		27.16/0.805	0.1229
C^2-Matching+FRF	✓	28.05/0.834	0.1198
Ours		28.29/0.840	0.0992
Ours+FRF	✓	28.71/0.852	0.0974

4.4.2. *Dynamic residual block*

The aligned reference features contain a lot of noise information, and using ResBlock to directly aggregate the reference features will cause the SR image to have irrelevant textures and noise. As shown in Fig. 12, we add dynamic filters and enhanced spatial attention (ESA) in the residual block, which can effectively perceive relevant textures and adaptively aggregate them. Even though dynamic filters can effectively select reference textures, the feature visualization in Fig. 13 reveals that there are still many non-relevant textures present after dynamic filtering. Therefore, we add ESA to the ResBlock to further eliminate non-relevant textures. Fig. 14 shows the feature visualization of ESA. It can be seen that after adding ESA, the texture features with higher relevance become more prominent, and the texture edges become sharper. As shown in Table 8, compared with Baseline, the model with DRB has a 0.25 dB improvement on PSNR and the LPIPS [66] also decreased by 0.0067, and the smaller LPIPS corresponds to better performance.

4.4.3. *Feature reuse framework*

We found that compared with the model trained with reconstruction loss, the model trained with all loss exhibited a worse performance on texture transfer and reconstruction. To reduce the impact of adversarial loss and perceptual loss, we used a Feature Reuse Framework (FRF) to supplement the texture that could not be reconstructed. Table 10 shows the effect of FRF on MASA and C^2-Matching. It can be seen that after adding FRF, all models consistently improved. Although PSNR and SSIM cannot determine visual quality, we also use LPIPS as an evaluation indicator. It is noted that our model is slightly lower than MASA on LPIPS, which is because MASA uses larger weights for adversarial loss and perceptual loss, resulting in better visual effects for the final output. Our loss weights are consistent with C^2-matching, but compared with C^2-Matching with FRF added, our model improved by 0.66 dB and 0.018 on PSNR and SSIM respectively, and LPIPS decreased by 0.0224. We visualize the methods with FRF added, and the visualization results are shown in Fig. 15. It is noted that the model trained with only reconstruction loss (-*rec*) has more texture details than the model trained with all losses. However, after adding FRF, the texture can be restored to normal. This indicates that this framework can reduce the impact of adversarial loss and perceptual loss on texture reconstruction.

4.4.4. *Web search applications*

Image search by image is a common feature of the existing network, which is also one of the most typical applications of RefSR. By searching for reference images based on the user input LR image, RefSR can reconstruct LR. At the same time, this is also a way to verify the generalization ability of RefSR. We selected two low-resolution images

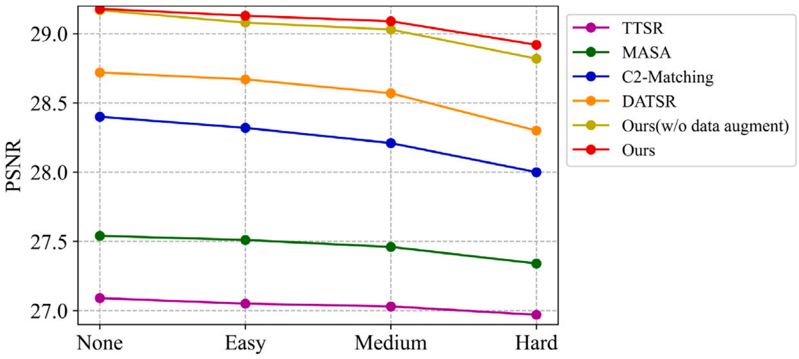


Fig. 10. Robustness of different models in long distance feature alignment. Our FRFSR is better than TTSR, MASA and C^2-Matching with varying levels of graphic clutter.

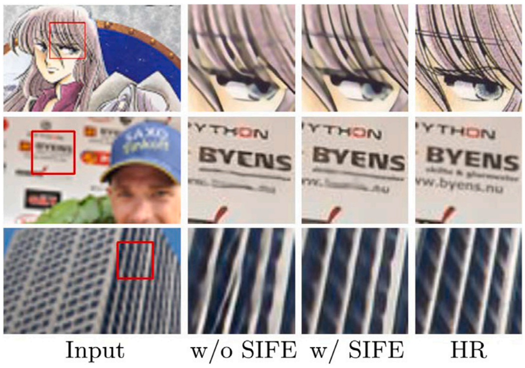


Fig. 11. Ablation analysis of the SIFE module. In addition to boosting the performance of texture transfer and reconstruction, the SIFE module effectively suppresses irrelevant textures from being introduced.

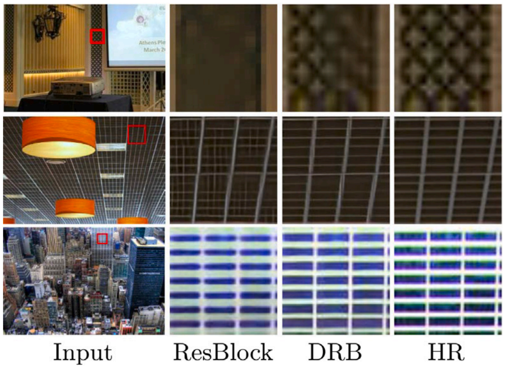


Fig. 12. Replacing ResBlock with DRB can effectively enhance the network's ability to learn texture transfer, suppress the introduction of unrelated texture, and enable the generated texture to approach the GT.

Fig. 13. Visualization of features at different stages on the 4× feature scale. From left to right are feature visualization after PiexlShuffle, features before input Dynamic Filter, and features after output Dynamic Filter.

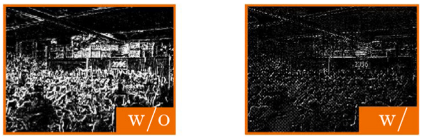


Fig. 14. Feature visualization of Enhanced Spatial Attention (ESA). After undergoing ESA processing, a substantial amount of noisy textures have been eliminated, and the textures have become sharper.

from DIV2K and RealSR, respectively, and utilized Google's image search function to find corresponding reference images. For the RealSR input images, we introduced random Gaussian noise, JPEG noise, and Poisson noise. Our FRFSR method was then compared with other SISR and RefSR methodologies, and the results are presented in Fig. 16. Compared with ESRGAN, and existing RefSR methods (SRNTT, TTSR, MASA, C^2-Matching, DATSR), our method can transfer more details and textures from the images found in the web, even if there are differences in lighting, texture size or perspective in the reference image. Therefore, the SR image reconstructed by our method has a better visual quality.

4.4.5. *Discussion in remote sensing*

RefSR has been applied in various fields such as remote sensing [67], thanks to its excellent texture transfer and reconstruction performance. Therefore, to further highlight the advantages of our method, we have added a performance comparison of multiple RefSR methods on remote sensing images in this section. Typically, remote sensing datasets lack corresponding reference images. Thus, we selected one image from the HRSCD dataset [68] for downsampling as the input image and found another image with similar textures from the same dataset as the reference image. Both qualitative and quantitative comparisons are depicted in Fig. 17. It can be observed that our method still performs well in reconstructing textures in remote sensing images, demonstrating the scalability of our approach.

5. Conclusion

In this paper, we introduce a feature reuse framework that successfully mitigates the negative impacts of the perceptual and adversarial losses that arise during the texture reconstruction process. Our method is composed of two modules: a single-image feature embedding module for reconstructing the self-features of the LR input image, and a texture adaptive aggregation module for reconstructing the effective texture of the perceptual aggregate reference image. Our approach improves

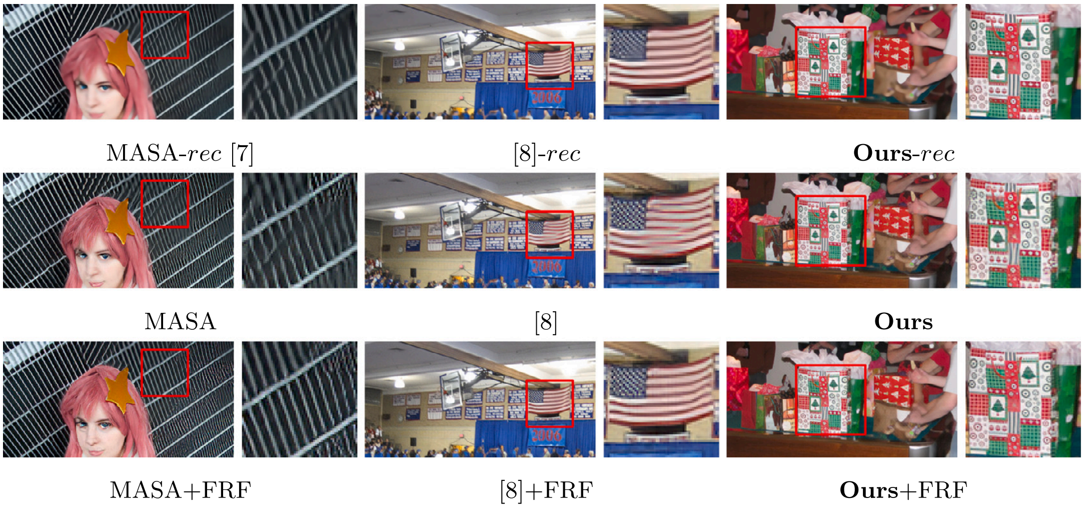


Fig. 15. Qualitative ablation study on adding FRF (Feature Reuse Framework) to multiple methods, where the first row is the SR results of each model trained with only reconstruction loss, the second row is the output results of models trained with all losses, and the third row is the output results of each model after adding FRF.

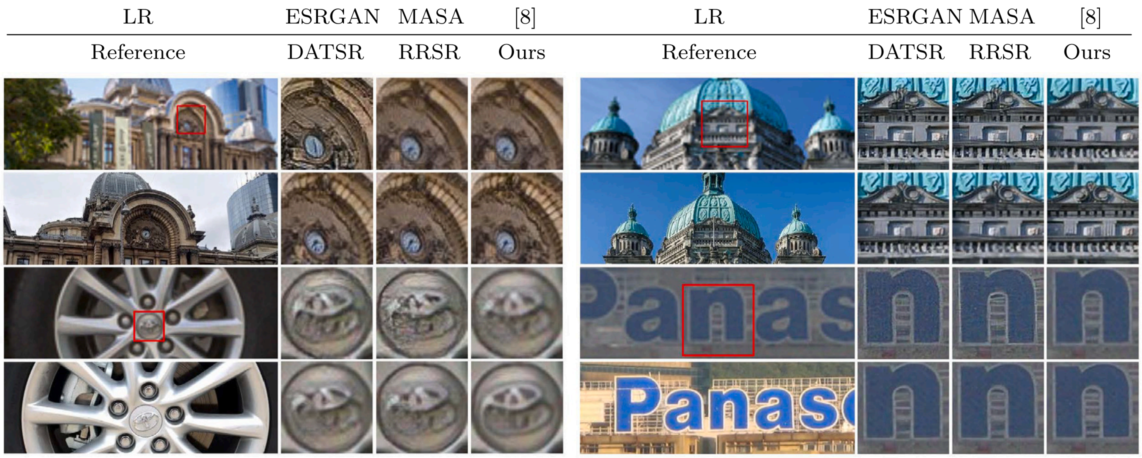


Fig. 16. Web search applications on multiple methods. The first line involves selecting two images from DIV2K for traditional downsampling, followed by searching for similar images on Google. The second line involves choosing two high-resolution images from the RealSR dataset and introducing random Gaussian noise, JPEG noise, and Poisson noise.

robustness to unrelated references. The experiments conducted on various benchmarks show that our approach outperforms existing RefSR methods in both qualitative and quantitative measures.

CRediT authorship contribution statement

Xiaoyong Mei: Writing – original draft, Validation, Methodology, Investigation, Conceptualization. **Yi Yang:** Writing – original draft, Visualization, Validation, Software, Methodology, Conceptualization. **Ming Li:** Writing – original draft, Validation, Supervision, Methodology, Conceptualization. **Changqin Huang:** Writing – review & editing, Visualization, Validation. **Kai Zhang:** Writing – review & editing, Visualization, Validation. **Fudan Zheng:** Writing – review & editing, Visualization, Validation.

Declaration of competing interest

The authors declare that there are no known competing financial interests or personal relationships that might have influenced the work reported in this paper. We affirm that our research has been conducted without bias, and all findings are presented in an objective and transparent manner.

Acknowledgments

This work was supported in part by the National Key Research and Development Program of China (No. 2022YFC3303600), and in part by the Research Project on Ideological and Political Work under the Philosophy and Social Science Planning Program of Zhejiang Province (No. 25GXSZ009YB). Ming Li acknowledged the support from Jinhua Science and Technology Plan (No. 2023-3-003a).

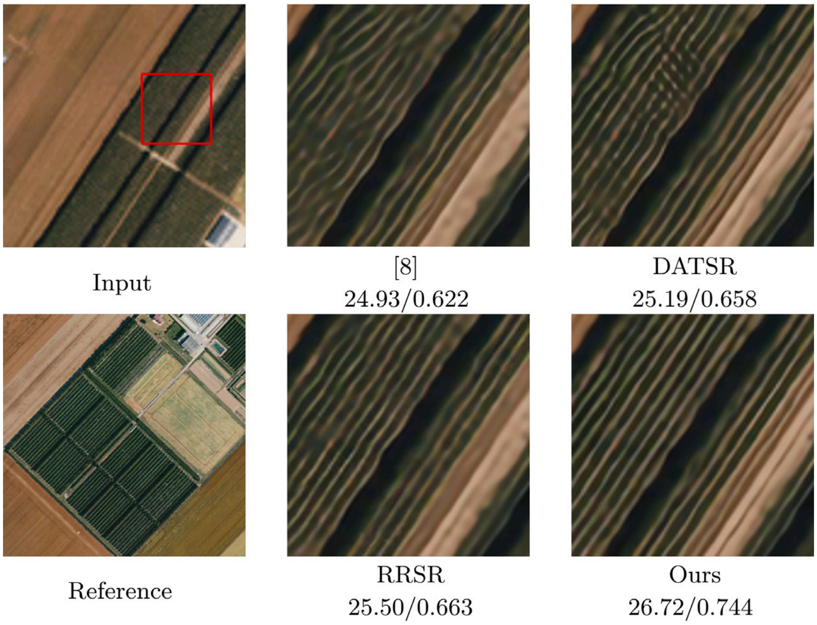


Fig. 17. In both qualitative and quantitative comparisons within the HRSCD dataset, our model demonstrates significant advantages in texture reconstruction for remote sensing images.

Data availability

Data will be made available on request.

References

[1] Y. Guo, J. Chen, J. Wang, Q. Chen, J. Cao, Z. Deng, Y. Xu, M. Tan, Closed-loop matters: Dual regression networks for single image super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 5407–5416.

[2] D. Ulyanov, A. Vedaldi, V. Lempitsky, Deep image prior, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2018, pp. 9446–9454.

[3] H. Zheng, M. Ji, H. Wang, Y. Liu, L. Fang, CrossNet: An end-to-end reference-based super resolution network using cross-scale warping, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 88–104.

[4] G. Shim, J. Park, I.S. Kweon, Robust reference-based super-resolution with similarity-aware deformable convolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 8425–8434.

[5] Z. Zhang, Z. Wang, Z. Lin, H. Qi, Image super-resolution by neural texture transfer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 7982–7991.

[6] F. Yang, H. Yang, J. Fu, H. Lu, B. Guo, Learning texture transformer network for image super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 5791–5800.

[7] L. Lu, W. Li, X. Tao, J. Lu, J. Jia, MASA-SR: Matching acceleration and spatial adaptation for reference-based image super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2021, pp. 6368–6377.

[8] Y. Jiang, K.C. Chan, X. Wang, C.C. Loy, Z. Liu, Robust reference-based super-resolution via C^2-matching, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2021, pp. 2103–2112.

[9] J. Cao, J. Liang, K. Zhang, Y. Li, Y. Zhang, W. Wang, L.V. Gool, Reference-based image super-resolution with deformable attention transformer, in: Proceedings of the European Conference on Computer Vision, ECCV, 2022, pp. 325–342.

[10] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, 2021, pp. 10012–10022.

[11] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, Y. Wei, Deformable convolutional networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2017, pp. 764–773.

[12] X. Zhu, H. Hu, S. Lin, J. Dai, Deformable convnets v2: More deformable, better results, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 9308–9316.

[13] Y. Huang, X. Zhang, Y. Fu, S. Chen, Y. Zhang, Y.-F. Wang, D. He, Task decoupled framework for reference-based super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2022, pp. 5931–5940.

[14] C. Dong, C.C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks, IEEE Trans. Pattern Anal. Mach. Intell. (2) (2015) 295–307.

[15] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2016, pp. 770–778.

[16] B. Lim, S. Son, H. Kim, S. Nah, K. Mu Lee, Enhanced deep residual networks for single image super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, 2017, pp. 136–144.

[17] N. Ahn, B. Kang, K.-A. Sohn, Fast, accurate, and lightweight super-resolution with cascading residual network, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 252–268.

[18] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, Y. Fu, Image super-resolution using very deep residual channel attention networks, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 286–301.

[19] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial networks, Commun. ACM (11) (2020) 139–144.

[20] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al., Photo-realistic single image super-resolution using a generative adversarial network, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2017, pp. 4681–4690.

[21] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, C. Change Loy, ESRGAN: Enhanced super-resolution generative adversarial networks, in: Proceedings of the European Conference on Computer Vision (ECCV) Workshops, 2018.

[22] W. Zhang, Y. Liu, C. Dong, Y. Qiao, RankSRGAN: Generative adversarial networks with ranker for image super-resolution, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, 2019, pp. 3096–3105.

[23] Q. Wang, Q. Gao, L. Wu, G. Sun, L. Jiao, Adversarial multi-path residual network for image super-resolution, IEEE Trans. Image Process. (2021) 6648–6658.

[24] X. Wang, L. Xie, C. Dong, Y. Shan, Real-ESARGAN: Training real-world blind super-resolution with pure synthetic data, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, 2021, pp. 1905–1914.

[25] Y. Yan, W. Ren, X. Hu, K. Li, H. Shen, X. Cao, SRGAT: Single image super-resolution with graph attention network, IEEE Trans. Image Process. (2021) 4905–4918.

[26] Q. Cai, J. Li, H. Li, Y.-H. Yang, F. Wu, D. Zhang, TDPN: Texture and detail-preserving network for single image super-resolution, IEEE Trans. Image Process. (2022) 2375–2389.

[27] R. Dong, L. Zhang, H. Fu, RRSGAN: Reference-based super-resolution for remote sensing image, IEEE Trans. Geosci. Remote Sens. (2022).

[28] X. Yan, W. Zhao, K. Yuan, R. Zhang, Z. Li, S. Cui, Towards content-independent multi-reference super-resolution: Adaptive pattern matching and feature aggregation, in: Proceedings of the European Conference on Computer Vision, ECCV.

[29] Y. Xie, J. Xiao, M. Sun, C. Yao, K. Huang, Feature representation matters: End-to-end learning for reference-based image super-resolution, in: European Conference on Computer Vision, ECCV.

[30] Z. Li, Z.-S. Kuang, Z.-L. Zhu, H.-P. Wang, X.-L. Shao, Wavelet-based texture reformation network for image super-resolution, IEEE Trans. Image Process. (2022) 2647–2660.

[31] Y. Kim, J. Lim, H. Cho, M. Lee, D. Lee, K.-J. Yoon, H.-J. Choi, Efficient reference-based video super-resolution (ERVSR): Single reference image is all you need, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, WACV.

[32] L. Zhang, X. Li, D. He, F. Li, Y. Wang, Z. Zhang, RRSR: Reciprocal reference-based image super-resolution with progressive feature alignment and selection, in: Proceedings of the European Conference on Computer Vision, ECCV, 2022, pp. 648–664.

[33] L. Zhang, X. Li, D. He, F. Li, E. Ding, Z. Zhang, LMR: A large-scale multi-reference dataset for reference-based super-resolution, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, CVPR, 2023, pp. 13118–13127.

[34] H. Zou, L. Xu, T. Okatani, Geometry enhanced reference-based image super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2023, pp. 6123–6132.

[35] B. Yang, G. Bender, Q.V. Le, J. Ngiam, CondConv: Conditionally parameterized convolutions for efficient inference, Adv. Neural Inf. Process. Syst. (NeurIPS) (2019).

[36] N. Ma, X. Zhang, J. Huang, J. Sun, WeightNet: Revisiting the design space of weight networks, in: Proceedings of the European Conference on Computer Vision, ECCV, 2020, pp. 776–792.

[37] F. Wu, A. Fan, A. Baevski, Y.N. Dauphin, M. Auli, Pay less attention with lightweight and dynamic convolutions, 2019, arXiv preprint arXiv:1901.10430.

[38] J. Zhou, V. Jampani, Z. Pi, Q. Liu, M.-H. Yang, Decoupled dynamic filter networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2021, pp. 6647–6656.

[39] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2018, pp. 7132–7141.

[40] A.B. Molini, D. Valsesia, G. Fracastoro, E. Magli, DeepSUM: Deep neural network for super-resolution of unregistered multitemporal images, IEEE Trans. Geosci. Remote Sens. (5) (2019) 3644–3656.

[41] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, J. Sun, Meta-SR: A magnification-arbitrary network for super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 1575–1584.

[42] Y. Shi, H. Zhong, Z. Yang, X. Yang, L. Lin, DDet: Dual-path dynamic enhancement network for real-world image super-resolution, IEEE Signal Process. Lett. (2020) 481–485.

[43] J. Lee, H. Son, J. Rim, S. Cho, S. Lee, Iterative filter adaptive network for single image defocus deblurring, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2021, pp. 2034–2042.

[44] R. Ma, S. Li, B. Zhang, Z. Li, Generative adaptive convolutions for real-world noisy image denoising, in: Proceedings of the AAAI Conference on Artificial Intelligence, (2) 2022, pp. 1935–1943.

[45] H. Zheng, Z. Lin, J. Lu, S. Cohen, E. Shechtman, C. Barnes, J. Zhang, N. Xu, S. Amirghodsi, J. Luo, Image inpainting with cascaded modulation GAN and object-aware training, in: European Conference on Computer Vision, ECCV.

[46] T. Park, M.-Y. Liu, T.-C. Wang, J.-Y. Zhu, Semantic image synthesis with spatially-adaptive normalization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR.

[47] P. Chandran, G. Zoss, P. Gotardo, M. Gross, D. Bradley, Adaptive convolutions for structure-aware style transfer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2021, pp. 7972–7981.

[48] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2017, pp. 4700–4708.

[49] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Proceedings, Part III 18, 2015, pp. 234–241.

[50] S. Zagoruyko, N. Komodakis, Wide residual networks, 2016, arXiv preprint arXiv:1605.07146.

[51] R.K. Srivastava, K. Greff, J. Schmidhuber, Highway networks, 2015, arXiv preprint arXiv:1505.00387.

[52] T. Tong, G. Li, X. Liu, Q. Gao, Image super-resolution using dense skip connections, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, 2017, pp. 4799–4807.

[53] Y. Mei, L. Li, Z. Li, F. Li, Learning-based scalable image compression with latent-feature reuse and prediction, IEEE Trans. Multimed. (2021) 4143–4157.

[54] Z. Wang, X. Cun, J. Bao, W. Zhou, J. Liu, H.U. Li, A general u-shaped transformer for image restoration, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2022, pp. 19–24.

[55] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2014, arXiv preprint arXiv:1409.1556.

[56] W. Wang, J. Dai, Z. Chen, Z. Huang, Z. Li, X. Zhu, X. Hu, T. Lu, L. Lu, H. Li, et al., Internimage: Exploring large-scale vision foundation models with deformable convolutions, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR.

[57] J. Liu, J. Tang, G. Wu, Residual feature distillation network for lightweight image super-resolution, in: Proceedings of the European Conference on Computer Vision, ECCV, 2020, pp. 41–55.

[58] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: International Conference on Machine Learning, ICML, 2015, pp. 448–456.

[59] Z. Li, Y. Liu, X. Chen, H. Cai, J. Gu, Y. Qiao, C. Dong, Blueprint separable residual network for efficient image super-resolution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2022, pp. 833–843.

[60] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in: Proceedings of the European Conference on Computer Vision, ECCV, 2016, pp. 694–711.

[61] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, A.C. Courville, Improved training of wasserstein gans, Adv. Neural Inf. Process. Syst. (NeurIPS) (2017).

[62] J.-B. Huang, A. Singh, N. Ahuja, Single image super-resolution from transformed self-exemplars, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2015, pp. 5197–5206.

[63] Y. Matsui, K. Ito, Y. Aramaki, A. Fujimoto, T. Ogawa, T. Yamasaki, K. Aizawa, Sketch-based manga retrieval using manga109 dataset, Multimedia Tools Appl. (2017) 21811–21838.

[64] L. Sun, J. Hays, Super-resolution from internet-scale scene matching, in: 2012 IEEE International Conference on Computational Photography, ICCP, 2012, pp. 1–12.

[65] M.S. Sajjadi, B. Scholkopf, M. Hirsch, EnhanceNet: Single image super-resolution through automated texture synthesis, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV, 2017, pp. 4491–4500.

[66] R. Zhang, P. Isola, A.A. Efros, E. Shechtman, O. Wang, The unreasonable effectiveness of deep features as a perceptual metric, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2018, pp. 586–595.

[67] R. Dong, S. Yuan, B. Luo, M. Chen, J. Zhang, L. Zhang, W. Li, J. Zheng, H. Fu, Building bridges across spatial and temporal resolutions: Reference-based super-resolution via change priors and conditional diffusion model, 2024, arXiv preprint arXiv:2403.17460.

[68] R.C. Daudt, B. Le Saux, A. Boulch, Y. Gousseau, Multitask learning for large-scale semantic change detection, Comput. Vis. Image Underst. 187 (2019) 102783.