Feature Representation Matters: End-to-End Learning for Reference-based Image Super-resolution

Yanchun Xie[1], Jimin Xiao[1,⋆], Mingjie Sun[1], Chao Yao[2], and Kaizhu Huang[1,3]

[1] School of Advanced Technology, Xi'an Jiaotong-Liverpool University, Suzhou, China
[2] University of Science and Technology Beijing, Beijing, China
[3] Alibaba-Zhejiang University Joint Institute of Frontier Technologies, Hangzhou, China

Abstract. In this paper, we are aiming for a general reference-based super-resolution setting: it does not require the low-resolution image and the high-resolution reference image to be well aligned or with a similar texture. Instead, we only intend to transfer the relevant textures from reference images to the output super-resolution image. To this end, we engaged neural texture transfer to swap texture features between the low-resolution image and the high-resolution reference image. We identified the importance of designing a super-resolution task-specific features rather than classification oriented features for neural texture transfer, making the feature extractor more compatible with the image synthesis task. We develop an end-to-end training framework for the reference-based super-resolution task, where the feature encoding network prior to matching and swapping is jointly trained with the image synthesis network. We also discovered that learning the high-frequency residual is an effective way for the reference-based super-resolution task. Without bells and whistles, the proposed method E2ENT^2 achieved better performance than state-of-the method (i.e., SRNTT with five loss functions) with only two basic loss functions. Extensive experimental results on several datasets demonstrate that the proposed method E2ENT^2 can achieve superior performance to existing best models both quantitatively and qualitatively.

Keywords: super-resolution; reference-based; feature matching; feature swapping; CUFED5; Flickr1024

1 Introduction

Image super-resolution (SR) is an essential task in computer vision, aiming to transfer low-resolution (LR) images to their high-resolution (HR) counterparts. SR remains to be a long-standing and ill-posed problem due to the non-unique mapping between high and low-resolution samples. A single low resolution (LR)

⋆ Corresponding author. Email: jimin.xiao@xjtlu.edu.cn

image could correspond to multiple high resolution (HR) images. A large number of deep SR models have been proposed to solve this problem in recent years [3, 10, 7, 11, 13, 1]. However, in case of a large upsampling factor, recovering an HR image requires to provide sufficient information to fill the missing contents in the LR image.

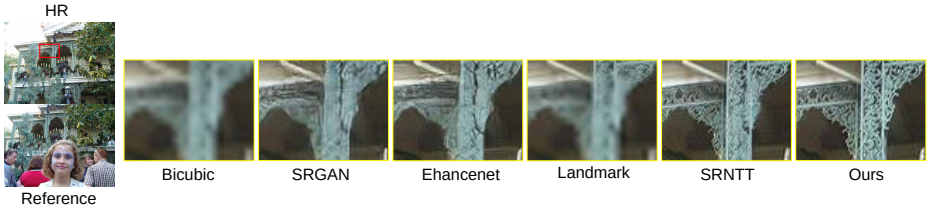

Fig. 1. Left: High resulotion image (up) and reference (bottom). Right: zoomed results of different SR algorithms, including SRGAN[11], Ehancenet[13], Landmark[22], SRNTT[24], and ours. Our end-to-end learning method produces the best result.

Reference-based super-resolution (RefSR) is a new SR branch in recent years, which has been proven to be effective in recovering lost high-frequency details in the LR images [16, 22, 27, 28]. These reference-based methods generally require reference images to have similar content with the LR image or with proper alignment. For example, prior work [28] focuses on RefSR for light field images where the LR image and the HR reference image are very similar as they have relatively small disparities. It estimates the transformation by optical flow and uses the multi-scale warping technique for feature alignment. For these RefSR methods, if the reference images do not possess relevant textures with the LR image, their performance would significantly degrade and even be worse than signal image SR methods.

In this paper, we are aiming for a more general RefSR setting: it does not require the LR image and the HR reference image to be well aligned or with a similar texture. Instead, we only intend to transfer the relevant texture from reference images to the output SR image. Ideally, a robust RefSR algorithm should outperform single image super-resolution (SISR) when a better reference image is provided, whilst achieving comparable performance when reference images do not possess relevant texture at all.

Based on this goal, SRNTT [24] proposes a neural texture transfer approach that breaks the limitation of reference images. In SRNTT, local texture matching is conducted in the feature space, and the matched textures are transferred to the synthesized high-resolution image through a deep neural network. However, there are three main issues for SRNTT: (1) the features used in this image synthesis task are extracted from a VGG net. Initially designed for image classification, VGG may not lead to the best features for SR. (2) WIth the fixed VGG net, SRNTT does not take advantage of the end-to-end learning in the SR task.

(3) VGG features in shallow layers involve a high computational and enormous memory cost, making it time-consuming to process images with large size.

In this paper, we argue that that the matching feature does matter for neural texture transfer in RefSR. Thus, we analyze the feature extractor in the RefSR method and propose to use features designed for SR (i.e., SRGAN [11]) instead of features designed for classification (VGG). Such features, on the other hand, are more compatible with the image synthesis network where the adversarial loss is used [5]. Secondly, Distinctive with previous RefSR methods, the whole neural network, including the feature representation part, is able to be trained in an end-to-end manner. Visual quality comparisons between our approach and other state-of-the-art methods are shown in Fig.1.

Our contributions are summarized as follows:

- We identified the importance of using a task-specific feature extractor for matching and swapping in RefSR, and proposed to use features designed for SR (i.e., SRGAN [11]) instead of features designed for classification (VGG), making the feature extractor more compatible with the image synthesis task.
- We designed an end-to-end training framework for the RefSR task, where the feature extraction network for matching and swapping is jointly trained with the image synthesis network. We also discovered that learning the high-frequency residual is an effective and efficient way for the reference-based super-resolution task. Without bells and whistles, we achieved better performance than the state-of-the method (i.e., SRNTT [24] with five loss functions) with only two basic loss functions.
- We evaluated our method in RefSR datasets, achieving the new quantitatively results (24.01dB for PSNR, 0.705 for SSIM) in the CUFED5 dataset. Qualitative results also demonstrate the superiority of our method.

2 Related Work

2.1 Image Super-resolution

Deep learning based methods have been applied to image SR in recent years [3, 9, 10, 12, 23], and significant progress have been obtained due to its powerful feature representation ability. These methods learn an end-to-end mapping from LR to HR directly with a mean squared loss function, treating the super-resolution as a regression problem. SRGAN [11] considers both perceptual similarity loss and adversarial loss for super-resolution. The perceptual similarity is obtained by computing the feature distance extracted from the VGG middle layer. The adversarial loss enables us to generate realistic visual results for humans by using a discriminator to distinguish between real HR images and super-resolved images generated from generators.

The super-resolution performance has been boosted with deep features and residual learning. For example, Dong et al. first introduced a three-layer convolutional network SRCNN [3] for image super-resolution. After that, Kim et al. reformed the problem based on residual learning and proposed VDSR [9] and

DRCN [10] with deeper layers. Lim et al. proposed two very deep multi-scale super-resolution networks EDSR and MDSR [12] by modifying residual units and further improve the performance. Zhang et al. [23] proposed a residual in residual structure to allows focusing on learning high-frequency information and a channel attention mechanism to rescale channel-wise features by considering inter-dependencies among channels adaptively.

2.2 Reference-based Super-resolution

Different from single image super-resolution with the only low-resolution image provided, RefSR methods utilize additional images that have more texture information to assist the recovery process. Generally, the reference images contain similar objects, scenes, or texture with the low-resolution image. The reference images can be obtained from different frames in a video sequence, different viewpoints in light field images or multiview videos, or by web retrieval. Many works study the reference-based super-resolution by extra examples or similar scenes from web [14, 17, 15]. Other works [26, 21, 27, 28] use reference images from different viewpoints to enhance light field images. These works mostly build the mapping from LR to HR patches, and fuse the HR patches at the pixel level or using a shallow model. To overcome inter-patch misalignment and the grid effect, CrossNet[28] uses optical flow to spatially align the reference feature map with the LR feature map and then aggregates them into SR images. SRNTT [24] further proposes a neural texture transfer approach to improve the matching and fusing ability. In their approach, VGG features with semantically relevant textures from reference images are transferred to the LR image.

Unlike the flow and wrapping based approach [28], our method could further handle the images with much larger disparities than that in light field data. Different from the existing neural texture transfer approach [24], our texture matching and swapping part is end-to-end trainable.

3 Our Method

In this section, our proposed method, namely End-to-End learning for Neural Texture Transfer (E2ENT^2), will be introduced in detail. We first present the network framework of our proposed E2ENT^2, as shown in Fig.2, which consists of 3 key blocks, including (1) a feature encoding module which extracts features from the LR input and reference images; (2) a newly designed match and swap (MS) module which identifies similar LR-HR feature pairs and conducts feature swapping, where gradients can back-propagate through it to enable end-to-end learning; (3) an image synthesis module which fuses the LR image feature and swapped feature, and outputs the SR image.

3.1 Notations

The input of our network includes a LR input image I_{in}, an HR reference image I_{ref} and a corresponding LR reference image $I^{\downarrow}_{ref}$. $I^{\downarrow}_{ref}$ is the down-sampled

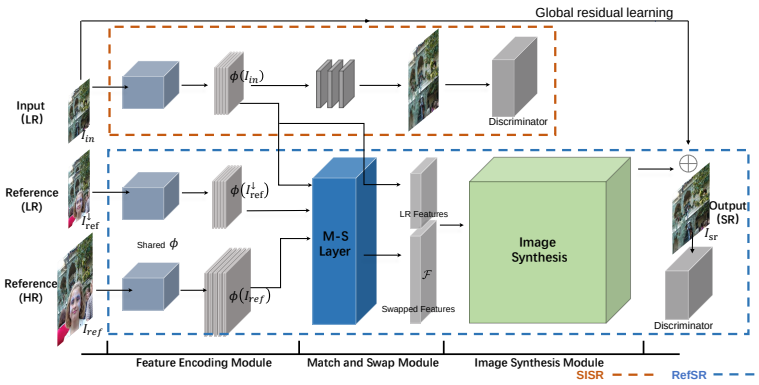

Fig. 2. The framework of our proposed network. The network consists of three main modules: feature encoding module, match and swap module, and image synthesis module. The network takes the LR image, HR reference image as input, and outputs the super-resolved image.

version of the HR reference image I_{ref}. I_{in} is with size $W_{in} \times H_{in}$; I_{ref} is with size $W_{ref} \times H_{ref}$, which does not need to be the same size as I_{in}, and $I^{\downarrow}_{ref}$ is with size $\frac{W_{ref}}{r} \times \frac{H_{ref}}{r}$, with r being the super-resolution ratio.

After the feature encoding module, we get feature maps $\phi(I_{in})$, $\phi(I_{ref})$ and $\phi(I^{\downarrow}_{ref})$ for I_{in}, I_{ref} and $I^{\downarrow}_{ref}$, respectively. The feature map size is $W_{in} \times H_{in}$ for $\phi(I_{in})$, $W_{ref} \times H_{ref}$ for $\phi(I_{ref})$, and $\frac{W_{ref}}{r} \times \frac{H_{ref}}{r}$ for $\phi(I_{ref})$. In other words, the feature map shares the same width and height with the image, so that could minimize the loss of details.

Feature maps $\phi(I_{in})$, $\phi(I^{\downarrow}_{ref})$ and $\phi(I_{ref})$ are fed to the match and swap module ψ, and a new swapped feature map $\mathcal{F}$ is obtained,

$$\mathcal{F} = \psi(\phi(I_{in}), \phi(I^{\downarrow}_{ref}), \phi(I_{ref})), \tag{1}$$

where the size of $\mathcal{F}$ is $rW_{in} \times rH_{in}$.

Finally, the swapped feature $\mathcal{F}$ together with the LR feature $\phi(I_{in})$ are fed into the image synthesis module ζ to generate the super-resolution image I_{sr}, as

$$I_{sr} = \zeta(\mathcal{F}, \phi(I_{in})), \tag{2}$$

where the size of I_{sr} is $rW_{in} \times rH_{in}$.

3.2 Feature Encoding Module

Single image super-resolution benefits a lot from skip-connections [9, 10, 12, 23], and various deep learning models have achieved state-of-the-art performance. Thus, we propose to utilize the residual learning in the SR feature encoding

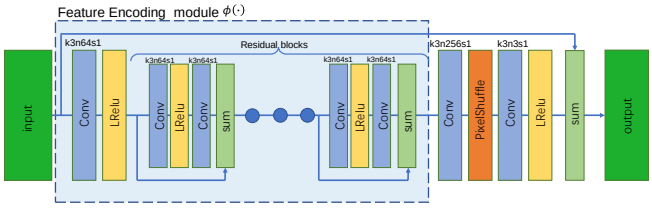

Fig. 3. The structure of our single-image super-resolution (SISR) branch with the residual connection. The network consists of several residual blocks for feature encoding. The feature encoding module is marked within the dashed line.

module to improve the accuracy of feature representation for the reference-based super-resolution task.

Our proposed RefSR network shares the same feature encoding module $\phi(\cdot)$ in the SISR branch to produce features for I_{in}, I_{ref} and $I_{ref}^{\downarrow}$. The SISR branch has a deep residual-based structure without the BN layer, as shown in Fig.3. The SISR branch is composed of stacked residual blocks with 3×3 Conv kernels and followed by pixelshuffle layers for upsampling. The skip connections allow the network to focus on informative features rather than the LR features. After the feature encoding module, we can get $\phi(I_{in})$, $\phi(I_{ref})$ and $\phi(I_{ref}^{\downarrow})$.

In addition to being used in the RefSR branch, $\phi(I_{in})$ is also passed to the rest of the SISR branch to complete a SISR task, which ensures feature consistency between the two standalone SR tasks. Meanwhile, introducing a shared trainable feature encoding module in both SISR and RefSR can generate discriminative features for the match and swap module due to end-to-end learning.

To further enhance the subjective visual quality of the SR image, we also adopt a discriminator for adversarial learning in both SISR and RefSR branches.

3.3 Match and Swap Module

To transfer the semantically relevant texture from reference images to the output SR image, we adopt a patch-based feature match and swap module. As shown in Fig.4, the match and swap module takes the feature maps obtained in the encoding stage as input, including $\phi(I_{in})$, $\phi(I_{ref})$ and $\phi(I_{ref}^{\downarrow})$. This module outputs a fused feature map $\mathcal{F}$.

Forward Pass. Our proposed matching process is conducted at patch level, which is a 3×3 feature block. Firstly, we crop $\phi(I_{in})$, $\phi(I_{ref}^{\downarrow})$ and $\phi(I_{ref})$ into 3×3, 3×3 and $3r \times 3r$ patches with stride 1, 1 and r, respectively. These patches are indexed based on the horizontal and vertical position. Matching similarity is computed between patches in $\phi(I_{in})$ and $\phi(I_{ref}^{\downarrow})$.

To recover the missing details as much as possible, in the feature matching process, for each LR feature patch in $\phi(I_{in})$, we need to search for the most similar feature patch in $\phi(I_{ref}^{\downarrow})$, and the corresponding feature patch in $\phi(I_{ref})$ will be used to replace the original patch.

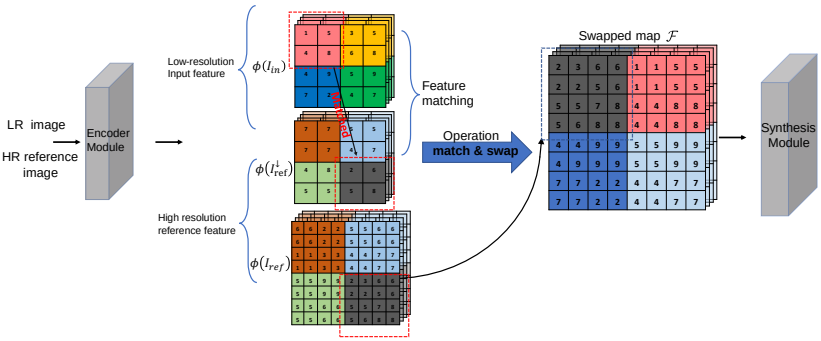

Fig. 4. Illustration of the forward pass in the match and swap module. Feature patch matching are conducted based on the feature similarity between $\phi(I_{in})$, $\phi(I^{\downarrow}_{ref})$. The corresponding matched HR reference feature patches replace the LR features, and finally a swapped feature $\mathcal{F}$ is produced.

Computation of patch similarity is efficiently implemented as convolution operations. The matching result is recorded in a 3-dimensional similarity map $\mathcal{S}$, with $\mathcal{S}_i(x, y)$ denoting the similarity between the patch centered at the location (x, y) in $\phi(I_{in})$ and the i-th reference patch in $\phi(I^{\downarrow}_{ref})$. Computation of $\mathcal{S}_i$ can be efficiently implemented as a set of convolution operations over all patches in $\phi(I_{in})$ with a kernel corresponding to reference feature patch i:

$$\mathcal{S}_i = \phi\left(I_{in}\right) * \frac{\mathcal{P}_i\left(\phi\left(I^{\downarrow}_{ref}\right)\right)}{\left\|\mathcal{P}_i\left(\phi\left(I^{\downarrow}_{ref}\right)\right)\right\|}, \tag{3}$$

where $\mathcal{P}_i(\cdot)$ denotes to sample the i-th patch from a feature map, $*$ is a 2D convolution operation, and $\|\cdot\|$ is used to get the feature length (L1). Note that $\mathcal{S}_i$ is a 2-dimensional map.

After the feature matching, we can obtain a swapped feature map $\mathcal{F}$ based on the 3D similarity map $\mathcal{S}$. Each patch in $\mathcal{F}$ centered at (x, y) is defined as:

$$\mathcal{F}^{p}_{(x,y)} = \mathcal{P}_{i^*}\left(\phi\left(I_{ref}\right)\right), i^* = \arg\max_i \mathcal{S}_i(x, y), \tag{4}$$

where i^* is the patch index for the most similar one in the reference feature. $\mathcal{P}_{i^*}(\cdot)$ denotes to sample the i^*-th patch from a feature map. Note that the patch size of $\mathcal{P}_{i^*}\left(\phi\left(I_{ref}\right)\right)$ is r^2 times that of $\mathcal{P}_{i^*}\left(\phi\left(I^{\downarrow}_{ref}\right)\right)$. Therefore, after swapping, the feature size of $\mathcal{F}$ is r^2 times that of $\phi(I_{in})$.

In the forward pass, we use $\mathcal{K}_{(x,y)}$ to record the number of times that the reference patch centered at (x, y) in $\phi(I_{ref})$ is selected for swapping, and use $\mathcal{Q}_{(x,y)}$ to record a list of patch center coordinates for all the LR patches in $\phi(I_{in})$ that matches with the reference patch centered at (x, y) in the matching process. $\mathcal{K}_{(x,y)}$ and $\mathcal{Q}_{(x,y)}$ will be used in the gradient backpropagation process.

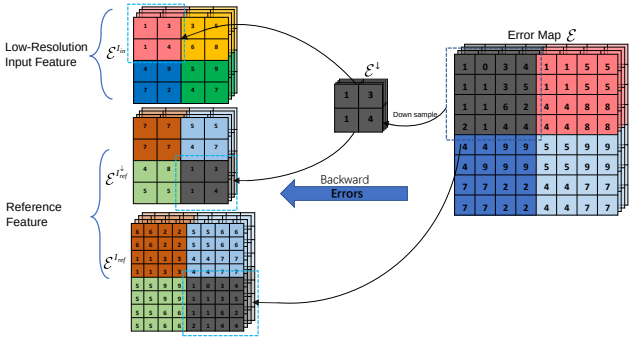

Fig. 5. Illustration of the error backward propagation in the match and swap module. Error gathered at $\mathcal{F}$ from the loss layer backward propagates through the match and swap module to the image encoding module. In this figure, we assume $\alpha 1 = \alpha 2 = \alpha 3 = 1$ for simplicity.

We conduct the feature matching at low-resolution (using $\phi(I_{in})$ and $\phi(I^{\downarrow}_{ref})$) to boost the matching speed for fast training. Traditional feature matching methods [4, 24] use a bicubic up-sampling strategy on the LR image to get an up-sampled image that shares the same spatial size as an HR image. However, such operation brings exponential computation in the feature matching process, especially when the image size is large.

Backward Pass. To have an end-to-end training, we design a mechanism to enable the gradients to back-propagate through the match and swap module, from the image synthesis module to the feature encoding module, as shown in Fig.5.

The error term $\mathcal{E} = \partial\mathcal{J}/\partial\mathcal{F}$ for $\mathcal{F}$ can be calculated from the loss layer, with $\mathcal{J}$ being the loss function. $\mathcal{E}$ is with the same size as the swapped map $\mathcal{F}$. Notice that the argmax function in Eq.(4) is non-differentiable, a new mechanism to back-propagate $\mathcal{E}$ to the feature encoding module is needed.

As demonstrated in Fig.4, features $\phi(I_{in})$, $\phi(I_{ref})$ and $\phi(I^{\downarrow}_{ref})$ all affect the swapped map $\mathcal{F}$. We define the error term for $\phi(I_{in})$, $\phi(I_{ref})$ and $\phi(I^{\downarrow}_{ref})$ are $\mathcal{E}^{I_{in}}$, $\mathcal{E}^{I_{ref}}$ and $\mathcal{E}^{I^{\downarrow}_{ref}}$, respectively. Since the feature matching location information, $\mathcal{K}_{(x,y)}$ and $\mathcal{Q}_{(x,y)}$, are recorded in the forward process, for each matching patch centered at (x, y), we have their error terms:

$$\begin{aligned}
\mathcal{E}^{I_{in}}_{(x,y)} &= \alpha_1 \mathcal{E}^{\downarrow}_{(x,y)}, \\
\mathcal{E}^{I^{\downarrow}_{ref}}_{(x,y)} &= \alpha_2 \sum_{j=1}^{\mathcal{K}_{(x,y)}} \mathcal{E}^{\downarrow}_{\mathcal{Q}^j_{(x,y)}}, \\
\mathcal{E}^{I_{ref}}_{(x,y)} &= \alpha_3 \sum_{j=1}^{\mathcal{K}_{(x,y)}} \mathcal{E}_{\mathcal{Q}^j_{(x,y)}},
\end{aligned} \tag{5}$$

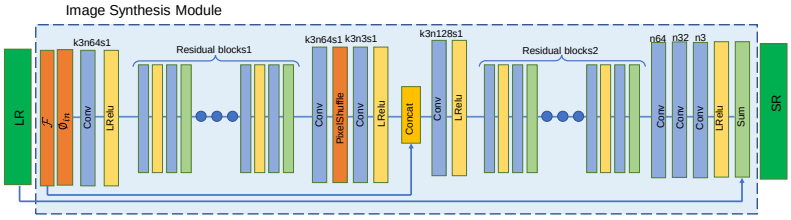

Fig. 6. The structure of the feature transfer and image synthesis network. The network consists of several residual blocks for feature decoding. The image synthesis module is marked with the dashed line.

where $\mathcal{E}^{\downarrow}$ is the corresponding downsampled error term for $\mathcal{E}$; $\mathcal{E}^{\downarrow}_{(x,y)}$, $\mathcal{E}^{\downarrow}_{\mathcal{Q}^{j}_{(x,y)}}$ are downsampled error term for patch centered at (x, y) and $\mathcal{Q}^{j}_{(x,y)}$, respectively; $\mathcal{E}_{\mathcal{Q}^{j}_{(x,y)}}$ is the error term for patch centered at $\mathcal{Q}^{j}_{(x,y)}$. α_1, α_2 and α_3 are different weighting factors. Considering that each reference feature patch could have multiple matches with patches in $\phi(I_{in})$, the corresponding error terms are accumulated multiple times for $\mathcal{E}^{I^{\downarrow}_{ref}}$ and $\mathcal{E}^{I_{ref}}$.

We construct the whole error map $\mathcal{E}^{I_{in}}$, $\mathcal{E}^{I^{\downarrow}_{ref}}$ and $\mathcal{E}^{I_{ref}}$ in the feature encoding module by accumulating error terms for all the patches along with their coordinates. For the overlapped regions covered by multiple patches, the average error value is used.

Finally, the error map $\mathcal{E}^{I_{in}}$, $\mathcal{E}^{I^{\downarrow}_{ref}}$ and $\mathcal{E}^{I_{ref}}$ are used for the parameter update in the convolution layers of the feature encoding module:

$$\frac{\partial \mathcal{J}(\mathcal{W})}{\partial \mathcal{W}} = \mathcal{E}^{I_{in}} \frac{\partial \phi(I_{in})}{\partial \mathcal{W}} + \mathcal{E}^{I^{\downarrow}_{ref}} \frac{\partial \phi(I^{\downarrow}_{ref})}{\partial \mathcal{W}} + \mathcal{E}^{I_{ref}} \frac{\partial \phi(I_{ref})}{\partial \mathcal{W}}, \tag{6}$$

where $\mathcal{W}$ is the parameter set, and η is the update rate.

3.4 Image Synthesis Module

In the image synthesis module, the LR image I_{in}, its features $\phi(I_{in})$, and the swapped feature map $\mathcal{F}$ are used to fuse and synthesize the SR image with residual learning. The swapped feature $\mathcal{F}$ contains HR textures to recover the details.

Similar to the structure in our feature encoding module, we also utilize the stacked residual blocks to fuse the high-frequency features to the SR image. As shown in Fig.6, the first set of residual blocks on the left mainly focuses on upsampling the LR features $\phi(I_{in})$ for the next stage, while the second set of residual blocks focuses on the information fusion between the two kinds of features. The features at the concatenation operation are with the same feature size and they are concatenated at the channel dimension.

The final output super-resolution image I_{sr} can be defined as:

$$I_{sr} = I^{\uparrow}_{in} + Res2([\mathcal{F} \oplus Res1(\phi(I_{in}))]), \tag{7}$$

where $I_{in}^{\uparrow}$ is a bilinear interpolated upsampled input, $Res1$ and $Res2$ represent the left and right residual connection blocks, respectively, and $\oplus$ is the concatenation operation. The detailed structure of the image synthesis network is shown in Fig.6. Note that feature $\mathcal{F}$ and $\phi(I_{in})$ used for image synthesis are all obtained from our SR task, instead of coming from a classification model, *e.g.*, VGG [24].

The skip-connection between the LR image and the SR image could increase the image synthesis stability by making the network focus more on the high-frequency details during the training.

To further enhance the subjective visual quality of the SR image, we also adopt discriminators for adversarial learning in both SISR and RefSR branches.

3.5 Loss Function

Reconstruction Loss. Generally, the mean squared error (MSE) loss function is used in the SR task to achieve high PSNR. While in our work, we adopt the L1 norm to precisely measure the pixel difference. The L1 norm can sharpen the super-resolution image compared to that of MSE loss [25], though its PSNR is slightly lower than that of MSE loss.

$$\mathcal{L}_{rec} = \left\| I^{SR} - GT \right\|_1 . \tag{8}$$

Adversarial Loss. We introduce adversarial learning in our RefSR method, the loss function is define as:

$$\mathcal{L}_D = -\mathbb{E}_{x_{real}} \left[\log \left(D \left(x_{real}, x_{fake} \right) \right) \right] - \mathbb{E}_{x_{fake}} \left[\log \left(1 - D \left(x_{fake}, x_{real} \right) \right) \right], \tag{9}$$

where D is an relativistic average discriminator[8]. Respectively, x_{real} and x_{fake} are the groundtruth and generated output of our network.

$$\mathcal{L}_G = -\mathbb{E}_{x_{real}} \left[\log \left(1 - D \left(x_{real}, x_{fake} \right) \right) \right] - \mathbb{E}_{x_{fake}} \left[\log \left(D \left(x_{fake}, x_{real} \right) \right) \right], \tag{10}$$

It is observed using this adversarial loss [8] can make our training faster and more stable compared to a standard GAN objective. We also empirically conclude that the generated results possess higher perceptual quality than that of a standard GAN objective.

4 Experiments

4.1 Implementation Details

The proposed method is trained on CUFED[20], consisting of around 100,000 images. During training, a GAN-based SISR is firstly pre-trained on CUFED. Then followed by the end-to-end training of both SISR and RefSR. Specifically, each image of CUFED will be cropped within random bounding boxes twice, to generate two different patches with similar content. The crops image pair (input and reference) will be used for end-to-end training. Adam optimizer is used with

Table 1. A quantitative comparison of our approach with other SR methods on CUFED5 and SUN Hays dataset. The used super-resolution ratio is 4×4. PSNR and SSIM are used as the evaluation metrics.

	CUFED5[24]		**SUN Hays[15]**	
Method	**PSNR**	**SSIM**	**PSNR**	**SSIM**
Bicubic	22.64	0.646	27.25	0.742
DRCN[10]	23.56	0.692	-	-
EnhanceNet[13]	22.58	0.651	25.46	0.669
SRGAN[11]	22.93	0.656	26.42	0.696
Ours-SISR	23.75	0.697	26.72	0.712
Landmark[22]	23.23	0.674	-	-
SRNTT[24]	23.64	0.684	26.79	0.727
E2ENT^2-MSE(ours)	**24.24**	**0.724**	**28.50**	**0.789**
E2ENT^2(ours)	24.01	0.705	28.13	0.765

Table 2. A quantitative comparison of our approach with other SR methods on Flickr1024 dataset. The used super-resolution ratio is 4×4. PSNR and SSIM are used as the evaluation metrics.

Method	**Flickr1024 Test Set[19]**	
	PSNR	**SSIM**
SteroSR[6]	21.77	0.617
PASSRnet[18]	21.31	0.600
SRGAN[11]	21.67	0.567
SRNTT[24]	22.02	0.637
E2ENT^2(ours)	**22.89**	**0.680**

a learning rate of 1e-4 throughout the training. The weights for L_{rec},L_{adv}, is 1e-2 and 1e-5, respectively. The number of residual blocks is 16 for both encoder and decoder. The network is trained with the CUFED dataset for 20 epochs with two basic losses. In all our designated experiments, no augmentation other than image translation is applied.

The proposed method is evaluated on the datasets CUFED5[24], SUN hays[15] and Flickr1024[19], containing 126, 80 and 112 image pairs respectively. Each image pair contains one input image and one reference image for the evaluation of reference-based SR methods. To evaluate single-image SR methods, all images in these datasets are viewed as individual images. Moreover, compared with CUFED5 and SUN hays datasets, Flickr1024 is a stereo image dataset with higher resolution and similarity, and we use its testset for evaluation. The evaluation relies on two common metrics, including PSNR and SSIM.

4.2 Evaluations

The proposed method is compared with some related methods, which are classified into two groups. Methods in the first group are designed for single-image SR, including Bicubic, DRCN [10], EnhanceNet [13] and SRGAN [11]. Methods in the second groups are designed for reference-based SR, including Landmark [22], SRNTT [24], SteroSR [6] and PASSRnet [18]. The quantitative results are summarized in Table 1 and Table 2.

For the evaluation of reference-based SR methods, the proposed method also outperforms other methods and boosts PSNR by 0.6 dB on CUFED5 and 1.71 dB on SUN Hays against the previous state-of-the-art method (SRNTT). The SSIM gain over SRNTT is also substantial, being 0.040 and 0.062 for CUFED5 and SUN Hays, respectively. E2ENT^2-MSE denotes that the MSE loss is used to replace the L1 reconstruction loss. When evaluated on a stereo dataset (Flickr1024), as shown in Table 2, where the reference images are highly relevant, the proposed method shows a great advantage over the SISR method (SRGAN) and other RefSR based methods, demonstrating its robustness under different similarity levels between LR input images and HR reference images.

Some visualization comparisons are reported in Fig.7, including indoor objects, buildings, and natural scenes. For a clear illustration, some image patches are zoomed in to fully demonstrate the exquisite textures and details of the SR images generated by the proposed method. A user study is conducted, seven algorithms, including both single/reference-based image super-resolution results, are given to the respondents. The statistical results are shown in Fig. 8, compared with single image super-resolution methods, respondents favor the results of reference-based methods more.

4.3 Ablation study

Impact of feature encoding module. The first ablation study is about the impact of different feature encoding methods, with the comparisons reported in Table 3.

To do this, firstly, the SISR branch is pre-trained on the SR dataset, and the encoder of this pre-trained SISR will be utilized later. Then, SISR and RefSR are trained in an end-to-end way on the CUFED dataset, obtaining the feature encoding method E2ENT^2 in Table 3. Secondly, we replace the feature encoding module of E2ENT^2 with VGG (pre-trained on ImageNet[2]) and train the remaining network to obtain the model Feature-VGG. Similarly, by replacing the feature encoding module of E2ENT^2 with the encoder of the pre-trained SISR in the first step, we train the model Feature-preSISR. As can be observed from Table 3, E2ENT^2 obtains the highest PSNR and SSIM among all settings. The results demonstrate the effectiveness of the proposed trainable feature encoding module.

Besides, we calculate the feature distance (L1) between the swapped feature map $\mathcal{F}$ and that of the ground truth HR image without the match and swap

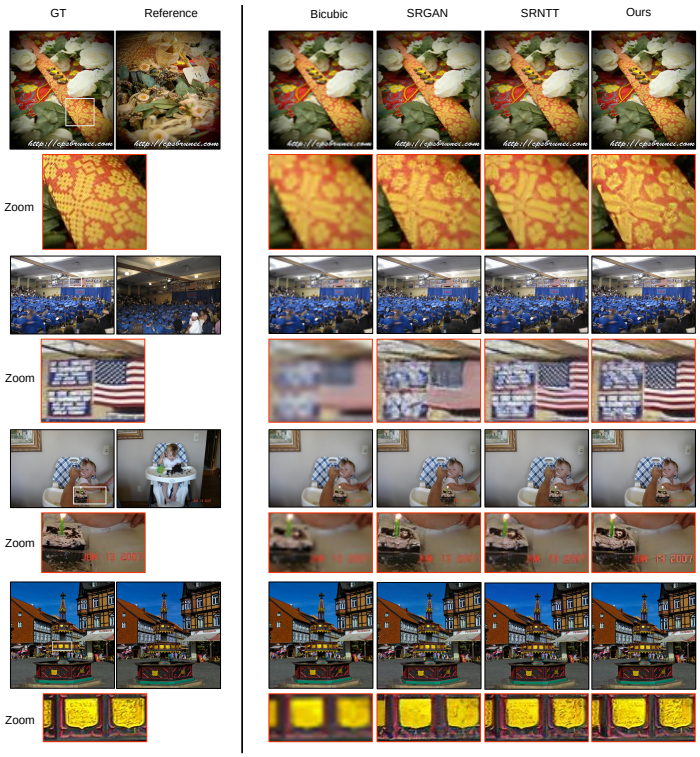

Fig. 7. Visualizations of generated images with different SR methods on CUFED5 (first 4 rows) and Flickr1024 datasets (last rows). Best viewed in color, and zoom-in mode.

module. The small feature distance of E2ENT^2 indicates that the feature of E2ENT^2 is closer to the feature of the HR ground truth image than others.

Table 3. A comparison study of three different feature coding methods. The used super-resolution ratio is 4×4. PSNR and SSIM are used as the evaluation metrics.

Feature Type	PSNR	SSIM	Feature Distance
Feature-VGG	22.85	0.647	106.77
Feature-preSISR	23.46	0.678	58.94
E2ENT^2(ours)	24.01	0.705	25.77

Impact of gradient allocation. The second ablation study is about the influence of gradient allocation, which is controlled through variable weights (α_1, α_2, α_3) in Eq.(5). As can be observed from Table 4, parameter set (α_1, α_2, α_3) = (0.25, 0.25, 0.50) outperforms (1, 0, 0), (0, 1, 0) and (0, 0, 1), indicating that only

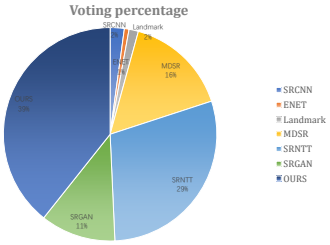

Fig. 8. The user study results. Our method is compared with different SR algorithms, more respondents favor our E2ENT results than that of SRNTT.

to consider the gradient for one feature in $\{\phi(I_{in}), \phi(I^{\downarrow}_{ref}), \phi(I_{ref})\}$ is not sufficient for the proposed method. We allocate slightly higher value to α_3 ($\alpha_3 = 0.5$), because the selected image patch in $\phi(I_{ref})$ will be finally used in $\mathcal{F}$. However, the similarity metric in the matching operation relies on both the LR features $\phi(I_{in})$ and the reference features $\phi(I^{\downarrow}_{ref})$, meaning that we can not neglect them during the gradient propagation process; thus, we set $\alpha_1 = \alpha_2 = 0.25$.

Table 4. A comparison of different settings for (α_1, α_2, α_3).

weights	**different combinations**			
(α_1, α_2, α_3)	(1, 0, 0)	(0, 1, 0)	(0, 0, 1)	(0.25, 0.25, 0.5)
PSNR	23.75	23.67	23.83	24.01
SSIM	0.697	0.672	0.695	0.705

5 Conclusions

In this paper, we explored a generalized problem for image super-resolution by utilizing high-resolution reference images. We proposed a match and swap module to obtain similar texture and high-frequency information from reference images, where end-to-end learning is enabled by properly distributing the gradients to the prior feature encoding module. Experiment results indicating that the matching feature is important in RefSR. For future work, we are going to study a better similarity metric for feature matching.

Acknowledgment

The work was supported by National Natural Science Foundation of China under 61972323, 61902022 and 61876155, and Key Program Special Fund in XJTLU under KSF-T-02, KSF-P-02, KSF-A-01, KSF-E-26.

References

1. Ahn, N., Kang, B., Sohn, K.A.: Fast, accurate, and lightweight super-resolution with cascading residual network. In: Proceedings of the European Conference on Computer Vision. pp. 252–268 (2018)
2. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255. IEEE (2009)
3. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE transactions on pattern analysis and machine intelligence **38**(2), 295–307 (2015)
4. Freedman, G., Fattal, R.: Image and video upscaling from local self-examples. ACM Transactions on Graphics **30**(2), 1–11 (2011)
5. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)
6. Jeon, D.S., Baek, S.H., Choi, I., Kim, M.H.: Enhancing the spatial resolution of stereo images using a parallax prior. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1721–1730 (2018)
7. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Proceedings of the European Conference on Computer Vision. pp. 694–711. Springer (2016)
8. Jolicoeur-Martineau, A.: The relativistic discriminator: a key element missing from standard gan. arXiv preprint arXiv:1807.00734 (2018)
9. Kim, J., Kwon Lee, J., Mu Lee, K.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1646–1654 (2016)
10. Kim, J., Kwon Lee, J., Mu Lee, K.: Deeply-recursive convolutional network for image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1637–1645 (2016)
11. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4681–4690 (2017)
12. Lim, B., Son, S., Kim, H., Nah, S., Mu Lee, K.: Enhanced deep residual networks for single image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 136–144 (2017)
13. Sajjadi, M.S., Scholkopf, B., Hirsch, M.: Enhancenet: Single image super-resolution through automated texture synthesis. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4491–4500 (2017)
14. Salvador, J.: Example-Based super resolution. Academic Press (2016)
15. Sun, L., Hays, J.: Super-resolution from internet-scale scene matching. In: 2012 IEEE International Conference on Computational Photography. pp. 1–12. IEEE (2012)
16. Timofte, R., De Smet, V., Van Gool, L.: Anchored neighborhood regression for fast example-based super-resolution. In: The IEEE International Conference on Computer Vision (December 2013)
17. Timofte, R., De Smet, V., Van Gool, L.: Anchored neighborhood regression for fast example-based super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1920–1927 (2013)

18. Wang, L., Wang, Y., Liang, Z., Lin, Z., Yang, J., An, W., Guo, Y.: Learning parallax attention for stereo image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 12250–12259 (2019)
19. Wang, Y., Wang, L., Yang, J., An, W., Guo, Y.: Flickr1024: A large-scale dataset for stereo image super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision Workshops. pp. 0–0 (2019)
20. Wang, Y., Lin, Z., Shen, X., Mech, R., Miller, G., Cottrell, G.W.: Event-specific image importance. In: The IEEE Conference on Computer Vision and Pattern Recognition (2016)
21. Wang, Y., Liu, Y., Heidrich, W., Dai, Q.: The light field attachment: Turning a dslr into a light field camera using a low budget camera ring. IEEE transactions on visualization and computer graphics **23**(10), 2357–2364 (2016)
22. Yue, H., Sun, X., Yang, J., Wu, F.: Landmark image super-resolution by retrieving web images. IEEE Transactions on Image Processing **22**(12), 4865–4878 (2013)
23. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: Proceedings of the European Conference on Computer Vision. pp. 286–301 (2018)
24. Zhang, Z., Wang, Z., Lin, Z., Qi, H.: Image super-resolution by neural texture transfer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7982–7991 (2019)
25. Zhao, H., Gallo, O., Frosio, I., Kautz, J.: Loss functions for image restoration with neural networks. IEEE Transactions on computational imaging **3**(1), 47–57 (2016)
26. Zheng, H., Guo, M., Wang, H., Liu, Y., Fang, L.: Combining exemplar-based approach and learning-based approach for light field super-resolution using a hybrid imaging system. In: Proceedings of the IEEE International Conference on Computer Vision Workshops. pp. 2481–2486 (2017)
27. Zheng, H., Ji, M., Han, L., Xu, Z., Wang, H., Liu, Y., Fang, L.: Learning cross-scale correspondence and patch-based synthesis for reference-based super-resolution. In: Proceedings of the British Machine Vision Conference (2017)
28. Zheng, H., Ji, M., Wang, H., Liu, Y., Fang, L.: Crossnet: An end-to-end reference-based super resolution network using cross-scale warping. In: Proceedings of the European Conference on Computer Vision. pp. 88–104 (2018)