A Practical Contrastive Learning Framework for Single-Image Super-Resolution

Gang Wu, Junjun Jiang, Senior Member, IEEE, and Xianming Liu, Member, IEEE

Abstract—Contrastive learning has achieved remarkable success on various high-level tasks, but there are fewer contrastive learning-based methods proposed for low-level tasks. It is challenging to adopt vanilla contrastive learning technologies proposed for high-level visual tasks to low-level image restoration problems straightforwardly. Because the acquired high-level global visual representations are insufficient for low-level tasks requiring rich texture and context information. In this article, we investigate the contrastive learning-based single-image super-resolution (SISR) from two perspectives: positive and negative sample construction and feature embedding. The existing methods take naive sample construction approaches (e.g., considering the low-quality input as a negative sample and the ground truth as a positive sample) and adopt a prior model (e.g., pretrained very deep convolutional networks proposed by visual geometry group (VGG) model) to obtain the feature embedding. To this end, we propose a practical contrastive learning framework for SISR (PCL-SR). We involve the generation of many informative positive and hard negative samples in frequency space. Instead of utilizing an additional pretrained network, we design a simple but effective embedding network inherited from the discriminator network which is more task-friendly. Compared with the existing benchmark methods, we retrain them by our proposed PCL-SR framework and achieve superior performance. Extensive experiments have been conducted to show the effectiveness and technical contributions of our proposed PCL-SR thorough ablation studies. The code and resulting models will be released via https://github.com/Aitical/PCL-SISR.

Index Terms—Contrastive learning, data augmentation, feature embedding, super-resolution.

I. INTRODUCTION

RECENTLY, contrastive learning has been an effective paradigm for unsupervised representation learning. Most approaches are developed based on the instance discrimination assumption and learn visual representations by making views from the same instance similar and that from different instances dissimilar [2], [3], [4], [5], [6]. The learned semantic representations are beneficial to a wide variety of downstream tasks, particularly high-level tasks, and achieve promising results, e.g., supervised image classification [7].

The authors are with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China (e-mail: gwu@hit.edu.cn; jiangjunjun@hit.edu.cn; cxmx@hit.edu.cn).

Manuscript received 16 September 2022; revised 23 February 2023 and 18 May 2023; accepted 20 June 2023. Date of publication 10 July 2023; date of current version 30 October 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 61971165 and Grant 61922027 and in part by the Natural Science Foundation of Heilongjiang Province under Grant YQ2020F004. (Corresponding author: Junjun Jiang.)

Digital Object Identifier 10.1109/TNNLS.2023.3290038

When it comes to low-level image processing tasks, there are some challenges to directly applying contrastive learning approaches. First, the learned global visual representations are inadequate for low-level tasks that call for rich texture and context information. Second, a series of data augmentations have been developed to generate positive and negative samples for high-level downstream tasks [3], [4], [11], [12]. However, except for some simple geometric augmentations (e.g., rotation augmentation in the widely used self-ensemble scheme), most of the complicated data augmentations cannot maintain the dense pixel correspondences and, thus, are not suitable for low-level tasks. Third, a meaningful latent space (or embedding space) is required for contrastive loss. In contrast to high-level tasks that try to obtain the best semantic representations, low-level tasks aim at reconstructing restored results in data space. It is of great importance to explore a proper and meaningful embedding space where the contrastive loss can be effectively defined.

Current contrastive learning-based methods for low-level tasks mainly focus on exploiting negative samples, while let the ground-truth image as the positive sample. For example, [13] treats the degraded image (the input hazy image) as a negative sample and presents a novel image dehazing method with a contrastive regularization. In [14] and [15], they take other examples in the dataset as negative samples for contrastive image super-resolution and underwater image restoration. These methods make a preliminary attempt and have proved the effectiveness of incorporating the contrastive constraint to low-level tasks. Another line of research is to model the statistical characteristics of the image by contrastive learning. Dong et al. [16] assume that two patches from the same sample have similar noise distributions, and two these from two different samples have two different noise distributions and develop a residual contrastive loss for joint demosaicking and denoising. In a similar vein, Chen et al. [17] develop an unpaired image deraining method based on rain space contrastive learning, which can better help rain removal when compared with the original image space. Zhang et al. [18] present a contrastive learning strategy in the feature channel space to obtain resolution-invariant features. They take feature maps of different channels as samples and postulate that the corresponding channels of low-resolution (LR) and high-resolution (HR) feature maps are positive, while those from different are negative. Wang et al. [19] apply the...
Our contributions are summarized as follows: 1) we propose a practical and general contrastive learning framework for the single-image super-resolution (SISR) task, and we take a valid data augmentation strategy to generate informative positive and negative samples for SISR; 2) we rethink and explore a novel way to obtain a task-friendly embedding network where contrastive loss works efficiently by reusing the discriminator of SR network; and 3) extensive experiments show that our method, dubbed PCL-SR, outperforms several representative SISR methods. In addition, ablation studies are conducted to analyze different components of our proposed method.

We will first give some related work of SISR and contrastive learning approaches in Section II. In Section III, we introduce our proposed PCL framework and explain it in detail. Then, Section IV describes our training settings and experimental results, including ablation analysis, where we compare the performance of our approach with other state-of-the-art (SoTA) methods. Finally, some conclusions are drawn in Section V.

II. RELATED WORK

In this section, we will briefly introduce the related work about self-supervised contrastive learning methods, contrastive learning methods for image restoration, contrastive learning for image-to-image (I2I) translation, and SISR methods.

A. Contrastive Learning

Contrastive learning has emerged as an effective paradigm for unsupervised representation learning, by maximizing mutual information. In recent years, several works [2], [3], [4], [23], [24] have studied this paradigm. The contrastive loss, which is similar to previous works [25], [26], [27] in deep metric learning, aims to push the anchor sample away from negative samples and pull it closer to positive samples in the latent space. The selection of negative and positive samples is crucial and depends on downstream tasks. In [2], [3], and [4], randomly augmented samples from the anchor sample serve as positive samples, while samples from others are negative. Tian et al. [11] analyze the optimal augmentation strategy in detail. Cai et al. [12] propose a feature space augmentation strategy, sampling numerous positive and negative samples from the latent feature distribution. Chiang et al. [20], Robinson et al. [21], and Sinha et al. [28] analyzed the impact of hard negative samples and exploited hard negative mining strategies to dig more informative negative samples.

B. Constractive Learning for Image Restoration

For single-image dehazing, Wu et al. [13] propose a method that utilizes a pretrained VGG model to obtain latent embeddings, where positive and negative samples are selected from...
the ground truth and corresponding hazy image, respectively, and contrastive loss is conducted using intermediate feature maps extracted from the VGG network. For blind super-resolution (BSR), Wang et al. [19] apply the contrastive loss to pretrain a kernel estimation model, which obtains a degradation-aware representation by distinguishing different degradations. The authors assumed that patches from the same image are under the same degradation and that patches from different images are not the same, thus selecting positive and negative samples accordingly. Zhang et al. [18] propose a two-stage BSR method. First, an image encoder is pretrained to learn resolution-invariant features by contrastive loss. Then, another contrastive loss is adopted to finetune the super-resolved results.

Recently, Liu et al. [22] introduce the contrastive prior into underwater image enhancement, which takes observed underwater images as negative samples and clear-in-air images as positive, and the pretrained VGG network is adopted as the feature embedding network to provide multilayer intermediate features.

C. Constrictive Learning for I2I Translation

Contrastive unpaired image-to-image translation (CUT) [29] is a pioneering work that proposed patch-based contrastive loss for unpaired I2I translation, which effectively obtains structure-preserved and style-transferred features in latent space. For a given patch of the input image, positive and negative samples are the corresponding transferred patch and other random patches from the same domain. Han et al. [30] propose the dual contrastive learning framework for unsupervised image-to-image translation (DCLGAN), which utilizes a bidirectional patch-based contrastive loss between the source and target domain. In recent work, Liu et al. [22] explore the effectiveness of negative samples in patch-based contrastive loss and proposed a negative pruning technology called ranking and pruning noise contrastive estimation (RankNCE), which reduces the number of negative samples by ranking the similarity score between the corresponding negative patches and the anchor patch feature. Ko et al. [31] introduce the self-supervised paradigm proposed in [6] into the I2I task and explored a self-supervised dense consistency regularization (DCR), where two augmented samples with an overlap region are taken as the input, and the DCR is calculated on the dense representation of the overlap region.

D. Single-Image Super-Resolution

Deep-learning-based methods have dominated the SISR field in recent years. Dong et al. propose the first convolutional neural network (CNN)-based SR (SRCNN) method. Since that, various efficient and deeper architectures have been proposed for SR, and the performance on benchmark datasets has been continuously improved by newly developed network architectures [32], [33], [34], [35], [36], [37]. In addition, Zhou et al. [38] propose adaptive patch feature aggregation with a graph neural network (IGNN). Chen et al. [39] proposed a multitask pretrained image processing transformer (IPT). Liang et al. [40] develop the image restoration using swin transformer (SwinIR) method, which introduces a hybrid architecture that combines Swin Transformer [41] and convolutions into images restoration tasks. Besides investigating more powerful network architectures, some perceptual-driven approaches have also been proposed utilizing perceptual loss functions to achieve better visual quality [42], [43], [44], [45].

In this article, we present a PCL-SR. We investigate the contrastive learning paradigm for SISR from two perspectives: sample generation and feature embedding. In contrast to the existing methods, which propose hard negative samples from semantic discrimination [20], [21], [28] or patch feature similarity [22], we exploit task-valid hard negative samples for SISR tasks by performing a slight blur operation. Moreover, while most existing methods adopt pretrained VGG as the feature embedding network, we propose an adaptive frequency-aware feature embedding approach, which learns task-friendly embedded features and is able to be sensitive to high-frequency information.

III. Method

In this section, we introduce our proposed PCL-SR in detail. We first introduce the preliminaries of contrastive learning, and then, we describe our positive and negative sample generation strategy and the training of our feature encoder. Finally, we present our main framework, which employs contrastive learning to further improve the performance of the existing SISR works.

A. Preliminaries

Contrastive learning is one of the most powerful approaches for representation learning. It aims at pulling the anchor sample close to the positive samples and pushing it far away from negative samples in latent space [2]. For the image dataset \( I \), the representation learning model \( E \) is trained to extract representations \( R = \{ r_i | r_i = E(I_i) \}, I_i \in I \) with InfoNCE loss [23], [24]. The loss \( \mathcal{L}_{\text{InfoNCE}} \) is based on a softmax formulation, and for the \( i \)th sample, the loss \( \mathcal{L}_i \) is as formulated as follows:

\[
\mathcal{L}_i = -\log \frac{\exp(r_i^T \cdot r_i^+ / \tau)}{\exp(r_i^T \cdot r_i^+ / \tau) + \sum_{j=1}^{K} \exp(r_i^T \cdot r_j^- / \tau)}
\]

(1)

where \( \tau \) is the temperature hyperparameter. \( r_i^+ \) represents the representation of the positive sample usually generated by random data augmentations from the same sample \( I_i \). \( K \) is the number of negative samples, and \( \{ r_j^- \}^K_{i=1} \) is the set of negative representations from negative samples \( \{ I_j | I_j \in I, j \neq i \}^K_{i=1} \) that are randomly selected other images from dataset. Finally, total contrastive loss is as follows:

\[
\mathcal{L}_{\text{InfoNCE}} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_i.
\]

(2)

In addition, the work in [7] modified and applied contrastive loss to the supervised classification task where there are more than one positive samples. For the \( i \)th image, this supervised contrastive loss is as follows:

\[
\mathcal{L}_i = -\frac{1}{P} \sum_{p=1}^{P} \log \frac{\exp(r_i^T \cdot r_i^+_p / \tau)}{\exp(r_i^T \cdot r_i^+_p / \tau) + \sum_{j=1}^{K} \exp(r_i^T \cdot r_j^-_p / \tau)}
\]

(3)
where $P$ is the number of the positive set, noted as $\{r^+_{p} \}_{p=1}^P$. Then, the total supervised contrastive loss $L_{\text{SupCL}}$ is

$$L_{\text{SupCL}} = \frac{1}{N} \sum_{i=1}^{N} L_i.$$  \hspace{1cm} (4)

Contrastive learning is beneficial to various downstream tasks and achieves promising performance. As we described in Section I, we can find that to employ contrastive learning methods, carefully designed sample selection and construction strategy and a task-related latent space need to be explored. Next, we will describe our sample generation strategy from the perspective of frequency domain and how to train a task-friendly embedding network in SISR instead of a pretrained model, such as VGG network.

**B. Positive and Negative Sample Generations**

The SISR task aims to transform LR images (noted as $I_{LR}$) into sharp, realistic, and HR images (noted as $I_{HR}$). As LR images are formulated from an image degradation process and contain only the low-frequency information, SR model focuses on learning a reverse translation to recover the lost high-frequency components (e.g., edge and texture information). With this in mind, we take valid data augmentations to generate our positive and negative samples.

1) **Informative Positive Sample Generation:** In addition to the only HR ground truth, we further generate $K_p$ sharpened images as positive set $P_i$ by applying different high-pass kernels on HR image. For the $i$th image, we denote its positive set as follows:

$$P_i = \{ P_j | P_j = \text{Sharpen}(I_{HR}^i) \}_{j=1}^{K_p},$$ \hspace{1cm} (5)

where $K_p$ is the number of positive samples. Sharpen presents a random sharpness function. This is different from the existing contrastive learning-based image restoration methods, which consider only the ground truth as the positive sample [13], [18]. It should be noted that in order to generate more informative positive samples, we apply different high-pass kernels to the HR ground truth. This positive sample generation strategy is designed following two observations: 1) the object of the SISR task is to obtain detailed results; we can use some informative positive samples to induce more high-frequency details for the reconstruction results and 2) SISR is an ill-posed problem, and the mapping between the LR and HR images is “one-to-many.” That is, the number of ground truth should not only be one, and there are many possible HR samples except the given ground truth [46]. Our proposed positive sample generation method can be seen as a very coarse one.

2) **Hard Negative Sample Generation:** In order to introduce contrastive learning to the low-level image restoration problems, recent works [13], [14], [15] simply take degraded images (e.g., the input hazy image or LR image) or other images in the dataset. When compared with the super-resolved image, low-quality negative samples are dissimilar and easy to be distinguished. Inspired from the hard negative samples mining and adversarial training methods [20], [21], [47], a natural idea is whether we can feed some difficult hard examples that are similar to ground truth as hard negative samples.

Here, we generate slightly blurry images from the ground truth as our hard negative sample set $N_i$, because they are close to the ground truth, thus forcing the reconstructed SR image become closer to the ground truth. For the $i$th image, we denote its negative set as follows:

$$N_i = \{ N_j | N_j = \text{Blur}(I_{HR}^i) \}_{j=1}^{K_N},$$ \hspace{1cm} (6)

where $K_N$ is the number of negative samples and we use $K_N = K_p = 4$ as default. Blur is blur function with a random Gaussian kernel. For the blur kernel setting, the size is uniformly sampled from $[3, 5, 7, 9, 11]$, and random Gaussian kernel samples the kernel width uniformly from $[0.3, 1.5]$.

**C. Feature Embedding Network**

In this section, we introduce a simple but efficient way to obtain a task-friendly embedding network. As described in Section I, VGG-based perceptual loss is widely adopted [43], [44], and recent work in [13] designs a contrastive loss based on the pretrained VGG model. We believe that a task-friendly embedding network is better, because features obtained by pretrained VGG tend to be high-level semantic information. In addition, compared with the SISR task, pretraining with ImageNet is a very heavy task. Furthermore, a good embedding network for SISR should be degradation-aware in order that contrastive loss can work even if the super-resolved results are very close to the ground truth. In other words, a good embedding network should be able to distinguish changes in detail.

Inspired by adversarial learning approaches in [43] and [48], we find that the discriminator learned in the vanilla generative adversarial networkinfo (GAN) framework is degradation-aware, because it can correctly distinguish whether the input image is fake or not. With this in mind, we employ a GAN-like framework to obtain our task-friendly embedding network by forcing it to distinguish SR and HR images, as illustrated in Fig. 1. Notably, to enhance the high-frequency components learning, we separate the image to low- and high-frequency parts, and our embedding network is trained with only the frequency components, which has been verified to be effective in real-world SR problems [49]. Here, we use Haar wavelet transform to extract the informative high-frequency components. The four sub-bands decomposed by Haar wavelet transform, noted as low-low frequency band (LL), low-high frequency band (LH), high-low frequency band (HL), and high-high frequency band (HH). Then, we stack the three high-frequency-related components (LH, HL, and HH) as the input and feed them to discriminator network $E_D$. Instead of training with conventional real or fake binary classification network, we adopt a contrastive discriminator loss proposed in [50] to train our $E_D$. It is a one-against-a-batch classification in the softmax cross-entropy formulation, and thus, the loss function of the $E_D$ can be formulated as follows:

$$L_{\text{CLD}} = \mathbb{E}_{\text{HR}} \left[ \log \frac{e^{E_D(\mathcal{H}_s(I_{LR}))}}{e^{E_D(\mathcal{H}_s(I_{LR}))} + \sum_{k=1}^{K_p} e^{E_D(\mathcal{H}_s(I_{LR}^k))}} \right]$$

$$+ \mathbb{E}_{\text{LR}} \left[ \log \frac{e^{-E_D(\mathcal{H}_s(I_{LR}))}}{e^{-E_D(\mathcal{H}_s(I_{LR}))} + \sum_{j=1}^{K_N} e^{-E_D(\mathcal{H}_s(I_{LR}^j))}} \right]$$ \hspace{1cm} (7)

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
where \( f_{SR}^{i} = F_{SR}(I_{LR}^{i}) \) represents super-resolved image and \( \mathcal{H}_w \) is the operator extracting LH, HL, and HH sub-bands and concatenating them.

**D. Contrastive Loss**

As described in Sections III-B and III-C, we employ a task-friendly embedding network to obtain a frequency-based latent space, and we also generate sharpness and blurry images as our multiple positive and negative samples. To fully utilize these samples, we conduct our contrastive loss with multi-intermediate features from embedding network \( E_D \) and propose our contrastive loss based on (3). For the target super-resolved image \( I_{SR}^{i} \), its generated positive and negative sets are noted as \( P_i \) and \( N_i \), respectively. The feature representations for the super-resolved image, positive sample, and negative sample are noted as \( f, p, \) and \( n \), respectively. We use superscript \( l \) as the layer index in \( E_D \). Our contrastive loss for the \( i \)th sample on the \( l \)th layer is defined as follows:

\[
    
    L_{i,l} = \frac{1}{K_P} \sum_{j=1}^{K_P} \log \frac{\exp(s(f_{l}^{i}, p_{l}^{j})/\tau)}{\sum_{k=1}^{K_P} \exp(s(f_{l}^{i}, n_{l}^{k})/\tau)} + \sum_{k=1}^{K_N} \exp(s(f_{l}^{i}, n_{l}^{k})/\tau)
\]

where \( s \) is the similarity function. Let us note the shape of feature map as \( C \times H \times W \). We adopt the mean value of pixelwise cosine similarity as the similarity between feature maps. \( s \) is defined as follows:

\[
    s(f, f') = \frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} \frac{f_h^x f_h'^y}{\|f_h^x\| \|f_h'^y\|}.
\]

Then, our total contrastive loss \( L_{CL} \) is as follows:

\[
    L_{CL} = \frac{1}{NL} \sum_{i=1}^{N} \sum_{l=1}^{L} L_{i,l}
\]

where \( N \) is the number of the training images and \( L \) is the number of feature layers we used. We take the first four intermediate layers to calculate our contrastive loss \( L_{CL} \), and the \( L \) is 4 as default.

**E. Training and Implementation Details**

For image reconstruction, we utilize \( \ell_1 \) loss as follows:

\[
    L_1 = \frac{1}{N} \|I_{SR}^{i} - I_{HR}^{i}\|_1.
\]

We train our SR network \( F_{SR} \) using \( L_1 \) loss and the proposed contrastive loss in (10). The total loss function \( L_{SR} \) is defined as follows:

\[
    L_{SR} = L_1 + \alpha L_{CL}
\]

where \( \alpha \) is a scaling parameter and we take \( \alpha = 0.1 \) as default.

In this article, we propose an efficient contrastive learning framework for SISR, as illustrated in Fig. 1. The discriminator

---

**Step 1:** Fix the SR network, train the embedding network with discriminator loss \( L_{CLD} \)

**Step 2:** Fix the embedding network, train the SR network with the reconstruction and contrastive loss \( L_1 + \alpha L_{CL} \)

---

Fig. 1. Overview of our proposed contrastive learning framework for SISR. We adopt the GAN-like training processing that updating our embedding network \( E_D \) and target SR network \( F_{SR} \) iteratively. We train our embedding network to learn degradation-aware features. Then, \( E_D \) is frozen, and SR network is trained with the pixelwise construction loss \( L_P \) and our contrastive loss \( L_{CL} \).
and the SR network are trained alternately. When training the discriminator, we fix the SR network, whose parameters are shared from the previous step. When training the SR network, the embedding network is frozen, and its parameters are the same as the discriminator of the previous step.

For training, we crop patches of size $48 \times 48$ from LR image with the corresponding HR patches. We augment the training data with random horizontal flips and 90 rotations. Our model is trained by Adam optimizer [51] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The batch size we used is 16.

### IV. Experiments

#### A. Experiment Setup

1) **Datasets and Metrics:** Following [32] and [35], we use diverse 2K resolution image dataset (DIV2K) dataset [52], which contains 800 images for training and 100 images for evaluation. Datasets for testing include Set5 [53], Set14 [54], B100 [55], Manga109 [56], and Urban100 [57] with the upscaling factor: $\times 4$. For comparison, we measure peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) [58] on the Y channel of transformed YCbCr space.

2) **Comparison Methods:** Our proposed contrastive learning framework for SISR task is generic, and it can be applied to any existing method. We compare our method with the SoTA methods. To evaluate our proposed method, we apply our propose ECL framework to the existing benchmark methods and retrain them. Here, we retrain enhanced deep residual networks for single image super-resolution (EDSR)\(^1\) [32], residual channel attention network (RCAN)\(^2\) [35], holistic attention network (HAN)\(^3\) [59], and non-local sparse network (NLSN)\(^4\) [60] with our contrastive learning framework. In addition, we also add some representative methods for comparison: SRCNN [61], fast super-resolution convolutional neural network (FSRCNN) [62], very deep super resolution (VDSR) [63], Laplacian pyramid super resolution network (LapSRN) [64], MemNet [65], single convolutional super-resolution network for multiple degradations (SRMDNF) [66], deep back-projection networks (D-DBPN) [34], residual dense network (RDN) [33], super-resolution feedback network (SRFBN) [67], second-order attention network (SAN) [68], IGNN [38], and SwinIR [40]. We use PyTorch [69] to implement our proposed approach and our code, and all trained models can be achieved at our project page.

#### B. Main Results

As summarized in Table II, we tabulate the quantitative results of different methods. The EDSR-S has 16 residual blocks and 64 channels, while EDSR-L is the large version and has 32 residual blocks and 256 channels. Compared with the existing methods, one can find that all of our retrained models surpass the original results on all the benchmark datasets.

| Method       | Set5 PSNR | Set5 SSIM | B100 PSNR | B100 SSIM | Urban100 PSNR | Urban100 SSIM | Manga109 PSNR | Manga109 SSIM |
|--------------|-----------|-----------|-----------|-----------|---------------|---------------|---------------|---------------|
| Bicubic      | 28.42     | 0.8104    | 26.00     | 0.7027    | 25.96         | 0.6675        | 23.14         | 0.6577        |
| SRCNN        | 30.48     | 0.8628    | 27.50     | 0.7513    | 26.90         | 0.7101        | 24.52         | 0.7221        |
| FSRCNN       | 30.72     | 0.8660    | 27.61     | 0.7550    | 26.98         | 0.7150        | 24.62         | 0.7280        |
| VDSR         | 31.35     | 0.8830    | 28.02     | 0.7680    | 27.29         | 0.7026        | 25.18         | 0.7540        |
| LapSRN       | 31.54     | 0.8850    | 28.19     | 0.7720    | 27.32         | 0.7270        | 25.21         | 0.7560        |
| MemNet       | 31.74     | 0.8893    | 28.26     | 0.7723    | 27.40         | 0.7281        | 25.30         | 0.7630        |
| SRMDNF       | 31.96     | 0.8925    | 28.35     | 0.7787    | 27.49         | 0.7337        | 25.68         | 0.7731        |
| D-DBPN       | 32.47     | 0.8980    | 28.82     | 0.7860    | 27.72         | 0.7400        | 26.38         | 0.7946        |
| RDN          | 32.47     | 0.8990    | 28.81     | 0.7871    | 27.72         | 0.7419        | 26.61         | 0.8028        |
| SRFBN        | 32.47     | 0.8983    | 28.81     | 0.7868    | 27.72         | 0.7409        | 26.60         | 0.8015        |
| SAN          | 32.64     | 0.9003    | 28.92     | 0.7888    | 27.78         | 0.7436        | 26.79         | 0.8068        |
| IGNN         | 32.57     | 0.8998    | 28.85     | 0.7891    | 27.77         | 0.7434        | 26.84         | 0.8090        |
| EDSR-S       | 32.09     | 0.8938    | 28.58     | 0.7813    | 27.57         | 0.7357        | 26.04         | 0.7849        |
| +PCL         | 32.17     | 0.8948    | 28.61     | 0.7825    | 27.58         | 0.7365        | 26.07         | 0.7863        |
| Gains        | 0.08      | 0.0010    | 0.03      | 0.0012    | 0.01          | 0.0008        | 0.03          | 0.0014        |
| EDSR-L       | 32.46     | 0.8968    | 28.80     | 0.7876    | 27.71         | 0.7420        | 26.64         | 0.8033        |
| +PCL         | 32.52     | 0.8993    | 28.84     | 0.7881    | 27.74         | 0.7428        | 26.71         | 0.8056        |
| Gains        | 0.06      | 0.0025    | 0.04      | 0.0005    | 0.03          | 0.0008        | 0.06          | 0.0019        |
| RCAN         | 32.64     | 0.9002    | 28.85     | 0.7885    | 27.75         | 0.7432        | 26.75         | 0.8066        |
| +PCL         | 32.70     | 0.9005    | 28.89     | 0.7889    | 27.78         | 0.7437        | 26.84         | 0.8083        |
| Gains        | 0.06      | 0.0003    | 0.04      | 0.0004    | 0.03          | 0.0005        | 0.09          | 0.0017        |
| HAN          | 32.60     | 0.9000    | 28.90     | 0.7891    | 27.79         | 0.7440        | 26.84         | 0.8093        |
| +PCL         | 32.68     | 0.9006    | 28.92     | 0.7897    | 27.80         | 0.7444        | 26.95         | 0.8115        |
| Gains        | 0.08      | 0.0006    | 0.02      | 0.0006    | 0.01          | 0.0004        | 0.11          | 0.0022        |
| NLSN         | 32.58     | 0.9000    | 28.87     | 0.7890    | 27.76         | 0.7441        | 26.96         | 0.8115        |
| +PCL         | 32.71     | 0.9012    | 28.99     | 0.7907    | 27.83         | 0.7455        | 27.12         | 0.8151        |
| Gains        | 0.13      | 0.0012    | 0.12      | 0.0017    | 0.07          | 0.0014        | 0.16          | 0.0036        |

\(^1\)https://github.com/sanghyun-son/EDSR-PyTorch  
\(^2\)https://github.com/yulunzhang/RCAN  
\(^3\)https://github.com/HarukiYuM/Non-Local-Sparse-Attention  
\(^4\)https://github.com/HarukiYuM/Non-Local-Sparse-Attention  

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Fig. 2. Visual comparison between the results of benchmark methods and our retrained counterparts. PSNR and SSIM scores are presented.
datasets. Specifically, compared with the NLSN model, our retrained NLSN has a gain of 0.16 dB/0.0022 (mean value of PSNR/SSIM) on five benchmark datasets, and when compared with the SoTA SwinIR, retrained NLSN can achieve comparable results on the term of PSNR among all benchmark datasets. In addition, retrained HAN can obtain comparable performance with the origin NLSN. Our retrained RCAN can achieve comparable performance with the newer SAN [68], and especially, it has superior performance on Set5, Urban100, and Manga109. From Table II, one can find that the proposed PCL framework further improves the existing SR models, and when we take a closer look at the performance on the Manga109 dataset where the larger gains can be obtained by our PCL, it shows that the proposed PCL can further improve the generalization of the SR models as well.

C. Qualitative Results

To intuitively demonstrate the effectiveness of our proposed method, we present visual results of the official and retrained models in Fig. 2. One can find that the retrained models produce clearer and more accurate textures than the official models by applying our contrastive learning constraint. Taking the images obtained by NLSN as an example, the retrained models are able to effectively recover clear content in the presence of artifacts. Furthermore, the proposed PCL paradigm enables more accurate texture reconstruction. For example, the retrained RCAN-PCL achieves accurate textures when the original RCAN fails to reconstruct, as shown in the second row of Fig. 2. We provide additional visual comparisons in Fig. 3 with diverse contents from the cross-domain Manga109 test dataset. One can find that retrained models always obtain more clear SR results. Furthermore, we present the SR results of both the original and retrained RCAN model, applied to one typical low-quality image, which is characterized by real-world scenarios and presents unknown degradation. These results, as shown in Fig. 4, indicate that the RCAN-PCL model achieves clearer edges and fewer artifacts, revealing the robustness and generalization of the proposed PCL framework.

In addition, we provide a quantitative comparison of our approach on three large test datasets using the NIQE and learned perceptual image patch similarity (LPIPS) metrics, as presented in Table III. Our results indicate that SR models retrained with the proposed PCL approach outperform the original models in most cases, revealing that the proposed PCL framework is effective at balancing distortion and perception.

D. Ablation Study

In this section, several ablation experiments are conducted to investigate different components in our proposed PCL-SR. We take the RCAN model as our baseline.
TABLE III
Qualitative Comparison on Test Datasets

| Method        | Set50 | Set14 | B100 | Urban100 | Manga109 |
|---------------|-------|-------|------|----------|----------|
|               | NIQE  | LPIPS | NIQE | LPIPS   | NIQE  | LPIPS  | NIQE | LPIPS  | NIQE | LPIPS  |
| EDSR          | 7.136 | 0.1801| 6.128| 0.2688  | 6.351 | 0.3023 | 5.435| 0.2356 | 5.143| 0.1364 |
| +PCL         | 7.072 | 0.1790| 6.021| 0.2681  | 6.276 | 0.3017 | 5.413| 0.2347 | 5.014| 0.1343 |
| RCAN         | 7.275 | 0.1801| 6.215| 0.2686  | 6.314 | 0.3022 | 5.321| 0.2324 | 5.225| 0.1354 |
| +PCL         | 6.964 | 0.1786| 6.126| 0.2681  | 6.389 | 0.3010 | 5.420| 0.2314 | 5.043| 0.1316 |
| NLSP         | 7.050 | 0.1787| 6.166| 0.2656  | 6.417 | 0.2975 | 5.586| 0.2358 | 5.344| 0.1311 |
| +PCL         | 6.861 | 0.1766| 6.214| 0.2639  | 6.372 | 0.2971 | 5.414| 0.2256 | 5.090| 0.1294 |

TABLE IV
Ablation Study Results. Here, Column $L_{CL}$ Presnets Types of Embedding Networks. VGG Means Adopting the Pretrained VGG Network, and D Presents Our Learnable Discriminator. H and F Are Shorted for Haar and Fourier Transformation, Respectively. $H + D$ and $F + D$ Present That the Embedding Network Takes the Frequency Transformed Input

| Config | $L_{CL}$ | Aug | Set50 | Set14 | B100 | Urban100 | Manga109 |
|--------|----------|-----|-------|-------|------|----------|----------|
|        |          |     |       |       |      |          |          |
| 1      | VGG      | ✓   | 32.62 | 0.8985| 28.84| 0.7875   | 27.73    | 0.7428   | 26.72 | 0.8059 | 31.12 | 0.9160 |
| 2      | D        | ✓   | 32.64 | 0.8994| 28.86| 0.7889   | 27.76    | 0.7444   | 26.82 | 0.8082 | 31.33 | 0.9177 |
| 3      | H + D    | ✓   | 32.70 | 0.9005| 28.89| 0.7889   | 27.78    | 0.7437   | 26.84 | 0.8083 | 31.41 | 0.9185 |
| 4      | H + D    | ✓   | 32.67 | 0.8999| 28.87| 0.7885   | 27.76    | 0.7435   | 26.83 | 0.8079 | 31.42 | 0.9185 |
| 5      | F + D    | ✓   | 32.66 | 0.8995| 28.80| 0.7893   | 27.77    | 0.7434   | 26.86 | 0.8089 | 31.40 | 0.9186 |

TABLE V
Results of EDSR-S. The PCL-F Presents That We Retrain EDSR-S Backbone With a Fixed Embedding Network, Which Is Pretrained With EDSR-L Backbone

| Method      | Set50 | Set14 | B100 | Urban100 | Manga109 |
|-------------|-------|-------|------|----------|----------|
|             | PSNR  | SSIM  | PSNR | SSIM    | PSNR     | SSIM   | PSNR | SSIM  | PSNR | SSIM  |
| EDSR-S      | 32.09 | 0.8938| 28.58| 0.7813  | 27.57    | 0.7357 | 26.04| 0.7849 | 30.35| 0.9067 |
| +PCL        | 32.17 | 0.8948| 28.61| 0.7825  | 27.58    | 0.7365 | 26.07| 0.7863 | 30.45| 0.9078 |
| +PCL-F      | 32.19 | 0.8948| 28.62| 0.7826  | 27.58    | 0.7365 | 26.09| 0.7866 | 30.49| 0.9079 |

1) Task-Generalized Embedding Network: In contrast to some other methods that employ prior VGG network to build their contrastive loss [13], [14], our embedding network is learned with the SR network like the discriminator in the GAN framework to obtain the task-friendly embedding network. Here, we analyze different types of the embedding network. Considering the prior VGG network is pretrained in RGB space, for a fair comparison, we remove the wavelet transform and train our embedding network in RGB space as well, indicated as Config 2. Then, we adopt the prior VGG-19 network to calculate our contrastive loss in RGB space, denoted as Config 1. The last is our full pipeline, where the wavelet transform is applied to highlight the difference in frequency space, and this embedding network is learned with frequency map correspondingly, noted as Config 3. Results are reported in Table IV. We can find that employing prior VGG network directly obtains even worse performance on benchmark datasets compared with baseline RCAN. Same with the observation in [13] and [14], they propose their contrastive losses instead of the original normalized $\ell_2$-based contrastive loss, because the original formulation cannot work properly. Since exploring a proper formulation for contrastive loss is beyond the scope of this article, we adopt the original contrastive loss and extend it to a pixelwise loss, such as [29]. Notably, our learned embedding network in Config 2 can achieve comparable performance with our basic contrastive loss, which shows that our embedding network is learned and becomes more proper for SISR task by distinguishing if the input is blurred. Finally, our full pipeline, employing the Haar transform, obtains the best performance.

In addition, when our embedding network is trained and finished, it is general to work with any SR backbones, such as the prior VGG. Here, we retrain the EDSR-S backbone with fixed embedding network pretrained with the EDSR-L backbone, noted with the suffix -F. Notably, when we apply our pretrained embedding network, it is used in the same way as the pretrained VGG network, and the results are shown in Table V. Some interesting observations can be obtained that the results of our PCL-F are slightly better. We think it is reasonable, because the pretrained embedding network based on the EDSR-L contains implicit knowledge to help the small backbone learn, and using a fixed embedding network avoids the unstable problems in GAN-like training processing. Considering there are several great GAN-based real-world SR works, such as [70], discriminators trained contain rich prior about the backbone and degradation. We think it is meaningful to explore these task-friendly prior in pretrained discriminators in feature works.

2) Sample Construction: As presented in Table IV, we compare the performance of Configs 3 and 4 with and without the proposed task-valid sample generation, demonstrating its effectiveness. In addition, we provide a decoupled analysis of the impact of positive and negative samples in Table VI. Primarily, it is notable that the incorporation of “hard”
negative samples can contribute positively to obtaining credible results. This is anticipated to enhance the texture accuracy of the super-resolved results. Second, our findings suggest that refining positive samples tends to yield superior qualitative performance, thereby bolstering the generalization capabilities of the SR network. This improvement is particularly evident in the cross-domain test set Manga109, further underscoring the effectiveness of our approach.

3) Spectrum Transformation: To help the SR network pay more attention to the losing high-frequency information, our contrastive loss works with the embedded frequency features, and we take the Haar transform as the default. Here, we take ablation studies to verify the impact of different types of wavelet transformation and results. Wavelet transformation is workable as discussed in Section IV-D1. Considering the Fourier transform is widely used as well, we adopt the FFT transform in Config 5, as shown in Table IV. Comparing Configs 3 and 5, we can find that utilizing the FFT transform can obtain comparable results as well, with better performance on Set14 and Urban100 while worse performance on Set5, B100, and Manga109. The transformed frequency map by Haar transform is only half the size of the original image, which highly speeds up training processing, and we take Haar transform as default.

4) Hyperparameters: To analyze the impact of the parameter \( \alpha \), we conduct training of the RCAN-PCL model at scale 4, employing various \( \alpha \) values. The results derived from the Set14 dataset are detailed in Table VII. Observably, an increasing of the \( \alpha \) value inversely affects the SR performance, leading to a decrease in effectiveness. In our experiments, we take \( \alpha = 0.1 \) as the default value.

E. Limitation and Discussion

In this article, we leverage contrastive learning approaches for low-level SISR tasks. While our proposed PCL paradigm can enhance the existing SR models, we also identify some challenges. First, in the proposed PCL, hard negative samples require a well-designed prior. It is not clear whether a potential metric can guide and improve sample generation. Second, we adopt a GAN-like framework to train the task-specific embedding network, which introduces some minor uncertainty during training. As shown in Table V, a pretrained embedding network can further improve performance. This observation is interesting and warrants further investigation in future work.

Our findings contribute to the development of contrastive learning approaches for low-level SR tasks. However, there are still challenges that need to be addressed in the future. We believe that improving task-friendly sample generation for image restoration tasks is a promising research direction. Moreover, as presented in Table V, embedding networks translated from pretrained SR models can achieve further improvement. For the low-level community, we suggest exploring how to extract more meaningful information or regularization from pretrained models.

V. Conclusion

In this study, we investigate the contrastive learning approaches in low-level tasks and propose a PCL framework for low-level SISR (PCL-SR) task. We take the contrastive learning paradigm into low-level SISR task from two perspectives: sample selection and feature embedding. Thus, valid data augmentations and task-friendly embedding networks are proposed. In contrast to the previous methods [13], [15], [17] that build contrastive samples with low-quality input and high-quality ground truth and employ the pretrained and fixed embedding network, we generate the negative and positive samples by our proposed task-valid augmentations, and to well capture the degradation-related difference, we employ an adaptive and learnable embedding network to obtain the task-friendly features. In experiments, some representative SR models are retrained by the proposed PCL. The retrained models achieve superior performance compared with the origin among all the test datasets, and detailed ablation studies are conducted to analyze the impact of different components. Furthermore, our PCL paradigm is generic and can be applied to many other image restoration tasks, and we will explore contrastive learning approaches for more low-level tasks in the future. Besides, as we summarized in Table V, expanding our PCL to explore and utilize the implicit information in pretrained models is another promising direction especially, since there are more and more huge pretrained models proposed nowadays.

### References

1. K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in Proc. ICLR, 2015, pp. 1–14.
2. Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, “Unsupervised feature learning via non-parametric instance discrimination,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 3733–3742.
3. T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A simple framework for contrastive learning of visual representations,” in Proc. 37th Int. Conf. Mach. Learn., vol. 119, Jul. 2020, pp. 1597–1607.
[55] D. Martin, C. Fowlkes, D. Tal, and J. Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,” in Proc. 8th IEEE Int. Conf. Comput. Vision (ICCV), Jun. 2001, pp. 416–423.

[56] Y. Matsui et al., “Sketch-based manga retrieval using manga109 dataset,” Multim. Tools Appl., vol. 76, pp. 2181–21838, Oct. 2017.

[57] J. Huang, A. Singh, and N. Ahuja, “Single image super-resolution from transformed self-exemplars,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 5197–5206.

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

[59] B. Niu et al., “Single image super-resolution via a holistic attention network,” in Proc. ECCV, 2020, pp. 191–207.

[60] Y. Mei, Y. Fan, and Y. Zhou, “Image super-resolution with non-local sparse attention,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 3516–3525.

[61] C. Dong, C. C. Loy, K. He, and X. Tang, “Image super-resolution using deep convolutional networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 2, pp. 295–307, Feb. 2016.

[62] C. Dong, C. C. Loy, and X. Tang, “Accelerating the super-resolution convolutional neural network,” in Proc. ECCV, 2016, pp. 391–407.

[63] J. Kim, J. K. Lee, and K. M. Lee, “Accurate image super-resolution using very deep convolutional networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 1646–1654.

[64] W. Lai, J. Huang, N. Ahuja, and M. Yang, “Deep Laplacian pyramid networks for fast and accurate super-resolution,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 3558–3566.

[65] Y. Tai, J. Yang, X. Liu, and C. Xu, “MemNet: A persistent memory network for image restoration,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 4549–4557.

[66] K. Zhang, W. Zuo, and L. Zhang, “Learning a single convolutional super-resolution network for multiple degradations,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 3262–3271.

[67] Z. Li, J. Yang, Z. Liu, X. Yang, G. Jeon, and W. Wu, “Feedback network for image super-resolution,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 3862–3871.

[68] T. Dai, J. Cai, Y. Zhang, S. Xia, and L. Zhang, “Second-order attention network for single image super-resolution,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 11057–11066.

[69] A. Paszke et al., “PyTorch: An imperative style, high-performance deep learning library,” in Proc. NeurIPS, vol. 32, 2019, pp. 1–11.

[70] X. Wang, L. Xie, C. Dong, and Y. Shan, “Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW), Oct. 2021, pp. 1905–1914.

Gang Wu received the B.E. degree from the School of Computer Science and Technology, Soochow University, Jiangsu, China, in 2020. He is currently pursuing the Ph.D. degree with the Faculty of Computing, Harbin Institute of Technology, Harbin, China.

His research interests include image restoration, representation learning, and self-supervised learning.

Junjun Jiang (Senior Member, IEEE) received the B.S. degree in mathematics from Huaqiao University, Quanzhou, China, in 2009, and the Ph.D. degree in computer science from Wuhan University, Wuhan, China, in 2014.

From 2015 to 2018, he was an Associate Professor with the School of Computer Science, China University of Geosciences, Wuhan. From 2016 to 2018, he was a Project Researcher with the National Institute of Informatics (NII), Tokyo, Japan. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China. His research interests include image processing and computer vision.

Dr. Jiang received the Best Student Paper Runner-Up Award at The International Multimedia Modeling Conference (MMM) 2017, the Finalist of the World’s FIRST 10k Best Paper Award at International Conference on Multimedia and Expo (ICME) 2017, and the Best Paper Award at International Forum of Digital TV and Multimedia Communication (IFTC) 2018. He received the 2016 China Computer Federation (CCF) Outstanding Doctoral Dissertation Award and the 2015 Association for Computing Machinery (ACM) Wuhan Doctoral Dissertation Award. He is an Associate Editor of Remote Sensing.

Xianming Liu (Member, IEEE) received the B.S., M.S., and Ph.D. degrees from the Harbin Institute of Technology (HIT), Harbin, China, in 2006, 2008, and 2012, respectively, all in computer science.

In 2011, he was a Visiting Student with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON, Canada, where he was a Post-Doctoral Fellow from 2012 to 2013. He was a Project Researcher with the National Institute of Informatics (NII), Tokyo, Japan, from 2014 to 2017. He is currently a Professor with the School of Computer Science and Technology, HIT.

Dr. Liu was a recipient of the IEEE International Conference on Multimedia and Expo (ICME) 2016 Best Student Paper Award.