Perception-Oriented Single Image Super-Resolution using Optimal Objective Estimation

Seung Ho Park[1,2], Young Su Moon[2], Nam Ik Cho[1,3]
[1]Department of Electrical and Computer Engineering, INMC, Seoul National University, Korea
[2]Visual Display Division, Samsung Electronics, Korea
[3]IPAI, Seoul National University, Korea

Abstract

Single-image super-resolution (SISR) networks trained with perceptual and adversarial losses provide high-contrast outputs compared to those of networks trained with distortion-oriented losses, such as L1 or L2. However, it has been shown that using a single perceptual loss is insufficient for accurately restoring locally varying diverse shapes in images, often generating undesirable artifacts or unnatural details. For this reason, combinations of various losses, such as perceptual, adversarial, and distortion losses, have been attempted, yet it remains challenging to find optimal combinations. Hence, in this paper, we propose a new SISR framework that applies optimal objectives for each region to generate plausible results in overall areas of high-resolution outputs. Specifically, the framework comprises two models: a predictive model that infers an optimal objective map for a given low-resolution (LR) input and a generative model that applies a target objective map to produce the corresponding SR output. The generative model is trained over our proposed objective trajectory representing a set of essential objectives, which enables the single network to learn various SR results corresponding to combined losses on the trajectory. The predictive model is trained using pairs of LR images and corresponding optimal objective maps searched from the objective trajectory. Experimental results on five benchmarks show that the proposed method outperforms state-of-the-art perception-driven SR methods in LPIPS, DISTS, PSNR, and SSIM metrics. The visual results also demonstrate the superiority of our method in perception-oriented reconstruction. The code is available at https://github.com/seungho-snu/SROOE.

1. Introduction

The purpose of single image super-resolution (SISR) is to estimate a high-resolution (HR) image corresponding to a given low-resolution (LR) input. SISR has many applications, mainly as a pre-processing step of computer vision or image analysis tasks, such as medical [13, 34, 51], surveillance [11, 43], and satellite image analysis [30, 41]. However, SISR is an ill-posed problem in that infinitely many HR images correspond to a single LR image. Recently, the performance of SISR has been greatly improved by adopting deep neural networks [7, 8, 15, 18, 21, 22, 26, 42, 54, 55]. Pixel-wise distortion-oriented losses (L1 and L2) were widely used in early research, which helped to obtain a high signal-to-noise ratio (PSNR). However, these losses lead the model to generate an average of possible HR solutions, which are usually blurry and thus visually not pleasing.

Subsequently, perception-oriented losses, such as perceptual loss [19] and generative adversarial loss [12], were

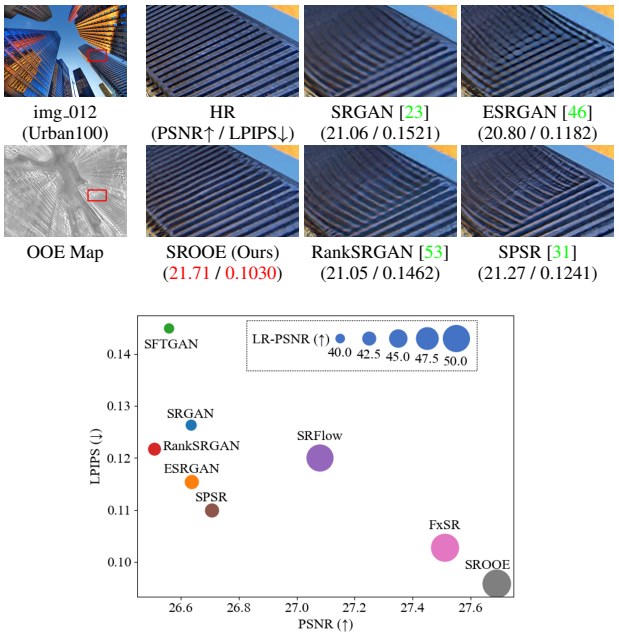


Figure 1. Visual and quantitative comparison. The proposed SROOE shows a higher PSNR, LR-PSNR [28] and lower LPIPS [52] than other state-of-the-art methods, *i.e.*, lower distortion and higher perceptual quality.

introduced to overcome this problem and produce realistic images with fine details [23]. Although these perception-oriented losses are used for various SR methods [38,46,50], they also bring undesirable side effects such as unnatural details and structural distortions. To alleviate these side effects and improve perceptual quality, various methods, such as the ones employing specially designed losses [40,53] and conditional methods utilizing prior information and additional network branches [31, 44], have been introduced. Meanwhile, different from the conventional SR methods, which optimize a single objective, some studies tried to apply multiple objectives to generate more accurate HR outputs. However, some of them [35, 45, 46] applied image-specific objectives without consideration for the regional characteristics, and the other [36] used region-specific objectives for the regions obtained using semantic image segmentation with a limited number of pre-defined classes.

In this paper, we propose a new SR framework that finds a locally optimal combination of a set of objectives in the continuous sample space, resulting in regionally optimized HR reconstruction. The upper part of Fig. 1 shows a visual comparison of our results with those of state-of-the-art perception-oriented methods. We can see that our SR method using optimal objective estimation (OOE), called SROOE, generates more accurate structures. The lower part of Fig. 1 shows that the SROOE is located on the far right and bottom, corresponding to the position where both PSNR and LPIPS [52] are desirable.

For this purpose, our SR framework consists of two models: a predictive model that infers the most appropriate objectives for a given input, and a generative model that applies locally varying objectives to generate the corresponding SR result. The main challenge is to train a single generator to learn continuously varying objectives over the different locations. For this, the objective is defined as the weighted sum of several losses, and we train the generator with various sets of weights. Meanwhile, the predictor is to estimate appropriate weights for a given image input.

For efficient training, we do not learn over the entire objective space spanned by the weight vector, but find a set of several objectives that have high impacts on optimization at each vision level and are close to each other in the objective space. This is because proximity between objectives improves the efficiency of learning and increases the similarity of their results, which helps reduce side effects. In addition, we train the generative model on a set of objectives on our defined trajectory, which is formed by connecting the selected objectives such that the trajectory starts with an objective suitable for a low-vision level and progresses through objectives suitable for higher levels. This enables us to replace high-dimensional weight vector manipulation with simple one-dimensional trajectory tracking, thereby simplifying the training process. The predictive model is trained using a dataset with pairs of LR images and corresponding optimal objective maps. We obtain these optimal training maps by using a grid search on the generator's objective trajectory.

Regarding the network structure, we employ spatial feature transform (SFT) layers [44] in the generator to flexibly change the network's behavior according to the objective. Our flexible model trained in this way has three advantages. First, the generalization capability to diversely structured images is improved since the network learns various cases. Second, the SR results are consistent with respect to the trajectory and given input. Third, the high-dimensional weight vector for loss terms can be replaced with a vector function with a one-dimensional input, and thus the optimal loss combinations can be easily found and controlled.

Our contributions are summarized as follows. (1) We propose an SISR framework that estimates and applies an optimal combination of objectives for each input region and thus produces perceptually accurate SR results. (2) While this approach requires training with various weighted combinations of losses, which needs the search on a high-dimensional weight vector space, we introduce an efficient method for exploring and selecting objectives by defining the objective trajectory controlled by a one-dimensional variable. (3) We propose a method for obtaining optimal objective maps over the trajectory, which are then used to train the objective estimator. (4) Experiments show that our method provides both high PSNR and low LPIPS, which has been considered a trade-off relation.

2. Related Work

Distortion-oriented SR. Dong *et al*. [8] first proposed a convolutional neural network (CNN)-based SR method that uses a three-layer CNN to learn the mapping from LR to HR. Since then, many deeper CNN-based SISR frameworks have been proposed [20,26]. Ledig *et al*. [23] proposed SRResNet, which uses residual blocks and skip-connections to further enhance SR results. Since Huang *et al*. [16] proposed DenseNet, the dense connections have become prevalent in SR networks [2, 3, 14, 46, 55]. Zhang *et al*. [54] introduced RCAN, which employs channel attention and improves the representation ability of the model and SR performance. More recently, SwinIR [24] and Uformer [49] reported excellent SISR performance by using the Swin Transformer architecture [27] and locally-enhanced window (LeWin) Transformer block, respectively. While there are many architectures for the SR as listed above, we employ plain CNN architectures as our predictor and generator. The structure is not an issue in this paper, and various CNNs and Transformers can be tried instead of our architecture.

Perception-oriented SR. Because the pixel losses, such as L1 and L2, do not consider perceptual quality, the results of using such losses often lack high-frequency de-

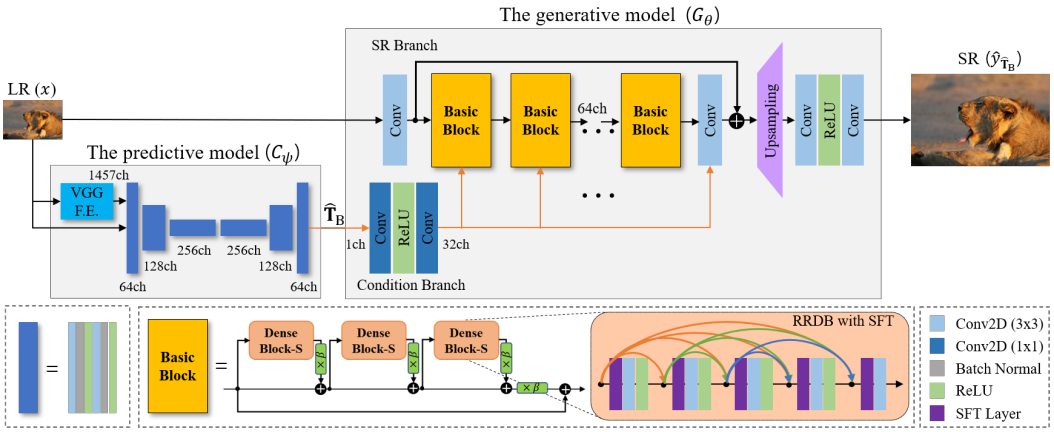


Figure 2. Architecture of the proposed method. The predictive model generates the optimal objective map $\hat{\mathbf{T}}_B$, which is fed to the generative model. The input LR image is super-resolved through our Basic Blocks and other elements of the generator, which are controlled by the map from the Condition Branch.

tails [47, 48]. Meanwhile, Johnson *et al*. [19] proposed a perceptual loss to improve the visual quality of the output. Ledig [23] introduced SRGAN utilizing adversarial loss [12], which can generate photo-realistic HR images. Wang *et al*. [46] enhanced this framework by introducing ESRGAN with Residual-in-Residual Dense Block (RRDB).

However, these perception-oriented SR models entail undesirable artifacts, such as unexpected textures on a flat surface. To alleviate such artifacts and/or further improve the perceptual quality, various methods have been proposed. Soh *et al*. [40] introduced NatSR, where they designed a loss to suppress aliasing. Wang *et al*. [44] proposed the use of semantic priors for generating semantic-specific details by using SFT layers. Zhang *et al*. [53] proposed a Ranker that learns the behavior of perceptual metrics. Ma *et al*. [31] proposed a structure-preserving super-resolution (SPSR) to alleviate geometric distortions. Liang *et al*. [25] proposed locally discriminative learning between GAN-generated artifacts and realistic details. However, Blau [5] argued that it is difficult to simultaneously achieve perceptual quality enhancement and distortion reduction because they involve a trade-off relationship. In this regard, there was an SR challenge [4] focused on the trade-off between generation accuracy and perceptual quality. One of the main claims of this paper is that we can further reduce distortion and increase perceptual quality simultaneously, as shown in Fig. 1.

3. Methods

3.1. Proposed SISR Framework

An overview of our SISR framework is presented in Fig. 2. Our framework consists of a predictive model C_ψ and generative model G_θ, parameterized by ψ and θ, respectively. Model C_ψ infers an LR-sized optimal objective map $\hat{\mathbf{T}}_\mathbf{B}$ for a given LR input x, and G_θ applies it to produce the corresponding SR output, which is as similar as possible to its corresponding HR counterpart y, as follows:

$$\hat{y}_{\hat{\mathbf{T}}_B} = G_\theta\left(x|\hat{\mathbf{T}}_B\right), \tag{1}$$

$$\hat{\mathbf{T}}_B = C_\psi\left(x\right). \tag{2}$$

3.2. Proposed Generative Model

Since using a single fixed objective cannot generate optimized HR results for every image region, it is beneficial to apply regionally different losses regarding the input characteristics. However, training multiple SR models, each of which is trained with a different objective, is impractical because it requires large memory and long training and inference times [10]. Hence, in this paper, we propose a method to train a single SR model that can consider locally different objectives.

Effective Objective Set. We first investigate which objectives need to be learned for accurate SR. For perception-oriented SR [23, 38], the objective is usually a weighted sum of pixel-wise reconstruction loss $\mathcal{L}_{rec}$, adversarial loss $\mathcal{L}_{adv}$, and perceptual loss $\mathcal{L}_{per}$, as follows:

$$\mathcal{L} = \lambda_{rec}\cdot\mathcal{L}_{rec} + \lambda_{adv}\cdot\mathcal{L}_{adv} + \sum_{per_l}\lambda_{per_l}\cdot L_{per_l}, \tag{3}$$

$$L_{per_l} = \mathbb{E}\left[\|\phi_{per_l}(\hat{y}) - \phi_{per_l}(y))\|_1\right], \tag{4}$$

$$per_l \in \{\text{V12, V22, V34, V44, V54}\}, \tag{5}$$

where λ_{rec}, λ_{adv}, and λ_{per_l} are weighting parameters for the corresponding losses, and $\phi_{per_l}(\cdot)$ represents feature maps of the input extracted at layer per_l of the 19-layer VGG network, where five layers denoted in Eqn. 5 are considered as in [35, 36, 39]. Since the receptive field becomes larger as we progress deeper into the VGG network [39],

Table 1. Performance comparison of SR results of ESRGAN models with different weight vectors for perceptual loss. Among the objectives in Sets A and B, except for $\boldsymbol{\lambda_0}$, the 1st and the 2nd best performances for each column are highlighted in red and blue.

Set	objective	$\boldsymbol{\lambda_{per}}$ $[\lambda_{V12}, \lambda_{V22}, \lambda_{V34}, \lambda_{V44}, \lambda_{V54}]$	Normalized L_{per_l} L_{V12}	L_{V22}	L_{V34}	L_{V44}	L_{V54}	Metric PSNR	LPIPS
	$\boldsymbol{\lambda_0}$	[0.0, 0.0, 0.0, 0.0, 0.0]	0.00	0.00	1.00	1.00	1.00	25.48	0.1960
A	$\boldsymbol{\lambda_1}$	[1.0, 0.0, 0.0, 0.0, 0.0]	0.71	0.53	0.76	0.38	0.25	23.95	0.1124
	$\boldsymbol{\lambda_2}$	[0.0, 1.0, 0.0, 0.0, 0.0]	0.72	0.26	0.35	0.22	0.15	23.84	0.1125
	$\boldsymbol{\lambda_3}$	[0.0, 0.0, 1.0, 0.0, 0.0]	0.81	0.51	0.00	0.02	0.04	23.66	0.1124
	$\boldsymbol{\lambda_4}$	[0.0, 0.0, 0.0, 1.0, 0.0]	0.92	0.78	0.51	0.11	0.05	23.28	0.1158
	$\boldsymbol{\lambda_5}$	[0.0, 0.0, 0.0, 0.0, 1.0]	1.00	1.00	0.93	0.32	0.12	23.00	0.1232
B	$\boldsymbol{\lambda_1}$	[1.0, 0.0, 0.0, 0.0, 0.0]	0.71	0.53	0.76	0.38	0.25	23.95	0.1124
	$\boldsymbol{\lambda_{1\text{-}2}}$	[1/2, 1/2, 0.0, 0.0, 0.0]	0.64	0.23	0.34	0.20	0.13	24.08	0.1075
	$\boldsymbol{\lambda_{1\text{-}3}}$	[1/3, 1/3, 1/3, 0.0, 0.0]	0.78	0.42	0.10	0.04	0.03	23.81	0.1112
	$\boldsymbol{\lambda_{1\text{-}4}}$	[1/4, 1/4, 1/4, 1/4, 0.0]	0.78	0.46	0.18	0.02	0.01	23.68	0.1110
	ESRGAN-OOS$_A$							24.03	0.0848
	ESRGAN-OOS$_B$							24.21	0.0848

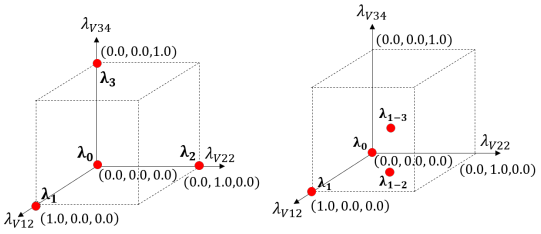


Figure 3. Set A (left) and set B (right) in the objective space. The objectives in set B are closer each other than those in set A.

features of shallow layers such as V12 and V22 and deeper layers such as V34, V44, and V54 correspond to relatively low-level and higher-level vision, respectively [36].

To find an effective set of objectives, we define an SR objective space. Since the objective for SR is a weighted sum of seven loss terms, as in Eqn. 3, an objective space is spanned by these basis loss terms, and any objective can be expressed by a seven-dimensional vector of weighting parameters, $\boldsymbol{\lambda_i} \in \mathbb{R}^7$ as $\boldsymbol{\lambda_i} = [\lambda_{rec}, \lambda_{adv}, \boldsymbol{\lambda_{per}}]$, where $\boldsymbol{\lambda_{per}} \in \mathbb{R}^5$ is a weight vector for perceptual loss.

Table 1 compares two objective sets, A and B, defined as shown in Fig. 3. Because ESRGAN [46] is the base model for this comparison, for all objectives in the table, except for $\boldsymbol{\lambda_0}$, λ_{rec} and λ_{adv} are set to 1×10^{-2} and 5×10^{-3}, respectively. These are the same as those for ESRGAN, except that $\boldsymbol{\lambda_{per}}$ changes, where $\|\boldsymbol{\lambda_{per}}\|_1 = 1$. In particular, in terms of $\boldsymbol{\lambda_{per}}$, whereas each objective $\boldsymbol{\lambda_i}$ in set A has weights for only one of the five VGG feature spaces, each objective in set B has equal weights for each loss in the feature space lower than the target vision level. Therefore, an objective corresponding to a high vision level also includes the losses for the lower-level feature spaces. Meanwhile, because $\boldsymbol{\lambda_0}$ corresponds to a distortion-oriented RRDB model [46], its λ_{rec} and λ_{adv} are set to 1×10^{-2} and 0, respectively. Note that $\boldsymbol{\lambda_0}$ is included in both sets A and B.

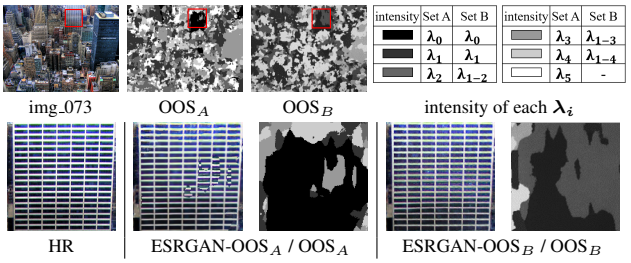


Figure 4. The OOS$_A$ and OOS$_B$ results using Sets A and B (top), their SR results, ESRGAN-OOS$_A$ and ESRGAN-OOS$_B$ (bottom).

In Table 1, the normalized versions (min-max feature scaling) of the averaged L_{per_l} from Eqn. 4 for five datasets (BSD100 [32], General100 [9], Urban100 [17], Manga109 [33], and DIV2K [1]) are reported. For all feature spaces, including the targeted V12 and V22 feature spaces, $\boldsymbol{\lambda_{1\text{-}2}}$ in set B has smaller L1 errors than those of $\boldsymbol{\lambda_1}$ and $\boldsymbol{\lambda_2}$ in set A. Moreover, $\boldsymbol{\lambda_{1\text{-}4}}$ exhibits smaller errors than those of $\boldsymbol{\lambda_4}$ and $\boldsymbol{\lambda_5}$. Although $\boldsymbol{\lambda_{1\text{-}3}}$ has slightly more errors in the V34 feature space than that of $\boldsymbol{\lambda_3}$, it has less errors in the V12 and V22 feature spaces therefore, $\boldsymbol{\lambda_{1\text{-}3}}$ has relatively less distortion than $\boldsymbol{\lambda_3}$ overfitted to the V34 feature space. That is supported by the fact that most of the objectives in set B, including $\boldsymbol{\lambda_{1\text{-}3}}$, have better PSNR and LPIPS on Urban100 [17] than those in set A. $\boldsymbol{\lambda_{1\text{-}5}}$ showing relatively poor performance compared to $\boldsymbol{\lambda_{1\text{-}4}}$ is not used.

To examine the SR result with locally appropriate objectives applied using set A, we mix the six SR results of ESRGAN-$\boldsymbol{\lambda_a}$, where $\boldsymbol{\lambda_a} \in A$, by selecting the SR result with the lowest LPIPS for each pixel position, as follows:

$$y^*_A(i,j) = \hat{y}_{\mathbf{T}^*_A(i,j)}(i,j), \tag{6}$$

$$\mathbf{T}^*_A(i,j) = \arg\min_{\boldsymbol{\lambda_a} \in A} \mathbf{LPIPS}_{\boldsymbol{\lambda_a}}(i,j), \tag{7}$$

$$\mathbf{LPIPS}_{\boldsymbol{\lambda_a}} = LPIPS\left(y, \hat{y}_{\boldsymbol{\lambda_a}}\right), \tag{8}$$

where $\hat{y}_{\boldsymbol{\lambda_a}}$ is the SR result of ESRGAN-$\boldsymbol{\lambda_a}$. The $LPIPS$ function computes the perceptual distance between two image patches for each pixel position, producing an LPIPS map, $\mathbf{LPIPS}$, of the input image size [28, 35]. The LPIPS metric in Table 1 is the average of this map. Since $\mathbf{T}^*_A$ is the optimal objective selection (OOS), $\mathbf{T}^*_A$ and its SR model for mixing are denoted as OOS$_A$ and ESRGAN-OOS$_A$, respectively. The upper part of Fig. 4 shows an example of OOS$_A$ and OOS$_B$ based on set A and B. PSNR and LPIPS [52] of ESRGAN-OOS$_A$ and ESRGAN-OOS$_B$ are reported in Table 1, where ESRGAN-OOS$_B$ is superior to any single objective model, demonstrating the potential for performance improvement of the locally suitable objective application. The lower part of Fig. 4 shows the side effects caused by mixing the SR results for set A with lower proximities between objectives than those in set B, as shown in Fig. 3. Since ESRGAN-OOS$_B$ in Fig. 4 has less artifact and better PSNR than those of ESRGAN-OOS$_A$, the proposed set B is more suitable for applying locally appropriate objectives.

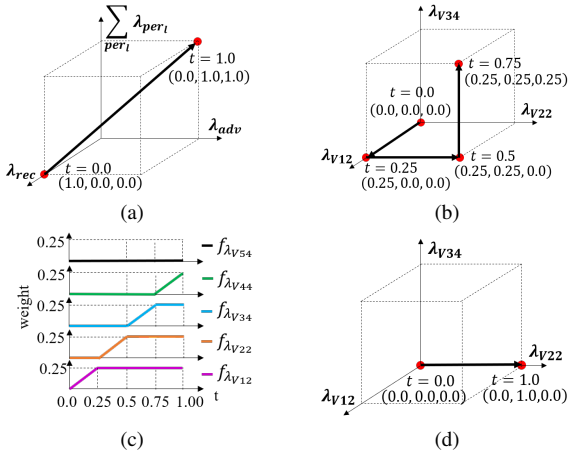


Figure 5. The proposed vector functions for loss weights, (a) $\boldsymbol{\lambda}(t)$ in Eqn. 9 when $\boldsymbol{\alpha}$=**1** and $\boldsymbol{\beta}$=**0**, (b) its $\boldsymbol{\lambda_{per}}(t)$ and (c) the weighting functions for $\boldsymbol{\lambda_{per}}(t)$. (d) $\boldsymbol{\lambda_{per}}(t)$ used for FxSR [35].

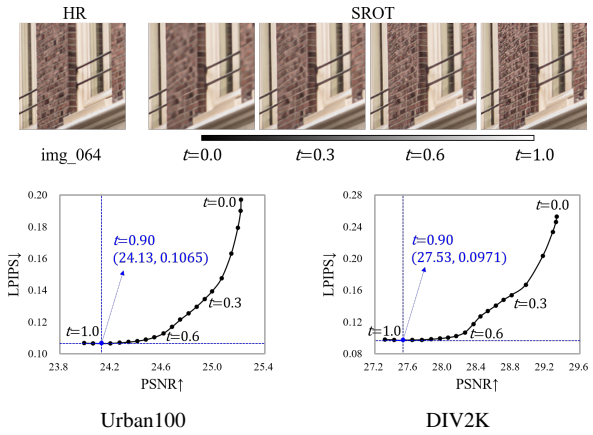


Figure 6. Changes in detail in the SROT results according to t-value (top) and changes in PSNR and LPIPS for test DBs (bottom).

Learning Objective Trajectory. We train our generative model on a set of objectives over the trajectory rather than a single objective, $\boldsymbol{\lambda_i}$. The objective trajectory is formed by connecting the selected objectives, i.e. the five objectives of set B, starting with an objective for a low-vision level and progressing through objectives for higher levels, i.e. from $\boldsymbol{\lambda_0}$ to $\boldsymbol{\lambda_{1\text{-}4}}$. It is parameterized by a single variable t , $\boldsymbol{\lambda}(t) = \langle \lambda_{rec}(t), \lambda_{adv}(t), \boldsymbol{\lambda_{per}}(t) \rangle$, as follow:

$$\boldsymbol{\lambda}(t) = \boldsymbol{\alpha} \cdot \boldsymbol{f}_{\lambda}(t) + \boldsymbol{\beta}, \tag{9}$$

$$\boldsymbol{f}_{\lambda}(t) = \left\langle f_{\lambda_{rec}}(t), f_{\lambda_{adv}}(t), \boldsymbol{f}_{\boldsymbol{\lambda_{per}}}(t) \right\rangle, \tag{10}$$

where $\boldsymbol{f}_{\boldsymbol{\lambda_{per}}}(t) \in \mathbb{R}^5$, $f_{\lambda_{rec}}(t)$, $f_{\lambda_{adv}}(t)$ are weighting functions, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are the scaling and offset vectors. As $\boldsymbol{f}_{\lambda} : \mathbb{R} \rightarrow \mathbb{R}^7$, this vector function enables the replacement of high-dimensional weight-vector manipulation with one-dimensional tracking, simplifying the training process.

Specifically, the trajectory design is based on the observation in Table 1 that the distortion-oriented RRDB model using $\boldsymbol{\lambda_0}$ has smaller L1 errors than those of all ESRGAN models for low-level feature spaces, such as V12 and V22, whereas ESRGAN models have smaller L1 errors for higher-level feature spaces, such as V34, V44, and V54. Thus, we design the weight functions $f_{\lambda_{rec}}$, $f_{\lambda_{adv}}$ and $\boldsymbol{f}_{\boldsymbol{\lambda_{per}}}$ such that when t approaches 0, $f_{\lambda_{rec}}$ increases and $\{f_{\lambda_{adv}}, \sum_{per_l} f_{\lambda_{per_l}}\}$ decrease to go to $\boldsymbol{\lambda_0}$, and conversely to go to $\boldsymbol{\lambda_{1\text{-}4}}$ when t increases to 1, as shown in Fig. 5(a).

In relation to the change in $\sum_{per_l} f_{\lambda_{per_l}}(t)$, we design each of five component functions, $f_{\lambda_{per_l}}(t)$ of $\boldsymbol{f}_{\boldsymbol{\lambda_{per}}}(t)$, as shown in Fig. 5(c), to obtain the objective trajectory from $\boldsymbol{\lambda_0}$ to $\boldsymbol{\lambda_{1\text{-}4}}$ of set B as shown in Fig. 5(b), illustrating only three out of five components because of the limitations of 3-dimensional visualization. Thus, as we progress through the trajectory by increasing t from 0 to 1, the weighting parameters for the objective start with the distortion-oriented objective, $\boldsymbol{\lambda_0}$, and then the losses of higher-vision-level feature spaces and adversarial loss are progressively added, making slight transitions on the objective toward $\boldsymbol{\lambda_{1\text{-}4}}$. Fig. 5(d) shows the objective trajectory used for FxSR [35], which uses only the V22 feature space, limiting the performance of the perceptually accurate restoration.

The proposed objective trajectory can efficiently improve the accuracy and consistency of the SR results. First, we can use any objective on the continuous trajectory from low to high-level vision, which allows the application of more accurate objectives to each region. Second, with regard to consistency, high-level objectives on our proposed trajectory include both low-level and high-level losses, thus also accounting for the low-level objectives. This weighting method allows the sharing of the structural components reconstructed mainly by low-vision-level objectives between all SR results on the trajectory. Finally, we need to train a single SR model only once, reducing the number of models required to produce diverse HR outputs [10, 35].

Fig. 6 shows the changes in the result of the generative SR model trained on the objective trajectory in Fig. 5(b), called SROT, as t changes from 0 to 1. The graphs in the bottom of Fig. 6 shows the trade-off curves in the perception-distortion plane according to the change of t, where t increases by 0.05 from 0.0 to 1.0 and has 21 sample points. Each SR result on the curve is obtained by inputting $\mathbf{T}$ with the same t throughout the image, as $\mathbf{T}_t = \mathbf{1} \times t$, into the condition branch of the generative model, as follows:

$$\hat{y}_{\mathbf{T}_t} = G_{\theta}\left(x | \mathbf{T}_t\right). \tag{11}$$

The horizontal and vertical dotted lines of the graphs in Fig. 6 indicate the lowest LPIPS values of the model and the corresponding PSNR values, respectively. The t values at that time are written next to the vertical lines. However, applying a specific t to the entire image still limits SR performance, and optimal t depending on images is unknown at inference time. We take this one step further and present later how to estimate and apply locally optimal objectives.

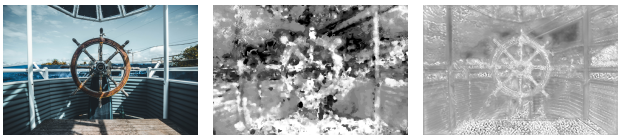


image (0824) OOS $\mathbf{T}_S^*$ OOE $\hat{\mathbf{T}}_B$

Figure 7. The input image, the optimal objective selection $\mathbf{T}_S^*$ obtained by parameter sweeping, and $\hat{\mathbf{T}}_B$ estimated by C_ψ.

Network Architecture and Training. The outline of the generator network is adopted from [35], *i.e.*, G_θ consists of two streams, an SR branch with 23 basic blocks and a condition branch as shown in Fig. 2. The condition branch takes an LR-sized target objective map $\mathbf{T}$ and produces shared intermediate conditions that can be transferred to all the SFT layers in the SR branch. Since the SFT layers [44] modulate feature maps by applying affine transformation, they learn a mapping function that outputs a modulation parameter based on $\mathbf{T}$. Specifically, $\mathbf{T}_t$, with t randomly changing in the pre-defined range, is fed into the condition branch during training, and this modulation layer allows the SR branch to optimize the changing objective by t. As a result, G_θ learns all the objectives on the trajectory and generates SR results with spatially different objectives according to the map at inference time. G_θ is optimized on the training samples $\mathcal{Z} = (x, y)$ with the distribution $P_\mathcal{Z}$, as follows:

$$\theta^* = \arg\min_\theta \mathbb{E}_{\mathcal{Z}\sim\mathcal{P}_\mathcal{Z}} \left[\mathcal{L}\left(\hat{y}_{\mathbf{T}_t}, y|t\right)\right], \tag{12}$$

$$\mathcal{L}(t) = \lambda_{rec}(t)\cdot\mathcal{L}_{rec} + \lambda_{adv}(t)\cdot\mathcal{L}_{adv} + \sum_{per_l}\lambda_{per_l}(t)\cdot L_{per_l}. \tag{13}$$

3.3. Optimal Objective Estimation (OOE)

To estimate an optimal combination of objectives for each region, we train a predictive model, C_ψ. This model produces an optimal objective map $\hat{\mathbf{T}}_B$ estimated for a given LR image, which is then delivered to the generative model in Eqn. 1. Since it is hard to find the ground truth map for C_ψ training, we obtain its approximation $\mathbf{T}_S^*$ via a simple exhaustive searching to narrow down the range of the best possible values. Specifically, we generate a set of 21 SR results by changing t from 0 to 1 with a step of 0.05, and the optimal objective maps are generated by selecting the t with the lowest LPIPS among them for each pixel, as

$$\mathbf{T}_S^*(i,j) = \arg\min_{t\in S} \mathbf{LPIPS}_t(i,j), \tag{14}$$

$$\mathbf{LPIPS}_t = LPIPS\left(y, \hat{y}_{\mathbf{T}_t}\right), \tag{15}$$

where $\mathbf{T}_t = \mathbf{1}\times t, t \in S = \{0.0, 0.05, 0.10, ..., 1.0\}$. Fig. 7 shows an example of the optimal objective selection (OOS) $\mathbf{T}_S^*$. The SR result using $\mathbf{T}_S^*$, SROOS, can be an upper-bound approximation for the performance of G_θ as

$$\hat{y}_{\mathbf{T}_S^*} = G_\theta\left(x|\mathbf{T}_S^*\right). \tag{16}$$

Although $\mathbf{T}_S^*$ is useful for training C_ψ, this pixel-wise objective selection without considering the interference caused by the convolutions of G_θ is not accurate ground truth. Therefore, C_ψ is optimized with three loss terms: pixel-wise objective map loss, pixel-wise reconstruction loss and perceptual loss, which measures the difference between the reconstructed and HR images, as follows:

$$\psi^* = \arg\min_\psi \mathbb{E}_{\mathcal{Z}_\mathbf{T}\sim\mathcal{P}_{\mathcal{Z}_\mathbf{T}}}\mathcal{L}, \tag{17}$$

$$\mathcal{L} = \lambda_\mathbf{T}\cdot\mathcal{L}_\mathbf{T} + \lambda_{rec}^{OOE}\cdot\mathcal{L}_{rec} + \lambda_R\cdot\mathcal{L}_R, \tag{18}$$

$$\mathcal{L}_R = \mathbb{E}\left[LPIPS\left(y, \hat{y}_{\hat{\mathbf{T}}_B}\right)\right], \tag{19}$$

where $\mathcal{L}_\mathbf{T}$ and $\mathcal{L}_{rec}$ is the L1 losses between $\mathbf{T}_S^*$ and $\hat{\mathbf{T}}_B$ and between y and $\hat{y}_{\hat{\mathbf{T}}_B}$, respectively. Meanwhile, $\mathcal{Z}_\mathbf{T} = (x, y, \mathbf{T}_S^*)$ is the training dataset, and $\lambda_\mathbf{T}$, λ_{rec}^{OOE} and λ_R are the weights for each of the loss terms, respectively. During the C_ψ model training, C_ψ is combined with the already trained generative model, and the generator parameters are fixed. Therefore, losses for C_ψ training, including LPIPS, are involved only in estimating locally-appropriate objective maps without changing the generator's parameters.

The architecture of C_ψ consists of two separate subnetwork: one is a feature extractor (F.E.) utilizing the VGG-19 [39] and the other is a predictor with the UNet architecture [37], as shown in Fig 2. For better performance, the feature extractor aims to get low to high-level features and delivers them to Unet, which makes the prediction. Since the structure of UNet has a wider receptive field, it is advantageous for predicting objectives in context.

4. Experiments

4.1. Experiment Setup

Materials, Evaluation Metrics and Training Details. We use either the DIV2K [1] (800 images) or the DF2K [26] (3450 images) dataset to train our models. Our test datasets include BSD100 [32], General100 [9], Urban100 [17], Manga109 [33], and DIV2K validation set [1]. To evaluate the perceptual quality, we report LPIPS [52] and DISTS [6], which are full-reference metrics. DISTS is a perceptual metric that focuses on detail similarity. PSNR and SSIM [47] are also reported as fidelity-oriented metrics. The LR-PSNR metric is the PSNR between the LR input and downscaled SR images. The higher the LR-PSNR, the better the consistency between the SR results and LR images, where 45 dB or more is recommended for good LR consistency as addressed in NTIRE challenge [28]. Because consistency with the LR input images is important, we also report the LR-PSNR. All training parameters are set to be equal to those of ESRGAN [46], except for the

Table 2. Comparison with state-of-the-art SR methods on benchmarks. The 1st and the 2nd best performances for each group are highlighted in red and blue, respectively. LR-PSNR values greater than 45dB are underlined.

		Distortion-oriented SR		Perception-oriented SR									
Model		RRDB [46]	SROOE ($\mathbf{T}=\mathbf{0}$)	SRGAN [23]	ESRGAN [46]	SFTGAN [44]	RankSR GAN [53]	SRFlow [29]	SPSR [31]	FxSR [35]	SROOE ($\hat{\mathbf{T}}_B$)	SROOE ($\hat{\mathbf{T}}_B$)	SROOS ($\mathbf{T}_S^*$)
Training dataset		DIV2K	DF2K	DIV2K	DF2K+OST	ImageNet+OST	DIV2K	DF2K	DIV2K	DIV2K	DIV2K	DF2K	DF2K
BSD100	PSNR↑	26.53	26.45	24.13	23.95	24.09	24.09	24.66	24.16	24.77	24.78	24.87	25.07
	SSIM↑	0.7438	0.7416	0.6454	0.6463	0.6460	0.6438	0.6580	0.6531	0.6817	0.6818	0.6869	0.6960
	LPIPS↓	0.3575	0.3546	0.1777	0.1615	0.1710	0.1750	0.1833	0.1613	0.1572	0.1530	0.1500	0.1388
	DISTS↓	0.2005	0.1996	0.1288	0.1158	0.1224	0.1252	0.1372	0.1165	0.1160	0.1139	0.1124	0.1104
	LR-PSNR↑	52.52	52.35	39.32	41.35	40.92	41.33	49.86	40.99	49.24	48.75	49.19	49.35
General100	PSNR↑	30.30	30.08	27.54	27.53	27.04	27.31	27.83	27.65	28.44	28.57	28.74	29.12
	SSIM↑	0.8696	0.8662	0.7998	0.7984	0.7861	0.7899	0.7951	0.7995	0.8229	0.8250	0.8297	0.8400
	LPIPS↓	0.1665	0.1658	0.0961	0.0880	0.1084	0.0960	0.0962	0.0866	0.0784	0.0764	0.0753	0.0682
	DISTS↓	0.1321	0.1311	0.0955	0.0845	0.1166	0.0938	0.1022	0.0857	0.0831	0.0811	0.0795	0.0783
	LR-PSNR↑	53.94	52.79	41.44	41.93	40.05	41.84	49.59	42.30	49.82	49.90	50.11	50.57
Urban100	PSNR↑	25.48	25.21	22.84	22.78	22.74	22.93	23.68	23.24	24.08	24.21	24.33	24.53
	SSIM↑	0.8097	0.8020	0.7196	0.7214	0.7107	0.7169	0.7316	0.7365	0.7641	0.7680	0.7707	0.7784
	LPIPS↓	0.1960	0.1961	0.1426	0.1230	0.1343	0.1385	0.1272	0.1190	0.1090	0.1066	0.1065	0.0988
	DISTS↓	0.1417	0.1409	0.1001	0.0818	0.0974	0.0987	0.0978	0.0798	0.0783	0.0773	0.0764	0.0774
	LR-PSNR↑	51.21	50.52	38.84	39.70	39.39	39.07	49.60	40.40	48.27	48.29	48.32	48.52
Manga109	PSNR↑	29.74	29.36	26.26	26.50	26.07	26.04	27.11	26.74	27.64	27.85	28.08	28.61
	SSIM↑	0.8997	0.8948	0.8285	0.8245	0.8182	0.8117	0.8244	0.8267	0.8440	0.8493	0.8554	0.8737
	LPIPS↓	0.0975	0.0972	0.0709	0.0654	0.0716	0.0773	0.0663	0.0683	0.0580	0.0566	0.0524	0.0431
	DISTS↓	0.0643	0.0605	0.0461	0.0397	0.0496	0.0488	0.0501	0.0403	0.0407	0.0382	0.0351	0.0344
	LR-PSNR↑	51.73	50.39	40.35	40.68	38.96	39.83	48.36	41.51	48.19	48.49	48.77	49.33
DIV2K	PSNR↑	29.48	29.33	26.63	26.64	26.56	26.51	27.08	26.71	27.51	27.57	27.69	28.03
	SSIM↑	0.8444	0.8413	0.7625	0.7640	0.7578	0.7526	0.7558	0.7614	0.7890	0.7906	0.7932	0.8031
	LPIPS↓	0.2537	0.2530	0.1263	0.1154	0.1449	0.1217	0.1201	0.1100	0.1028	0.0971	0.0957	0.0888
	DISTS↓	0.1261	0.1254	0.0613	0.0530	0.0858	0.0589	0.0622	0.0494	0.0513	0.0492	0.0491	0.0480
	LR-PSNR↑	53.71	53.59	40.87	42.61	40.40	41.90	49.96	42.57	50.54	50.36	50.80	51.04

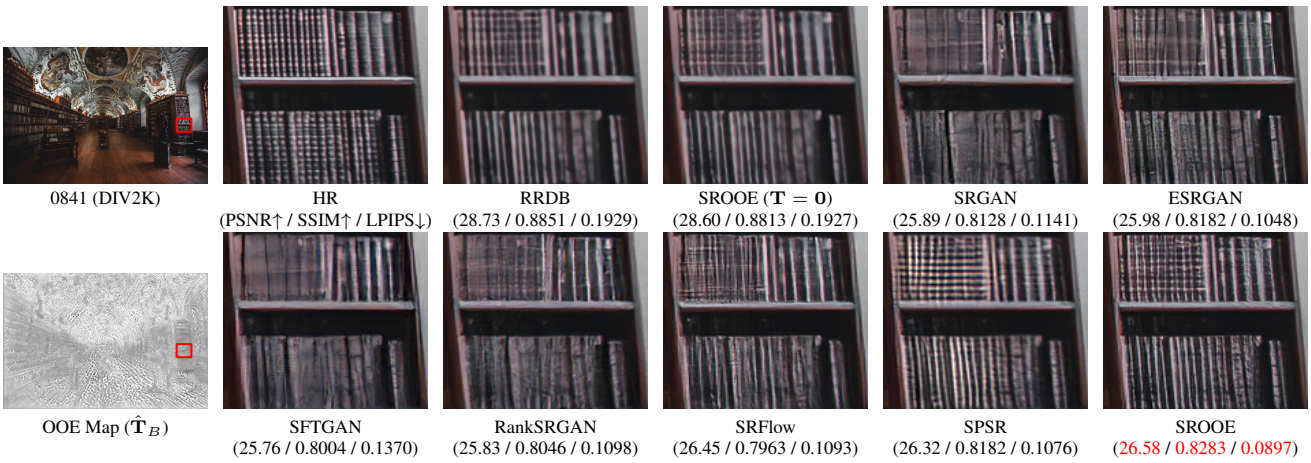


Figure 8. Visual comparison with state-of-the-art SR methods. Among the seven perception-oriented SR methods, the best performances are highlighted in red.

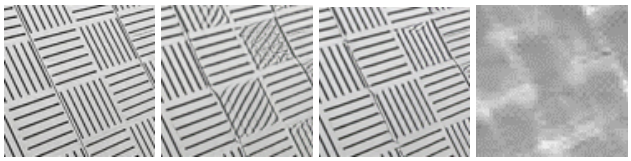
loss weights. For the generator training, t is a random variable with uniform distribution in [0, 1]. $\boldsymbol{\alpha}$=$[1 \times 10^{-2}, 1, \mathbf{1}]$ and $\boldsymbol{\beta}$=$[1 \times 10^{-2}, 0, \mathbf{0}]$.

4.2. Evaluation

Quantitative Comparison. Table 2 shows the quantitative performance comparison for the $4\times$ SR. We compared it with a distortion-oriented method, RRDB [46], and perception-oriented methods, such as SRGAN [23], ESRGAN [46], SFTGAN [44], RankSRGAN [53], SRFlow [29], SPSR [31], and FxSR [35]. The table shows that our method yields the best results among the perception-oriented methods on all datasets, not only in terms of LPIPS [52] and DISTS [6], but also in terms of distortion-

Table 3. Comparison of performance according to different selections. The bold checkmark indicates the change from the left selection. The 1st and the 2nd best performances except for SROOS are highlighted in red and blue, respectively.

Methods		SROT	SROOE			SROOS
Objective Trajectory	P2	✓	✓			
	P1234			**✓**	✓	✓
$\mathbf{T}$	$\mathbf{T}_{t=0.8}$	✓				
	$\hat{\mathbf{T}}_B$ (OOE)		**✓**	✓	✓	
	$\mathbf{T}_S^*$ (OOS)					**✓**
training DB	DIV2K	✓	✓	✓		
	DF2K				**✓**	✓
Metric	PSNR↑	26.49	26.53	26.65	26.74	27.07
	SSIM↑	0.7803	0.7816	0.7853	0.7872	0.7982
	LPIPS↓	0.1011	0.1000	0.0978	0.0960	0.0875
	LR-PSNR↑	49.21	49.21	49.32	49.44	49.76



Ground truth (PSNR↑ / LPIPS↓) | FxSR [35] (29.86 / 0.0525) | SROOE (30.14 / 0.0410) | OOE (4x magnified)

Figure 9. Visual comparison of the results of FxSR and SROOE.

Table 4. Comparison of the running time and the SR model size.

	SRGAN [23]	ESRGAN [46]	FxSR [35]	SROOE
Run Time (msec)	0.014	0.138	0.501	0.968
Param Size (MB)	1.51	16.69	18.30	70.20

oriented metrics such as PSNR and SSIM. It also exceeds 45 dB in LR-PSNR, indicating that LR consistency is well-maintained , as addressed in NTIRE [28]. In addition, SROOE using a local objective map outperforms SROT with the globally optimal t value for the Urban100 and DIV2K benchmarks in terms of both LPIPS and PSNR in Fig. 6. SROOS with $\mathbf{T}_S^*$ has the best PSNR, SSIM, LPIPS, and DISTS scores, which shows the approximated upper bounds of the proposed SROOE. On the other hand, when the objective map $\mathbf{T}$ is set to be $\mathbf{0}$, SROOE operates as a distortion-oriented SR model. Although it is slightly inferior to RRDB [46] in terms of PSNR, its performance is not far behind while showing better LPIPS. This implies that SROOE performs close to RRDB [46] for the regions needing distortion-oriented restoration, and thus the overall distortion is reduced while achieving high perceptual quality.

Qualitative Comparison. Fig. 8 shows a visual comparison, where we can observe that SROOE generates more accurate structures and details. In particular, it appears that there is little change in the structural component between the SROOE results using $\mathbf{T} = \mathbf{0}$ and $\hat{\mathbf{T}}_B$, and sharp edges and generated details are added to the structural components. Additional visual and quantitative comparisons for the $4\times$ and $8\times$ SR are provided in the supplementary.

5. Ablation Study

Table 3 reports average values in terms of each metric on all five benchmarks in Table 2, which vary according to the change in each element. The two different objective trajectories shown in Fig. 5(b) and (d) are referred to as P1234 and P2, respectively. The table shows that the SR performance improves step by step, when going from P2, fixed t, and DIV2K to P1234, OOE, and DF2K, respectively. Specifically, our proposed model, SROOE-P1234 trained with DF2K is improved by 0.25 dB in PSNR, 0.0069 in SSIM, 0.0051 in LPIPS, and 0.23 dB in LR-PSNR compared to SROT-P2 corresponding to FxSR with t=0.8 [35] as the example shwon in Fig. 9. A comparison of the running times and parameter sizes is presented in Table 4, where the time is for the $4\times$ SR of a 128×128 image on an NVIDIA RTX3090 GPU.

Limitations. Although applying locally appropriate objectives can significantly improve the LR to HR mapping accuracy, even if the generator uses an optimal objective map $\mathbf{T}_S^*$, it is still limited in achieving full reconstruction. This means that the proposed generator is still unable to generate all HRs with the objective set used for training in this study, and thus, more sophisticated perceptual loss terms than the VGG feature space are still required to overcome this. And still, there remains a limit to solving the ill-posed problem caused by high-frequency loss.

6. Conclusion

We proposed a novel SISR framework for perceptually accurate HR restoration, where an objective estimator provides an optimal combination of objectives for a given image patch, and a generator produces SR results that reflect the target objectives. For this purpose, we employed objective trajectory learning to efficiently train a single generative model that can apply varying objective sets. Experiments show that the proposed method reduces visual artifacts, such as structural distortion and unnatural details, and achieves improved results compared to those of state-of-the-art perception-oriented methods in terms of both perceptual and distortion metrics. The proposed method can be applied to off-the-shelf and other SISR network architectures.

7. Acknowledgment

This research was supported in part by Samsung Electronics Co., Ltd., in part by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (2021R1A2C2007220), and partially by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) [NO.2021-0-01343, Artificial Intelligence Graduate School Program (Seoul National University)].

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 126–135, 2017. 4, 6

[2] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Fast, accurate, and lightweight super-resolution with cascading residual network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2

[3] Saeed Anwar and Nick Barnes. Densely residual laplacian super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1192–1204, 2022. 2

[4] Yochai Blau, Roey Mechrez, Radu Timofte, Tomer Michaeli, and Lihi Zelnik-Manor. The 2018 pirm challenge on perceptual image super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, September 2018. 3

[5] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6228–6237, 2018. 3

[6] Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020. 6, 7

[7] Chao Dong, Chen Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2015. 1

[8] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*, pages 184–199. Springer, 2014. 1, 2

[9] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *European conference on computer vision*, pages 391–407. Springer, 2016. 4, 6

[10] Alexey Dosovitskiy and Josip Djolonga. You only train once: Loss-conditional training of deep networks. In *International Conference on Learning Representations*, 2019. 3, 5

[11] Muhammad Farooq, Matthew N Dailey, Arif Mahmood, Jednipat Moonrinta, and Mongkol Ekpanyapong. Human face super-resolution on poor quality surveillance video footage. *Neural Computing and Applications*, 33(20):13505–13523, 2021. 1

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 1, 3

[13] Hayit Greenspan. Super-resolution in medical imaging. *The Computer Journal*, 52(1):43–63, 2009. 1

[14] Muhammad Haris, Greg Shakhnarovich, and Norimichi Ukita. Deep back-projectinetworks for single image super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12):4323–4337, 2021. 2

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016. 1

[16] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017. 2

[17] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5197–5206, 2015. 4, 6

[18] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *JMLR Proceedings*, pages 448–456. JMLR.org, 2015. 1

[19] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 1, 3

[20] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1646–1654, 2016. 2

[21] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 1

[22] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 624–632, 2017. 1

[23] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 1, 2, 3, 7, 8

[24] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 1833–1844, 2021. 2

[25] Jie Liang, Hui Zeng, and Lei Zhang. Details or artifacts: A locally discriminative learning approach to realistic image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5657–5666, June 2022. 3

[26] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017. 1, 2, 6

[27] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10012–10022, October 2021. 2

[28] Andreas Lugmayr, Martin Danelljan, and Radu Timofte. Ntire 2021 learning the super-resolution space challenge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 596–612, June 2021. 1, 4, 6, 8

[29] Andreas Lugmayr, Martin Danelljan, Luc Van Gool, and Radu Timofte. Srflow: Learning the super-resolution space with normalizing flow. In *European Conference on Computer Vision*, pages 715–732. Springer, 2020. 7

[30] Yimin Luo, Liguo Zhou, Shu Wang, and Zhongyuan Wang. Video satellite imagery super resolution via convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 14(12):2398–2402, 2017. 1

[31] Cheng Ma, Yongming Rao, Yean Cheng, Ce Chen, Jiwen Lu, and Jie Zhou. Structure-preserving super resolution with gradient guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7769–7778, 2020. 1, 2, 3, 7

[32] 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 *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 416–423 vol.2, 2001. 4, 6

[33] Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. *Multimedia Tools and Applications*, 76(20):21811–21838, 2017. 4, 6

[34] Sung Cheol Park, Min Kyu Park, and Moon Gi Kang. Super-resolution image reconstruction: a technical overview. *IEEE Signal Processing Magazine*, 20(3):21–36, 2003. 1

[35] Seung Ho Park, Young Su Moon, and Nam Ik Cho. Flexible style image super-resolution using conditional objective. *IEEE Access*, 10:9774–9792, 2022. 2, 3, 4, 5, 6, 7, 8

[36] Mohammad Saeed Rad, Behzad Bozorgtabar, Urs-Viktor Marti, Max Basler, Hazim Kemal Ekenel, and Jean-Philippe Thiran. Srobb: Targeted perceptual loss for single image super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2710–2719, 2019. 2, 3, 4

[37] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. 6

[38] Mehdi SM Sajjadi, Bernhard Scholkopf, and Michael Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4491–4500, 2017. 2, 3

[39] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 3, 6

[40] Jae Woong Soh, Gu Yong Park, Junho Jo, and Nam Ik Cho. Natural and realistic single image super-resolution with explicit natural manifold discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8122–8131, 2019. 2, 3

[41] Huihui Song, Qingshan Liu, Guojie Wang, Renlong Hang, and Bo Huang. Spatiotemporal satellite image fusion using deep convolutional neural networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(3):821–829, 2018. 1

[42] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *Proceedings of the IEEE international conference on computer vision*, pages 4539–4547, 2017. 1

[43] Tõnis Uiboupin, Pejman Rasti, Gholamreza Anbarjafari, and Hasan Demirel. Facial image super resolution using sparse representation for improving face recognition in surveillance monitoring. In *2016 24th Signal Processing and Communication Application Conference (SIU)*, pages 437–440, 2016. 1

[44] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 606–615, 2018. 2, 3, 6, 7

[45] Xintao Wang, Ke Yu, Chao Dong, Xiaoou Tang, and Chen Change Loy. Deep network interpolation for continuous imagery effect transition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1692–1701, 2019. 2

[46] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018. 1, 2, 3, 4, 6, 7, 8

[47] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 3, 6

[48] Zhihao Wang, Jian Chen, and Steven C. H. Hoi. Deep learning for image super-resolution: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3365–3387, 2021. 3

[49] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17683–17693, June 2022. 2

[50] Xiangyu Xu, Deqing Sun, Jinshan Pan, Yujin Zhang, Hanspeter Pfister, and Ming-Hsuan Yang. Learning to super-resolve blurry face and text images. In *Proceedings of the*

IEEE international conference on computer vision, pages 251–260, 2017. 2

[51] Chenyu You, Guang Li, Yi Zhang, Xiaoliu Zhang, Hongming Shan, Mengzhou Li, Shenghong Ju, Zhen Zhao, Zhuiyang Zhang, Wenxiang Cong, Michael W. Vannier, Punam K. Saha, Eric A. Hoffman, and Ge Wang. Ct super-resolution gan constrained by the identical, residual, and cycle learning ensemble (gan-circle). *IEEE Transactions on Medical Imaging*, 39(1):188–203, 2020. 1

[52] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 1, 2, 4, 6, 7

[53] Wenlong Zhang, Yihao Liu, Chao Dong, and Yu Qiao. Ranksrgan: Generative adversarial networks with ranker for image super-resolution. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3096–3105, 2019. 1, 2, 3, 7

[54] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 286–301, 2018. 1, 2

[55] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2472–2481, 2018. 1, 2