IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 11, 2021, accepted August 23, 2021, date of publication August 27, 2021, date of current version September 7, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3108394

Reliable Perceptual Loss Computation for GAN-Based Super-Resolution With Edge Texture Metric

J. KIM AND C. LEE, (Member, IEEE)
School of Electrical and Electronic Engineering, Yonsei University, Seoul 03722, South Korea
Corresponding author: C. Lee (chulhee@yonsei.ac.kr)

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology under Grant NRF-2020R1A2C1012221.

ABSTRACT Super-resolution (SR) is an ill-posed problem. Generating high-resolution (HR) images from low-resolution (LR) images remains a major challenge. Recently, SR methods based on deep convolutional neural networks (DCN) have been developed with impressive performance improvement. DCN-based SR techniques can be largely divided into peak signal-to-noise ratio (PSNR)-oriented SR networks and generative adversarial networks (GAN)-based SR networks. In most current GAN-based SR networks, the perceptual loss is computed from the feature maps of a single layer or several fixed layers using a differentiable feature extractor such as VGG. This limited layer utilization may produce overly textured artifacts. In this paper, a new edge texture metric (ETM) is proposed to quantify the characteristics of images and then it is utilized only in the training phase to select an appropriate layer when calculating the perceptual loss. We present experimental results showing that the GAN-based SR network trained with the proposed method achieves qualitative and quantitative perceptual quality improvements compared to many of the existing methods.

INDEX TERMS Artificial neural networks, computer vision, image enhancement, image resolution.

I. INTRODUCTION

Super-resolution (SR) techniques, which are low-level vision problem solving methods, have been widely studied for pre-processing of high-level vision problems such as image classification [10], object detection [37], and semantic segmentations [38]. Furthermore, they are often used as image enhancement methods to improve quality by enlarging the spatial resolution of certain images.

When the spatial resolution of images is reduced, some information is permanently lost and cannot be recovered. Most SR techniques obtain information within the image itself [39], or from outside [40], [41] to restore the lost information. Recently, deep convolutional neural networks (DCN) have been used for many SR problems and they have shown impressive performance improvement. Although the current DCN-based SR techniques have some limitations and reliability issues [42], they can provide noticeably improved performance compared to traditional SR methods under controlled conditions.

DCN-based SR techniques are largely divided into two categories: peak signal-to-noise ratio (PSNR) oriented SR networks [1]–[9] and generative adversarial network (GAN) based SR networks [9], [10], [13]–[15]. PSNR-oriented SR networks are trained by the pixel loss that can be computed based on the distance between the pixel values in the image space. GAN-based SR networks additionally use both perceptual loss and adversarial loss in training, which helps the SR networks reconstruct perceptually satisfying SR images.

SRCNN [1], FSRCNN [2], VDSR [3], DRCN [4], SRResNet [9], EDSR [5], RDN [6], RCAN [7] and SAN [8] are some representative PSNR-oriented SR networks. They are trained by a pixel loss based on the mean absolute error (MAE) or mean squared error (MSE) of the pixel values in the image space. Consequently, SR images reconstructed by PSNR-oriented SR networks are not immune to the regression-to-mean problem [10] which is the dominant cause of SR images that appear blurry or overly smooth.

The associate editor coordinating the review of this manuscript and approving it for publication was Mingbo Zhao.

GAN-based SR networks widely use perceptual loss [11], [12] and adversarial loss [23] to solve regression-to-mean problems. Perceptual loss may help SR networks to use image context better and adversarial loss may help SR images to reside in the natural manifold when training. By using these losses, GAN-based SR networks achieve high texture reconstruction ability, thereby reconstructing SR images that are more perceptually satisfactory to humans. However, this also contributes to the creation of overly textured artifacts (Fig. 1). Most recent GAN-based SR networks have computed the perceptual loss from the feature maps of fixed layers (Fig. 2).

The SROBB [15] method uses semantic segmentation priors to reclassify labeled regions into three large categories: object, background, and boundary regions. These regions can be used to calculate the perceptual loss from the different layers of each category. However, using semantic segmentation priors can also reduce the areas of application.

To solve these problems, we used a new edge texture metric (ETM) to quantify images from texture-like images to edge-dominant images. We used this metric to adaptively select the appropriate layer to calculate the perceptual loss. The proposed method does not require any additional priors and it can be used with any of the SR datasets, not like the SROBB [15] and SFT-GAN [14] methods, which require semantic segmentation priors in the training phase or in both the training and inference phases. For this paper, the main contributions are as follows:

1) We propose an edge texture metric (ETM) to quantify images from texture-like images to edge-dominant images.
2) We propose an adaptive appropriate layer selection method to calculate perceptual loss based on the ETM.
3) Our method does not require any additional priors and it can be used with any of the SR datasets, not like the SROBB [15] and SFT-GAN [14] methods, which require semantic segmentation priors in the training phase or in both the training and inference phases.

II. RELATED WORKS

In the field of image reconstruction, a number of techniques can be used to train the generator networks. [58] proposed discriminative networks with generative networks while introducing a context loss function. [59] used the consistency loss with redesigned perceptual loss when training image inpainting networks. In [15], the targeted perceptual loss was used to improve the perceptual quality of the super-resolution results. [60] introduced a feature map attention mechanism to effectively utilize the low-frequency and high-frequency components of feature maps.

Recently, DCN-based methods have been successfully applied to super-resolution images with promising results. DCN-based super-resolution methods are largely divided into PSNR-oriented SR networks and GAN-based SR networks, and this also explains the perceptual loss that has a significant influence on the SR images of GAN-based SR networks.

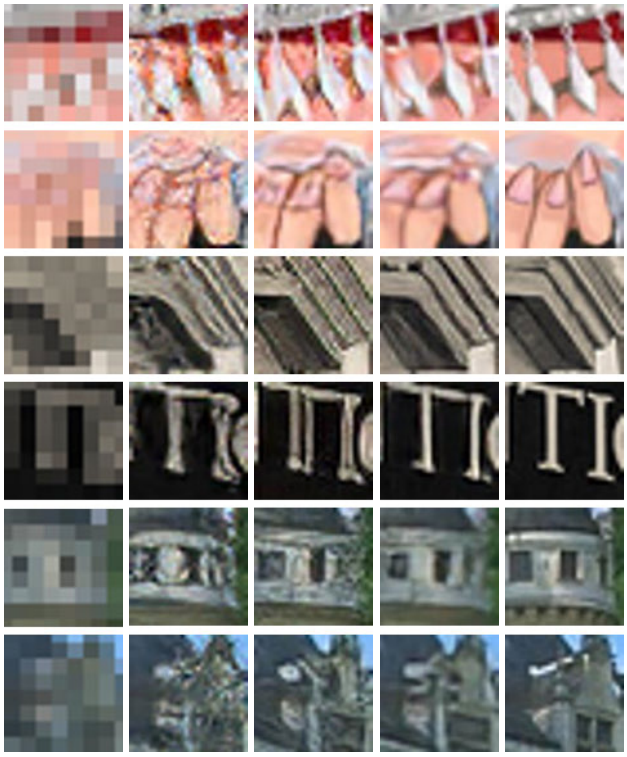

FIGURE 1. Examples of overly textured artifacts when the perceptual loss computed from feature maps of fixed layers were used for training (scale factor 4, enlarged).

A. PSNR-ORIENTED SUPER-RESOLUTION NETWORKS

SRCNN [1] is the pioneering work of DCN-based SR techniques which allows the DCN to train mapping between high-resolution (HR) images and low-resolution (LR) images:

$$\mathcal{L}_{pix}\left(\theta_F\right) = \frac{1}{N}\sum_{i=1}^{N} D(F(\mathrm{LR}_i), \mathrm{HR}_i). \tag{1}$$

where LR_i represents the i-th LR image, HR_i represents the i-th HR image of training images. $F(\cdot)$ denotes the SR network, $D(\cdot)$ denotes a distance measurement function such as MAE or MSE, N denotes the number of training image pairs (LR-HR), θ_F denotes the trainable parameters of F, and $\mathcal{L}_{pix}(\cdot)$ denotes the loss function.

Kim *et al.* [3] proposed the VDSR, which used a deep CNN architecture inspired by VGG networks [18] that showed high performance across a wide range of image classification problems. To train the deep structure of the SR network, Kim *et al.* designed the architecture of the SR network in a global residual way and used large learning rate settings and adaptive gradient clippings to achieve faster convergence of the SR network.

Inspired by ResNet [21], Ledig *et al.* [9] actively utilized the residual block and skip-connection in the design of SRResNet. This allowed a deeper architecture of the SR network to be stably trained. A number of SR networks have used several residual blocks and skip-connections in their architecture since the emergence of SRResNet.

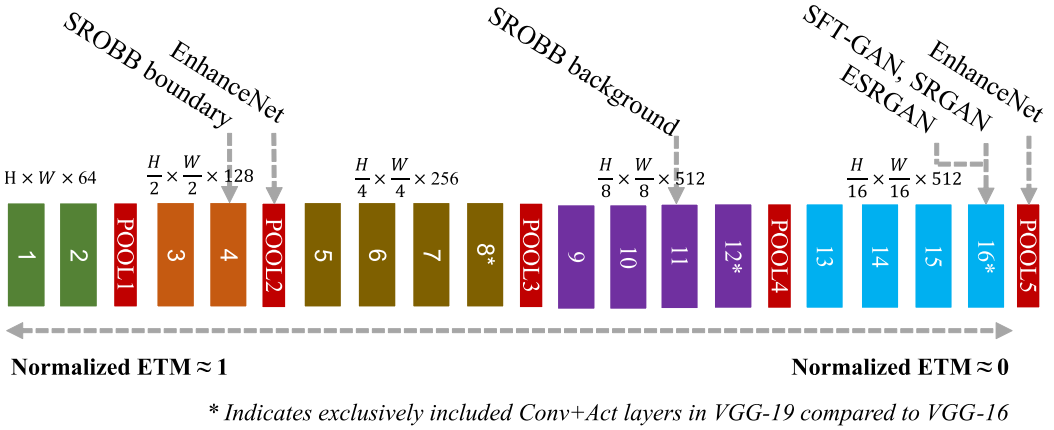

FIGURE 2. Indices represent the index of the VGG-16 & 19 [18] convolutional layers. The arrows indicate the location of the layers where the perceptual loss was calculated in recent GAN-based SR networks. The proposed adaptive loss layer selection method based on an edge texture metric is also illustrated. Whereas recent GAN-based SR networks calculated the perceptual loss by using a limited number of layers, the proposed method includes all layers by utilizing adaptive loss layer selection.

Lim *et al.* [5] proposed the EDSR with a structure similar to SRResNet. In addition, Lim *et al.* reported that removing the batch normalization (BN) layer [22] improved the performance of the SR network.

RCAN [7] and SAN [8] also improved the SR performance by analyzing the first-order and second-order correlations between the channels of the feature map. They also noted that some existing SR networks might have neglected to analyze the correlations between the channels.

Zhang *et al.* [61] proposed an end-to-end trainable SR model that blends the advantages of the training-based method and the model-based method with a half-quadratic splitting algorithm.

Mei *et al.* [62] noted that both non-local operation and sparse representation are crucial for SR networks and this led them to propose a non-local sparse attention (NLSA) method which is robust in terms of retaining long-range modeling capability.

B. PERCEPTUAL LOSS

The regression-to-mean problem commonly observed in SR images reconstructed by PSNR-oriented SR networks that use pixel loss in training SR networks was a major obstacle in previous image reconstruction problems. Johnson *et al.* [11] and Dosovitskiy *et al.* [12] proposed a perceptual loss method that might increase the range flexibility of the data generated by the SR networks and help reconstruct complex textures by calculating the distance from the feature space of selected layers (and not from the image space). The perceptual loss was computed as follows:

$$\mathcal{L}_{percep}(\theta_F) = \frac{1}{N}\sum_{i=1}^{N} D(E(F(\mathrm{LR}_i)), E(\mathrm{HR}_i)). \tag{2}$$

where $E(\cdot)$ denotes a pre-trained differentiable feature extractor such as the VGG [18], which is widely used as a feature extractor for computing perceptual loss in many recent GAN-based SR networks [10], [13]–[15], [43]. The pre-trained VGG-16 & 19 methods have a deep architecture with many trainable parameters. Also, VGG-16 & 19 do not have the type of residual structure that has been adopted by many recent DCN models. Thus, it is easy to extract features intuitively depending on the depth of the layer with VGG. $\mathcal{L}_{percep}(\cdot)$ represents the perceptual loss function. Perceptual loss is utilized not only by the GAN-based SR but also by image inpainting [45] and image-to-image translation [46].

C. GENERATIVE ADVERSARIAL NETWORKS

In [23], the GAN technique (the pioneering methodology for training generator networks with discriminator networks via an adversarial process) was proposed. The value function that is a type of minimax game was performed between the generator and the discriminator as:

$$\begin{aligned} &\min_{\theta_{F_G}} \max_{\theta_{F_D}} V(F_G, F_D) \\ &= E_{Y\sim p_{HR}}[log(F_D(Y))] + E_{X\sim p_{LR}}[1 - log(F_D(F_G(X)))]. \end{aligned} \tag{3}$$

This competitive training relationship defines adversarial loss applied to the generator that makes the distribution of the generator output closer to the distribution of the training data.

In [48], Pourya *et al.* presented a comprehensive survey of a wide range of aspects such as code, training datasets, and evaluation methods.

The GAN technique has been widely applied and studied in various computer vision fields, such as texture synthesis [49], text-to-image generation [50], [51], image-to-image translation [52]–[54], and image inpainting [55], [56].

D. GAN-BASED SR NETWORKS

Ledig *et al.* [9] and Sajjadi *et al.* [10] also introduced the concepts of perceptual loss and adversarial loss obtained by

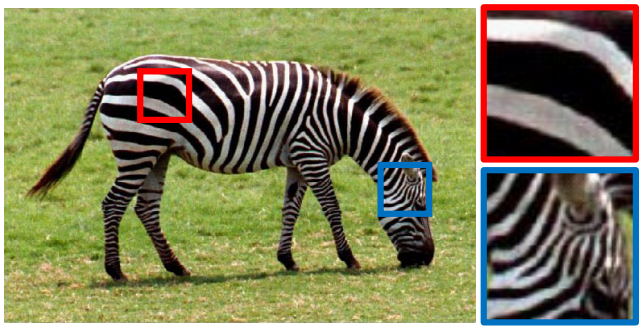

FIGURE 3. The same label of image patches does not mean that they have the same characteristics. Therefore, it is not reasonable to calculate the perceptual loss in the feature maps of the same layer of both patches.

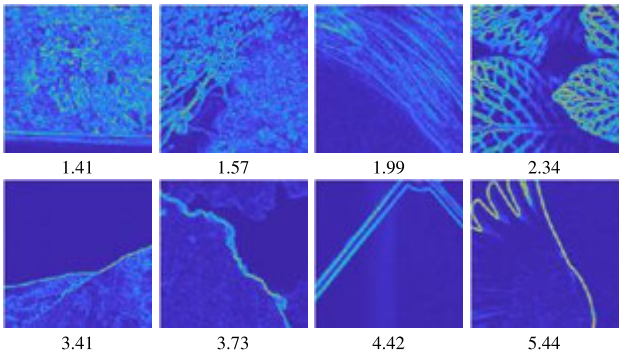

FIGURE 4. Randomly cropped 192 × 192 g_{ep} samples and corresponding ETM values.

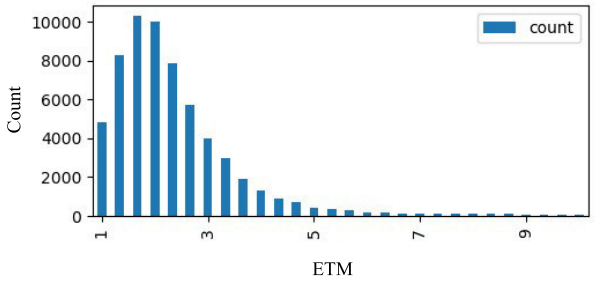

FIGURE 5. ETM value histogram of 61580 randomly cropped patches of the DIV2K dataset [28].

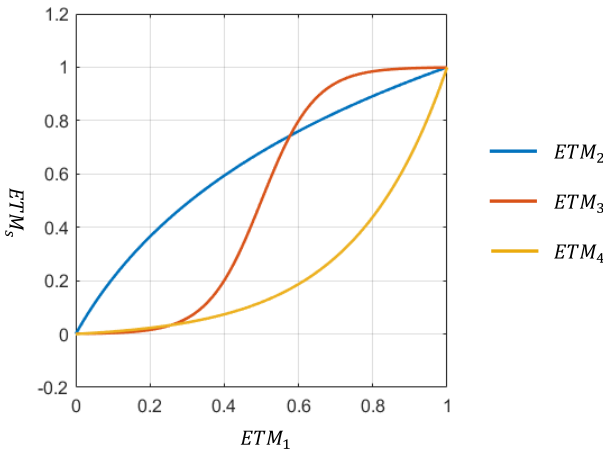

FIGURE 6. Normalized ETM_S.

the GAN frameworks in the training phase of the SR networks. This made the SR networks more able to reconstruct more perceptually satisfying SR images. Due to these characteristics, many GAN-based SR networks may perform poorly in terms of image space errors (i.e. MAE, MSE or PSNR values) compared to PSNR-oriented SR networks. However, their results may be more perceptually satisfying. For a quantitative evaluation of the perceptual quality of SR images, the perceptual index (PI) [24] and LPIPS [29] methods have already been widely used.

SFT-GAN [14] is a GAN-based SR network utilizing semantic segmentation priors with a spatial feature transform (SFT) layer that adaptively modifies the tendency of features for each segmented label. However, the major constraint of the SFT-GAN network is that the SFT layer requires semantic segmentation priors during the training and testing phases.

The SROBB [15] method also used semantic segmentation priors to reclassify the labeled regions into three categories: object, background, and boundary regions. The perceptual loss was calculated at different feature layers for each category. However, this also required semantic segmentation priors during the training phase and there were some errors in selecting the appropriate feature layer for various categories. Figure 3 illustrates the ambiguity of whether semantic segmentation prior is suitable for the proper selection of the layer to calculate the perceptual loss.

The ESRGAN [13] method introduced the relativistic discriminator [19] which helps to reconstruct realistic textures when training GAN-based SR networks.

III. PROPOSED METHODS

In this section, we propose a method to calculate the perceptual loss by adaptively selecting the appropriate feature layer (Fig. 2). First, we developed a metric to quantify all of the images from texture-like images to edge-dominant images.

A. EDGE TEXTURE METRIC

SSIM [26] is a metric that measures the structural similarity between two images through statistical analysis by utilizing the means and standard deviations of the two images. Inspired by the SSIM, we developed an edge texture metric (ETM) to quantify the images from texture-like images to edge-dominant images. We first converted an RGB image into a gray image (A) and then computed the edge magnitude image g_m using the Sobel operator:

$$g_m(A) = \|g(A)\|_2 = \sqrt{g_x^2(A) + g_y^2(A)}. \quad (4)$$

where $g(A)$ is composed of an approximation of column-wise gradient $g_x(A)$ and an approximation of row-wise gradient $g_y(A)$ values. Next, we computed a normalized edge image (g_p) from g_m as follows:

$$g_p(A) = \frac{g_m(A)}{max_{g_m}} = \frac{g_m(A)}{1140.395}. \quad (5)$$

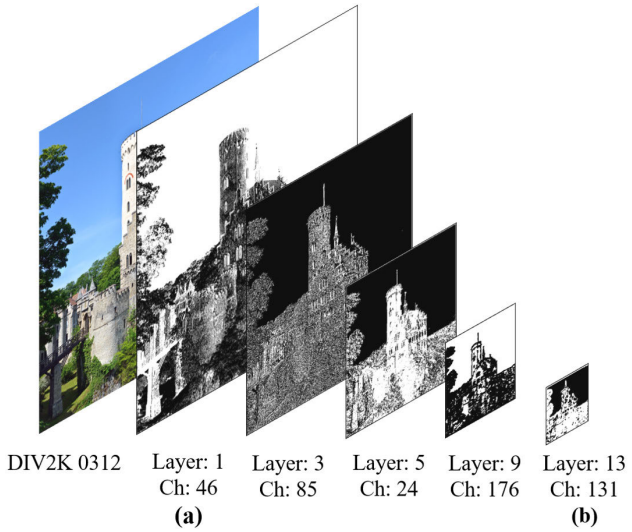

FIGURE 7. Some feature maps extracted from a pre-trained VGG-19 feature extractor of the DIV2K 0312 (layer indices are numbered based on Figure 2). 'Ch' represents the channel index of the layer feature map.

The pixel value of the normalized edge image (g_p) was from 0 to 1. To enhance the low values of g_p, the enhanced edge image (g_{ep}) was computed as follows:

$$g_{ep}(A) = -g_p^2(A) + 2g_p(A). \tag{6}$$

Finally, we defined the ETM as follows:

$$ETM(A) = 2\frac{std(g_{ep}(A))}{mean(g_{ep}(A))}. \tag{7}$$

The proposed ETM can be viewed as the standard deviation of the enhanced edge image normalized by the mean of the enhanced edge image. Texture areas tend to have low ETM values whereas strong edge areas tend to have high ETM values. Figure 4 illustrates randomly cropped 192 × 192 g_{ep} samples and their corresponding ETM values. Thus, the proposed ETM values are proportional to the degrees of texture and edge strength. From the DIV2K dataset [28] that has been widely used to train SR networks, we randomly extracted 61580 patches and computed the ETM values. Most ETM values fell between 1 and 5 (Fig. 5).

Since the ETM was used to select the loss layer (as explained in the next section), the ETM had to have a fixed range. Based on the ETM histogram, we applied clipping operations to propose a normalized ETM_s as follows:

$$ETM_1 = \begin{cases} 0 & ETM < 1 \\ (ETM - 1)/4 & 1 \le ETM \le 5 \\ 1 & 5 < ETM. \end{cases} \tag{8}$$

We also tested three other mapping functions as follows:

$$ETM_2 = \ln(4ETM_1 + 1)/\ln(5). \tag{9}$$

$$ETM_3 = 1/(1 + \exp(-3.454(4ETM_1 - 2))). \tag{10}$$

$$ETM_4 = (\exp(4ETM_1 - 1))/(\exp(4) - 1). \tag{11}$$

Figure 6 shows the three mapping functions of the normalized ETM_s.

The proposed method is somewhat similar to [57] because both methods use the Sobel kernel before computing the loss. However, [57] proposed an edge incoherence loss, which computes the distance between the gradient vector of low-dose CT input images and normal-dose CT target images directly. The proposed method performs the Sobel kernel only on the LR input images, and it utilizes the magnitude of the Sobel kernel for adaptive loss layer selection.

B. ADAPTIVE LOSS LAYER SELECTION USING ETM

Figure 7 shows some feature maps extracted from a pre-trained VGG-19 feature extractor of the DIV2K 0312 dataset. In the shallow layer of high-resolution feature maps, both texture and edge information are preserved (Fig. 7a). Since the max-pooling operation selected the largest pixel value with a block (e.g., 2 × 2), eventually only semantic information remained in the deep layer with low-resolution feature maps (Fig. 7b). In general, it has been reported that better perceptual quality might be obtained when computing the loss with low-resolution feature maps [9]–[13].

Based on this observation, we selected the layer from which we computed the perceptual loss. In this paper, we used the following equation:

$$i_{losslayer} = \lfloor (N_{conv,E} - 1)(1 - ETM_s) + 0.5 \rfloor. \tag{12}$$

where $N_{conv,E}$ represents the number of convolutional layers of differentiable feature extractors and $\lfloor \cdot \rfloor$ is the round-up operator. Since we used VGG-19 as the feature extractor in this paper, $N_{conv,E}$ was 16 (Fig. 2). Using the selected loss layer ($i_{losslayer}$), we computed the perceptual loss:

$$\mathcal{L}_{percep,i_{losslayer}}(\theta_F) = \frac{1}{N}\sum_{i=1}^{N} D(E_{i_{loss\_layer}}(F(\mathrm{LR}_i)), E_{i_{loss\_layer}}(\mathrm{HR}_i)). \tag{13}$$

where $E_{i_{loss\_layer}}(\cdot)$ represents the output of the i_{loss_layer}-th layer of a pre-trained differentiable feature extractor. Algorithm 1 shows the pseudo-code of the proposed method. The proposed computation method of ETM and adaptive loss layer selection have very low computational complexity so there is almost no difference in time complexity compared to existing methods. The method requires only a single first-order differential operation on one channel grayscale LR image patch processed with CUDA. Figure 8 shows some selected loss layers of sample image patches based on ETM_2.

C. TOTAL LOSS COMPUTATION

In order to directly compare the proposed perceptual loss computation with that of ESRGAN [13], we trained our SR networks with the same loss setting of ESRGAN except for the perceptual loss. In other words, the total loss was computed as follows:

$$\mathcal{L}_{total}(\theta_F) = \mathcal{L}_{percep,i_{losslayer}}(\theta_F) + \alpha\mathcal{L}_{adv,Ra}(\theta_F) + \beta\mathcal{L}_{pix}(\theta_F). \tag{14}$$

TABLE 1. Comparison of PSNR performance (bold for the highest performance, underlined for the second).

Test datsets	ESRGAN-DIV2K	ETM_1	ETM_2	ETM_3	ETM_4
BSD100 [32]	25.56	**25.75**	25.57	25.51	25.56
DIV2K [28]	28.01	**28.35**	28.17	28.11	28.17
manga109 [34]	28.30	**29.17**	29.08	28.78	28.44
Set5 [33]	30.34	**30.77**	30.59	30.19	30.45
Set14 [35]	26.32	**26.85**	26.69	26.54	26.67
urban100 [36]	24.64	25.07	**25.11**	24.77	24.74

TABLE 2. Comparison of LPIPS performance (bold for the highest performance, underlined for the second).

Test datsets	ESRGAN-DIV2K	ETM_1	ETM_2	ETM_3	ETM_4
BSD100 [32]	0.1638	0.1584	**0.1555**	0.1588	0.1646
DIV2K [28]	0.1140	**0.1071**	0.1100	0.1134	0.1121
manga109 [34]	0.0736	**0.0578**	0.0609	0.0651	0.0668
Set5 [33]	0.0692	**0.0608**	0.0616	0.0641	0.0643
Set14 [35]	0.1418	0.1408	**0.1283**	0.1393	0.1408
urban100 [36]	0.1224	0.1164	**0.1131**	0.1204	0.1206

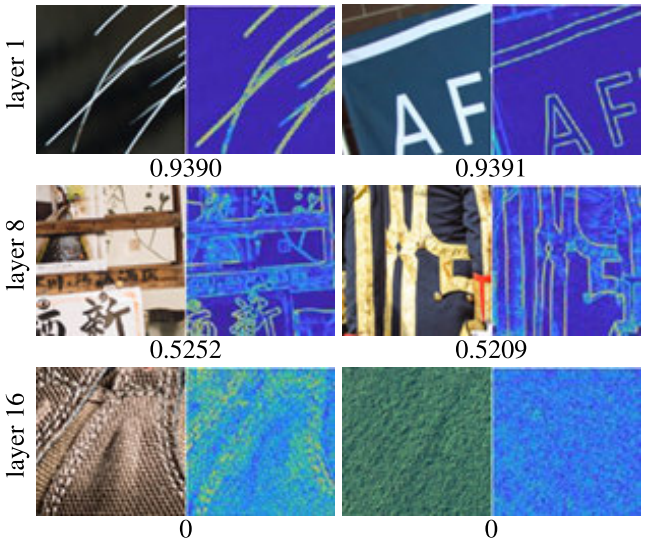

FIGURE 8. Adaptively selected loss layers based on ETM_2 corresponding to sample image patches (left), g_{ep} (right). ETM_2 values are shown below the image patches (layer indices are numbered based on Figure 2).

where $\mathcal{L}_{percep,i_{losslayer}}(\cdot)$ represents the perceptual loss computed by the proposed method based on ETM, $\mathcal{L}_{adv,Ra}(\cdot)$ represents an adversarial loss of the relativistic average discriminator [19] and $\mathcal{L}_{pix}(\cdot)$ represents a pixel loss computed as the distance of the pixel values in the image space. In (13), α was set to 0.005 and β was set to 0.01.

D. PERFORMANCE COMPARISON OF NORMALIZED ETMs

First, we evaluated the performance of the four normalized ETM_s (ETM_{1-4}) using only the DIV2K training dataset [28]. We also trained ESRGAN-DIV2K using only the DIV2K dataset for direct comparison. Tables 1-2 show the PSNR and LPIPS [29] results for quantitative evaluation. Figure 9 shows some of the SR images for qualitative evaluation. It can be seen that ETM_2 showed the best quantitative and qualitative performance. Therefore, we used ETM_2 to train our final model for extensive comparisons with some of the existing SR methods. ETM_2 also shows better reconstruction performance than the ESRGAN-DIV2K method.

Algorithm 1 Perceptual Loss Computation by Adaptive Loss Layer Selection by the Proposed ETMs

Input: LR images X, HR images Y the number of mini-batch size N and grayscale counterpart of LR images Z.

Output: Computed perceptual loss value $\mathcal{L}_{percep}$ of mini-batch by loss layer selection with proposed ETMs.

Sample a mini-batch of images $LR_i \in X, HR_i \in Y, A_i \in Z, i = 1, \ldots, N$.
Initialize loss value variable as, $\mathcal{L}_{i_{losslayer}} = 0$.
for *iteration* $= 1, 2, \ldots, N$ **do**
    Compute the edge magnitude image ($g_{m,i}$) using the Sobel operator as,
    $g_{m,i} = Sobel_{magnitude}(A_i)$
    Compute a normalized edge image $g_{p,i}$ as,
    $g_{p,i} = g_{p,i}/1140.395$
    Generate an enhanced edge map $g_{ep,i}$ for enhancing low values as,
    $g_{ep,i} = -g_{p,i}^2 + 2g_{p,i}$
    Finally, we defined ETM as,
    $ETM = 2std(g_{ep,i})/mean(g_{ep,i})$
    We applied clipping operations for normalizing ETM as,
    if $then < 1$
        $ETM_1 = 0$
    else if $then1 <= t <= 5$
        $ETM_1 = (ETM - 1)/4$
    else if $then5 < t$
        $ETM_1 = 1$
    end if
    We selected ETM_2 for final model training.
    $ETM_2 = \ln(4ETM_1 + 1)/\ln(5)$
    Select loss layer of 16 convolutional layers of VGG feature extractor based on calculated layer index of normalized ETM_2.
    $i_{losslayer} = \lfloor (N_{conv,E} - 1)(1 - ETM_2) + 0.5 \rfloor$
    Calculate the distance between features of selected loss layers, and add to $\mathcal{L}_{i_{losslayer}}$ as,
    $\mathcal{L}_{i_{losslayer}} += D(E_{i_{losslayer}}(F(HR_i)), E_{i_{losslayer}}(F(LR_i)))$.
end for
$\mathcal{L}_{i_{losslayer}} = \mathcal{L}_{i_{losslayer}}/N$

E. GENERATOR ARCHITECTURE

RRDB [13], which is the generator architecture of ESRGAN [13], has been widely used in many recent GAN-based SR networks. In the Ntire 2020 challenge on perceptual extreme super-resolution [25], many teams (including the top two generator architectures) used RRDB [25], [46], [47]. Furthermore, the loss setting and training methods of ESRGAN were widely used as a baseline. Therefore, we used RRDB as a baseline of our generator architecture to present performance comparison results with ESRGAN.

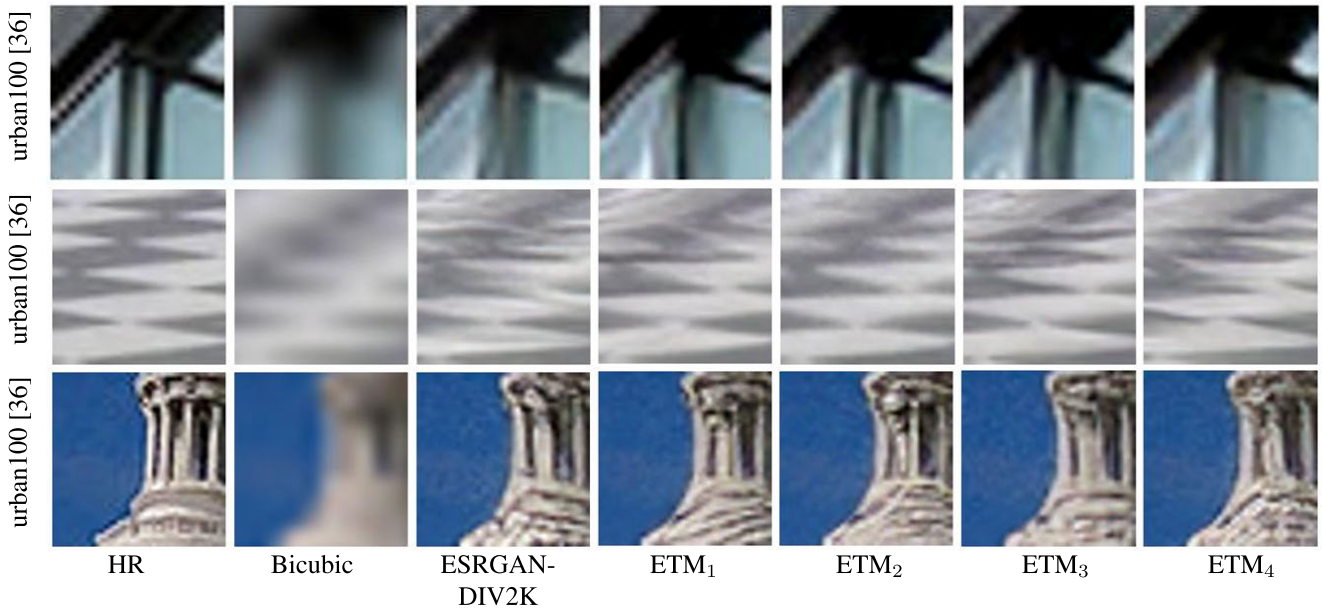

FIGURE 9. SR images reconstructed from GAN-based SR networks trained by various ETM$_s$ and original ESRGAN [13] trained with the DIV2K dataset (scale factor 4, enlarged).

TABLE 3. Performance comparison with recent SR models. (bold for the highest, underlined for the second of LPIPS).

Test datasets	EDSR [5] PSNR/SSIM/LPIPS	RCAN [7] PSNR/SSIM/LPIPS	SAN [8] PSNR/SSIM/LPIPS	ESRGAN [13] PSNR/SSIM/LPIPS	Proposed PSNR/SSIM/LPIPS
BSD100 [32]	27.73 / .7422 / .3627	27.76 / .7463 / .3580	27.78 / .7435 / .3629	25.31 / .6505 / .1621	25.88 / .6755 / **.1565**
DIV2K [28]	30.73 / .8449 / .2564	30.77 / .8460 / .2531	- / - / -	28.17 / .7759 / .1149	28.46 / .7794 / **.1101**
manga109 [34]	31.04 / .9158 / .1006	31.21 / .9172 / .0981	31.23 / .9173 / .1002	28.47 / .8595 / .0646	29.61 / .8731 / **.0564**
Set5 [33]	32.48 / .8987 / .1715	32.62 / .9002 / .1698	32.64 / .9004 / .1729	30.43 / .8516 / .0717	30.81 / .8585 / **.0653**
Set14 [35]	28.81 / .7879 / .2747	28.86 / .7889 / .2741	28.91 / .7889 / .2767	26.29 / .6988 / .1315	26.77 / .7144 / **.1210**
urban100 [36]	26.65 / .8037 / .2039	26.82 / .8087 / .1952	26.83 / .8081 / .2031	24.37 / .7337 / .1235	25.23 / .7589 / **.1134**
Average	29.36 / .8235 / .2414	29.48 / .8253 / .2375	- / - / -	26.99 / .7508 / .1161	27.60 / .7669 / .1089

IV. EXPERIMENTS

Our final model was trained through transfer learning from the RRDB's pre-trained PSNR-oriented parameters that were kindly shared by Wang *et al.* [13].

The final model was trained with the DF2K (DIV2K + Flickr2K [28]) and OST [14] training datasets. Also, we only trained and tested for scale factor 4. We used a MATLAB [30] bicubic kernel to generate the LR images. The spatial size of 128×128 HR patches was randomly cropped and extracted from the HR images and 32×32 LR patches (counterparts of the HR patches) were used as training data.

We set the batch size to 16 and the total number of iterations was 800K. The initial learning rate was 0.0001 and we annealed half at every 100K, 200K, 400K, and 600K iteration. Our model was trained with an Adam optimizer [27] by setting $\beta_1 = 0.9$, $\beta_2 = 0.99$.

Since quantitative evaluation results may differ depending on some changes in the environment, for qualitative evaluation we downloaded the pre-trained network parameters shared by the authors of various recent SR models and computed all of the PSNR, SSIM metrics with LPIPS performance on widely used SR test datasets (BSDS100 [32], DIV2K [28], manga109 [34], Set5 [33], Set14 [35]). We found very small differences in the performance metrics due to the framework SW version differences. We used PyTorch 1.5.0 (Ubuntu 16.04, 18.04) and several computers with different CPUs (Intel Core i7-9700K, AMD Ryzen Threadripper 1900X) and different GPUs (GTX 1080, 2080Ti).

Table 3 shows the quantitative results of the proposed method on public benchmark datasets. Although PSNR-oriented SR networks (EDSR, RCAN, SAN) show higher PSNR and SSIM performance, the proposed method shows better LPIPS performance in all the test datasets. The qualitative results are shown in Figures 10 and 11. Although ESRGAN succeeded in reconstructing the fine details of the SR images, it also generated many unpleasant artifacts that reduced the perceptual image quality. On the other hand, the proposed method produced sharper images than the PSNR-oriented SR networks and it displayed fine details with much fewer unpleasant artifacts than the GAN-based SR networks.

V. CONCLUSION

In this paper, we proposed a new edge texture metric (ETM) to quantify images from texture-like images to edge-dominant images. We used this metric to adaptively select the appropriate layer to calculate the perceptual loss.

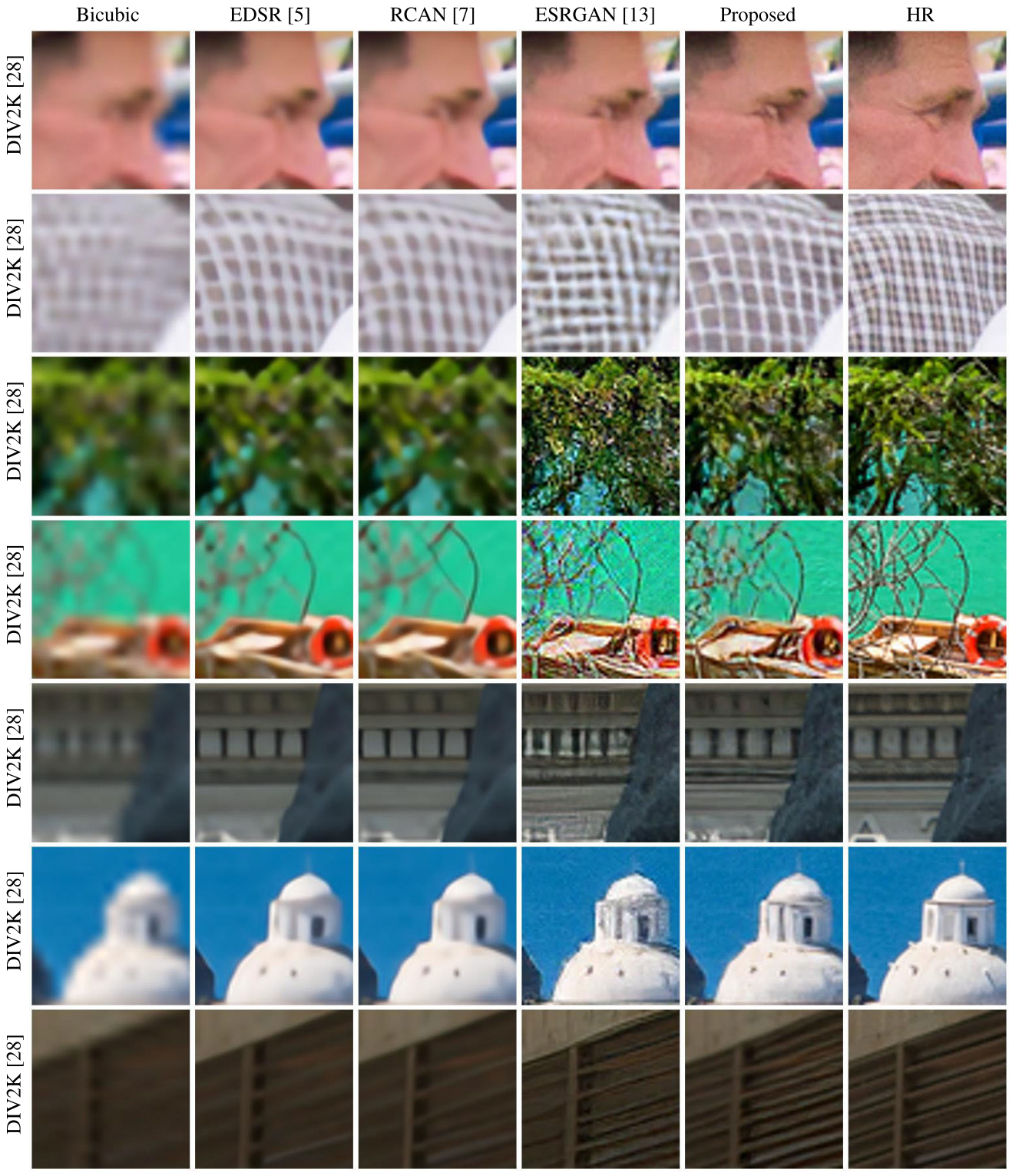

FIGURE 10. SR image results for qualitative evaluation of the proposed method (scale factor 4, enlarged).

Using ETM, we adaptively selected the appropriate layer feature map for calculating the perceptual loss without any other additional prior information (i.e. semantic segmentation priors) when training the GAN-based SR networks.

The need for training-based loss layer selection arose by introducing DCN in the adaptive layer selection method, which is something that remains open to exploration in future work.

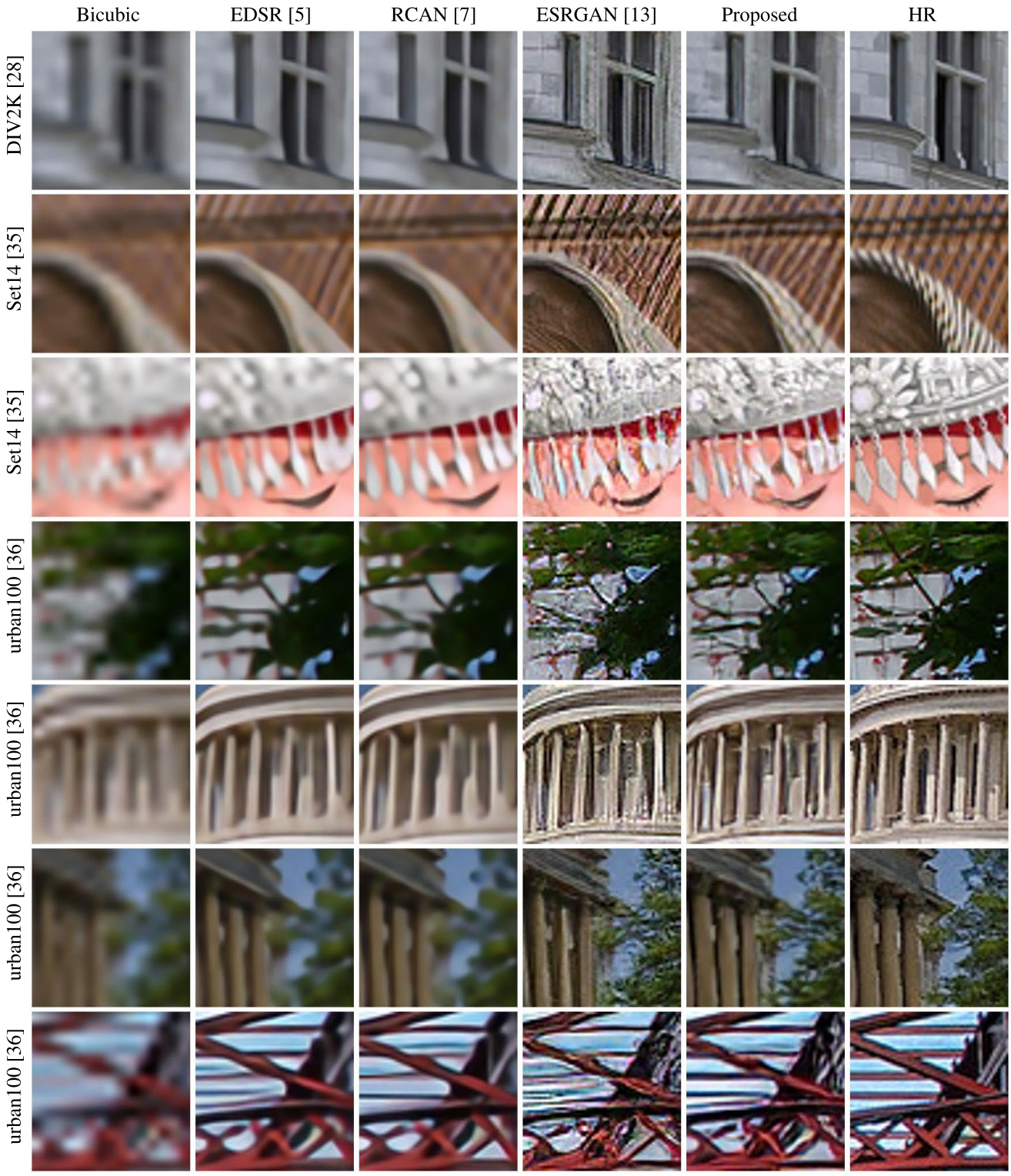

FIGURE 11. SR image results for qualitative evaluation of the proposed method (scale factor 4, enlarged).

Experiments show that GAN-based SR networks trained by the proposed method showed improved performance qualitatively and quantitatively as observed in all aspects of the PSNR, SSIM and LPIPS results and the reconstructed SR images compared to the original ESRGAN, which calculates the perceptual loss at a limited feature level.

REFERENCES

[1] 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. 2015.

[2] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 391–407.

[3] 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.*, Jun. 2016, pp. 1646–1654.
[4] J. Kim, J. K. Lee, and K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1637–1645.
[5] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jul. 2017, pp. 136–144.
[6] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2472–2481.
[7] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 286–301.
[8] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang, "Second-order attention network for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 11065–11074.
[9] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4681–4690.
[10] M. S. M. Sajjadi, B. Scholkopf, and M. Hirsch, "EnhanceNet: Single image super-resolution through automated texture synthesis," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 4491–4500.
[11] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 694–711.
[12] A. Dosovitskiy and T. Brox, "Generating images with perceptual similarity metrics based on deep networks," 2016, *arXiv:1602.02644*. [Online]. Available: http://arxiv.org/abs/1602.02644
[13] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy, "ESRGAN: Enhanced super-resolution generative adversarial networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 1–16.
[14] X. Wang, K. Yu, C. Dong, and C. C. Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 606–615.
[15] M. S. Rad, B. Bozorgtabar, U.-V. Marti, M. Basler, H. K. Ekenel, and J.-P. Thiran, "SROBB: Targeted perceptual loss for single image super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 2710–2719.
[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.
[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," in *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.
[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556
[19] A. Jolicoeur-Martineau, "The relativistic discriminator: A key element missing from standard GAN," 2018, *arXiv:1807.00734*. [Online]. Available: http://arxiv.org/abs/1807.00734
[20] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: http://arxiv.org/abs/1502.03167
[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[22] P. Li, J. Xie, Q. Wang, and W. Zuo, "Is second-order information helpful for large-scale visual recognition?" in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2070–2078.
[23] I. Goodfellow, J. Pouget-Abadie, M. Mizra, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1–9.
[24] Y. Blau, R. Mechrez, R. Timofte, T. Michaeli, and L. Zelnik-Manor, "The 2018 PIRM challenge on perceptual image super-resolution," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 1–22.
[25] K. Zhang, S. Gu, and R. Timofte, "NTIRE 2020 challenge on perceptual extreme super-resolution: Methods and results," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2020, pp. 492–493.
[26] 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.
[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980
[28] E. Agustsson and R. Timofte, "NTIRE 2017 challenge on single image super-resolution: Dataset and study," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jul. 2017, pp. 126–135.
[29] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 586–595.
[30] *MATLAB 9.8.0.1323502 (R2020a)*, MathWorks, Natick, MA, USA, 2020.
[31] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.
[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 *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, Vol. 2, Jul. 2001, pp. 416–423.
[33] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L.-A. Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*. BMVA Press, 2012, pp. 135.1–135.10.
[34] Y. Matsui, K. Ito, Y. Aramaki, A. Fujimoto, T. Ogawa, T. Yamasaki, and K. Aizawa, "Sketch-based Manga retrieval using Manga109 dataset," *Multimedia Tools Appl.*, vol. 76, no. 20, pp. 21811–21838, 2017.
[35] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *Proc. Int. Conf. Curves Surf.* Berlin, Germany: Springer, 2010, pp. 711–730.
[36] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5197–5206.
[37] M. Haris, G. Shakhnarovich, and N. Ukita, "Task-driven super resolution: Object detection in low-resolution images," 2018, *arXiv:1803.11316*. [Online]. Available: http://arxiv.org/abs/1803.11316
[38] Q. Delannoy, C.-H. Pham, C. Cazorla, C. Tor-Díez, G. Dollé, H. Meunier, N. Bednarek, R. Fablet, N. Passat, and F. Rousseau, "SegSRGAN: Super-resolution and segmentation using generative adversarial networks—Application to neonatal brain MRI," *Comput. Biol. Med.*, vol. 120, May 2020, Art. no. 103755.
[39] H. Zhang, J. Yang, Y. Zhang, and T. S. Huang, "Non-local kernel regression for image and video restoration," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2010, pp. 566–579.
[40] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Comput. Graph. Appl.*, vol. 22, no. 2, pp. 56–65, Mar./Apr. 2002.
[41] H. Chang, D.-Y. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, Jun. 2004, p. I.
[42] C. Lee, J. Park, S. Woo, J. Kim, and J. Yoon, "Mathematical analysis of DCN-based super-resolution," *IEEE Access*, vol. 8, pp. 90420–90429, 2020.
[43] F. Yang, H. Yang, J. Fu, H. Lu, and B. Guo, "Learning texture transformer network for image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 5791–5800.
[44] D. Wang, C. Xie, S. Liu, Z. Niu, and W. Zuo, "Image inpainting with edge-guided learnable bidirectional attention maps," 2021, *arXiv:2104.12087*. [Online]. Available: http://arxiv.org/abs/2104.12087
[45] X. Zhou, B. Zhang, T. Zhang, P. Zhang, J. Bao, D. Chen, Z. Zhang, and F. Wen, "CoCosNet v2: Full-resolution correspondence learning for image translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 11465–11475.
[46] T. Shang, Q. Dai, S. Zhu, T. Yang, and Y. Guo, "Perceptual extreme super-resolution network with receptive field block," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2020, pp. 440–441.
[47] Y. Jo, S. Yang, and S. J. Kim, "Investigating loss functions for extreme super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2020, pp. 424–425.
[48] P. Shamsolmoali, M. Zareapoor, E. Granger, H. Zhou, R. Wang, M. E. Celebi, and J. Yang, "Image synthesis with adversarial networks: A comprehensive survey and case studies," *Inf. Fusion*, vol. 72, pp. 126–146, Aug. 2021.

[49] C. Li and M. Wand, "Precomputed real-time texture synthesis with Markovian generative adversarial networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 702–716.
[50] T. Qiao, J. Zhang, D. Xu, and D. Tao, "MirrorGAN: Learning text-to-image generation by redescription," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 1505–1514.
[51] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1316–1324.
[52] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2223–2232.
[53] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8789–8797.
[54] Z. Yi, H. Zhang, P. Tan, and M. Gong, "DualGAN: Unsupervised dual learning for image-to-image translation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2849–2857.
[55] K. Nazeri, E. Ng, T. Joseph, F. Z. Qureshi, and M. Ebrahimi, "EdgeConnect: Generative image inpainting with adversarial edge learning," 2019, *arXiv:1901.00212*. [Online]. Available: http://arxiv.org/abs/1901.00212
[56] Y. Jo and J. Park, "SC-FEGAN: Face editing generative adversarial network with user's sketch and color," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1745–1753.
[57] H. Shan, A. Padole, F. Homayounieh, U. Kruger, R. D. Khera, C. Nitiwarangkul, M. K. Kalra, and G. Wang, "Competitive performance of a modularized deep neural network compared to commercial algorithms for low-dose CT image reconstruction," *Nature Mach. Intell.*, vol. 1, no. 6, pp. 269–276, Jun. 2019.
[58] Y. Chen, L. Liu, J. Tao, R. Xia, Q. Zhang, K. Yang, J. Xiong, and X. Chen, "The improved image inpainting algorithm via encoder and similarity constraint," *Vis. Comput.*, vol. 37, pp. 1691–1705, Jul. 2021.
[59] H. Liu, B. Jiang, Y. Xiao, and C. Yang, "Coherent semantic attention for image inpainting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 4170–4179.
[60] Y. Chen, L. Liu, V. Phonevilay, K. Gu, R. Xia, J. Xie, Q. Zhang, and K. Yang, "Image super-resolution reconstruction based on feature map attention mechanism," *Appl. Intell.*, vol. 51, pp. 4367–4380, Jan. 2021.
[61] K. Zhang, L. Van Gool, and R. Timofte, "Deep unfolding network for image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 3217–3226.
[62] Y. Mei, Y. Fan, and Y. Zhou, "Image super-resolution with non-local sparse attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 3517–3526.

J. KIM received the B.S. degree in electronics and electrical engineering from Dongguk University, Seoul, South Korea, in 2019, and the M.S. degree in electrical and electronics engineering from Yonsei University, Seoul, in 2021. He is currently working with Hyundai Mobis. His research interests include CNN-based super-resolution and dependable deep learning.

C. LEE (Member, IEEE) received the B.S. and M.S. degrees in electronics engineering from Seoul National University, in 1984 and 1986, respectively, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1992. From 1986 to 1987, he was a Researcher with the Acoustics Laboratory, Technical University of Denmark. From 1993 to 1996, he worked with the National Institutes of Health, Bethesda, MD, USA. In 1996, he joined as a Faculty Member with the Department of Electrical and Computer Engineering, Yonsei University, Seoul, South Korea. His research interests include image/signal processing, pattern recognition, and neural networks.