Boosting Flow-based Generative Super-Resolution Models via Learned Prior Supplementary Materials

1. Overview

This supplementary material first provides an expanded notation table used in this work. Following this, more detailed information is presented for enhanced clarity, including additional experimental results and implementation details. Lastly, more qualitative comparisons are shown to demonstrate the capabilities of the proposed framework.

2. Notation Table

Table 1 presents the notations used in the main text and their corresponding definitions.

3. Detailed Analyses

This section analyzes the design flexibility of our latent module in arbitrary-scale super-resolution (SR) tasks and demonstrates the visual effects of employing different objective functions.

Design flexibility in Arbitrary-scale SR tasks. In the main paper, we present the arbitrary-scale SR results of our proposed framework, which employs UNet [12] as the latent generator. To further validate the design flexibility of our latent module, we provide results in arbitrary-scale SR tasks using EDSR-baseline [8] and Swin Transformer [9] (Swin-T) as alternative latent generators. This analysis includes two variants of our model: EDSR-baseline-LINF-LP and RRDB-LINF-LP, which facilitates a comprehensive comparison. For clarity, we denote them as EDSR-baseline-Ours and RRDB-Ours, respectively. The results in Table 2 demonstrate that our framework achieves promising results with all these backbones as the latent generator, proving the flexibility of our framework in arbitrary-scale SR tasks.

Visual Effects with Different Objective Functions. Figure 1 demonstrates the qualitative results obtained by training our latent module using different objective functions. As shown in this figure, our framework generates sharper content when employing the perceptual loss, in contrast to the smooth but blurry results with the latent space loss. In addition, our framework effectively mitigates the grid artifacts presented in image produced by an LINF [14] model.

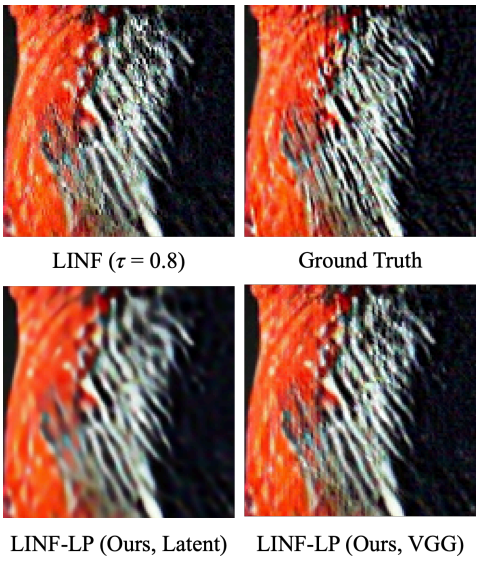

Figure 1. A qualitative comparison between the effects of employing different objective functions. "Latent" represents the exclusive use of latent space loss, while "VGG" denotes the sole employment of VGG perceptual loss [5].

4. Implementation Details

In this section, we illustrate the derivation of the "best temperature map" and additional details of SRFlow-LP.

Best Temperature Map. We adapt the derivation of the "Optimal Objective Estimation" in SROOE [13] to generate our "best temperature map". Specifically, for each image, we generate a total of 21 outcomes using an LINF model, with sampling temperatures ranging from 0 to 1 at intervals of 0.05. Then, for every pixel in each image, we compute the LPIPS [16] values and select the temperature that yields the optimal LPIPS from these 21 images. Once the optimal temperature for every pixel is determined, we assemble these selections to form a "best temperature map". Note that in [13], they search a t value as a conditional input to

Table 1. The notations used in this study and their corresponding definitions.

Notation	Definition
N	The size of a dataset.
x	Low-resolution image.
x^{up}	Bilinear upsampled low-resolution image.
y	High-resolution image.
$\hat{y}$	High-resolution image generated by the flow model.
m	The residual map between y and x_{up}.
z	A latent code in a standard normal distribution (i.e., $z \in \mathcal{N}(0, I)$).
f_θ	Pre-trained flow-based super-resolution model [10, 14], which is parameterized by θ.
τ	The sampling temperature (i.e., the standard deviation of a Gaussian distribution).
s	The scaling factor.
c	The coordinate of a patch [14].
n	The size of a patch [14].
G	The proposed latent module.
$\hat{z}$	The learned prior generated by G.
z^*	The latent code corresponding to the ground truth image y, transformed by f_θ.
$\mathcal{L}_{latent}$	The L1 loss calculated between z^* and $\hat{z}$.
$\mathcal{L}_{percep}$	The perceptual loss [5] calculated between y and $\hat{y}$.
$\mathcal{L}_{total}$	The final objective function.
Ψ_{per}	A pre-trained VGG19 [6] network.

Table 2. The arbitrary-scale SR results on SR benchmark datasets. The names in the parentheses (*e.g.*, UNet) refer to the architecture of our latent generator. "*In-scales*" and "*OOD-scales*" refer to in- and out-of-training-distribution scales. LPIPS [16] scores are reported (lower is better), with the best and second-best highlighted in **red** and blue, respectively.

Method	Set5 [1]					Set14 [15]					B100 [11]					Urban100 [4]				
	In-scales			OOD-scales		In-scales			OOD-scales		In-scales			OOD-scales		In-scales			OOD-scales	
	×2	×3	×4	×6	×8	×2	×3	×4	×6	×8	×2	×3	×4	×6	×8	×2	×3	×4	×6	×8
EDSR-baseline-MetaSR [3]	0.057	0.125	0.175	0.253	0.326	0.094	0.207	0.286	0.395	0.460	0.147	0.285	0.376	0.492	0.565	0.065	0.157	0.233	0.352	0.446
EDSR-baseline-LIIF [2]	0.056	0.124	0.173	0.248	0.307	0.093	0.205	0.284	0.390	0.449	0.147	0.282	0.372	0.486	0.556	0.064	0.155	0.228	0.338	0.422
EDSR-baseline-LTE [7]	0.056	0.123	0.174	0.257	0.326	0.092	0.203	0.283	0.396	0.463	0.146	0.280	0.371	0.495	0.570	0.063	0.152	0.224	0.345	0.436
EDSR-baseline-LINF [14] ($\tau = \tau_0$)	0.035	0.067	0.088	0.158	0.249	0.064	0.115	0.163	0.275	0.375	0.108	0.172	0.207	0.319	0.451	0.050	0.110	0.158	0.273	0.386
EDSR-baseline-Ours (EDSR-baseline)	**0.026**	0.051	**0.072**	0.157	0.276	**0.052**	0.097	0.147	0.269	0.390	**0.083**	0.132	0.177	0.304	0.440	**0.043**	**0.097**	0.144	0.260	0.388
EDSR-baseline-Ours (UNet)	**0.026**	**0.047**	0.074	**0.145**	**0.243**	0.054	**0.094**	0.144	0.253	**0.364**	0.084	**0.127**	0.177	0.289	0.425	0.440	0.098	0.146	0.253	**0.377**
EDSR-baseline-Ours (Swin-T)	0.029	0.053	0.073	0.149	0.277	0.055	0.100	**0.141**	**0.252**	0.379	0.090	0.135	**0.176**	**0.287**	**0.422**	**0.043**	**0.097**	**0.141**	**0.251**	0.382
RRDB-LINF [14] ($\tau = \tau_0$)	0.034	0.064	0.084	0.147	0.247	0.059	0.110	0.146	0.252	0.359	0.097	0.152	0.194	0.306	0.444	0.040	0.093	0.137	0.239	0.354
RRDB-Ours (EDSR-baseline)	0.025	0.044	0.066	0.136	0.249	0.046	0.087	0.129	0.230	0.329	0.069	0.118	0.165	0.272	0.385	0.035	0.082	0.126	0.230	0.352
RRDB-Ours (Unet)	**0.023**	**0.042**	0.066	0.131	**0.234**	**0.043**	0.087	0.124	**0.221**	**0.322**	**0.061**	**0.113**	0.163	**0.264**	**0.378**	**0.033**	0.081	0.126	**0.219**	**0.331**
RRDB-Ours (Swin-T)	0.025	0.045	**0.063**	**0.129**	0.247	0.046	**0.086**	**0.123**	0.223	0.328	0.071	0.115	**0.162**	0.267	0.383	0.034	**0.080**	**0.123**	0.221	0.343

their GAN model, while in our approach, the temperature τ represents the standard deviation of a Gaussian distribution.

SRFlow-LP Implementation Details. We found that employing the latent space loss as a regularization term for training SRFlow-LP effectively prevents exploding inverses. This effectiveness stems from the observation that models trained with the latent space loss tend to produce latent codes that fall within the training distribution, therefore avoiding subsequent exploding inverses. To further stabilize the inference process of SRFlow-LP, the initial prior is normalized before being processed by the latent module. In addition, we skip the iteration which encounters an exploding inverse during training. These techniques allow SRFlow-LP to be more stable during both training and inference without modifying the architecture or inference pipeline of the proposed framework.

5. Additional Qualitative Results

5.1. Qualitative Results of SRFlow-LP

Figs. 2 and 3 demonstrate our SRFlow-LP generates images with sharper details than the original SRFlow [10]. Fig. 4 also presents that SRFlow-LP effectively prevents SRFlow from encountering exploding inverses that display noisy patches within images.

5.2. Qualitative Results of LINF-LP

Fig. 5 illustrates that our LINF-LP mitigates the grid artifacts in images generated by LINF [14], especially in areas with thin, repetitive linear structures.

References

[1] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. Alberi Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *Proc. British Machine Vision Conf. (BMVC)*, pages 1–10, 2012. 2

[2] Y. Chen, S. Liu, and X. Wang. Learning continuous image representation with local implicit image function. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 8628–8638, 2021. 2

[3] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, and J. Sun. Meta-sr: A magnification-arbitrary network for super-resolution. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1575–1584, 2019. 2

[4] J.-B. Huang, A. Singh, and N. Ahuja. Single image super-resolution from transformed self-exemplars. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 5197–5206, 2015. 2

[5] J. Johnson, A. Alahi, and F.-F. Li. Perceptual losses for real-time style transfer and super-resolution. In *Proc. European Conf. on Computer Vision (ECCV)*, pages 694–711, 2016. 1, 2

[6] S. Karen and Z. Andrew. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[7] J. Lee and K. H. Jin. Local texture estimator for implicit representation function. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1929–1938, 2022. 2

[8] 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. on Computer Vision and Pattern Recognition Workshop (CVPRW)*, pages 1132–1140, 2017. 1

[9] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proc. IEEE Int. Conf. on Computer Vision (ICCV)*, pages 10012–10022, 2021. 1

[10] A. Lugmayr, M. Danelljan, L. Van Gool, and R. Timofte. Srflow: Learning the super-resolution space with normalizing flow. In *Proc. European Conf. on Computer Vision (ECCV)*, 2020. 2, 3, 4, 5, 6

[11] 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. IEEE Int. Conf. on Computer Vision (ICCV)*, pages 416–425, 2001. 2

[12] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, 2015. 1

[13] Park S.-H., Moon Y.-S., and Cho N.-I. Perception-oriented single image super-resolution using optimal objective estimation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1725–1735, 2023. 1

[14] J.-E. Yao, L.-Y. Tsao, Y.-C. Lo, R. Tseng, C.-C. Chang, and C.-Y. Lee. Local implicit normalizing flow for arbitrary-scale image super-resolution. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1776–1785, 2023. 1, 2, 3, 6

[15] R. Zeyde, M. Elad, and M. Protter. On single image scale-up using sparse-representations. In *Curves and Surfaces*, pages 711–730, 2010. 2

[16] 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. on Computer Vision and Pattern Recognition (CVPR)*, pages 586–595, 2018. 1, 2

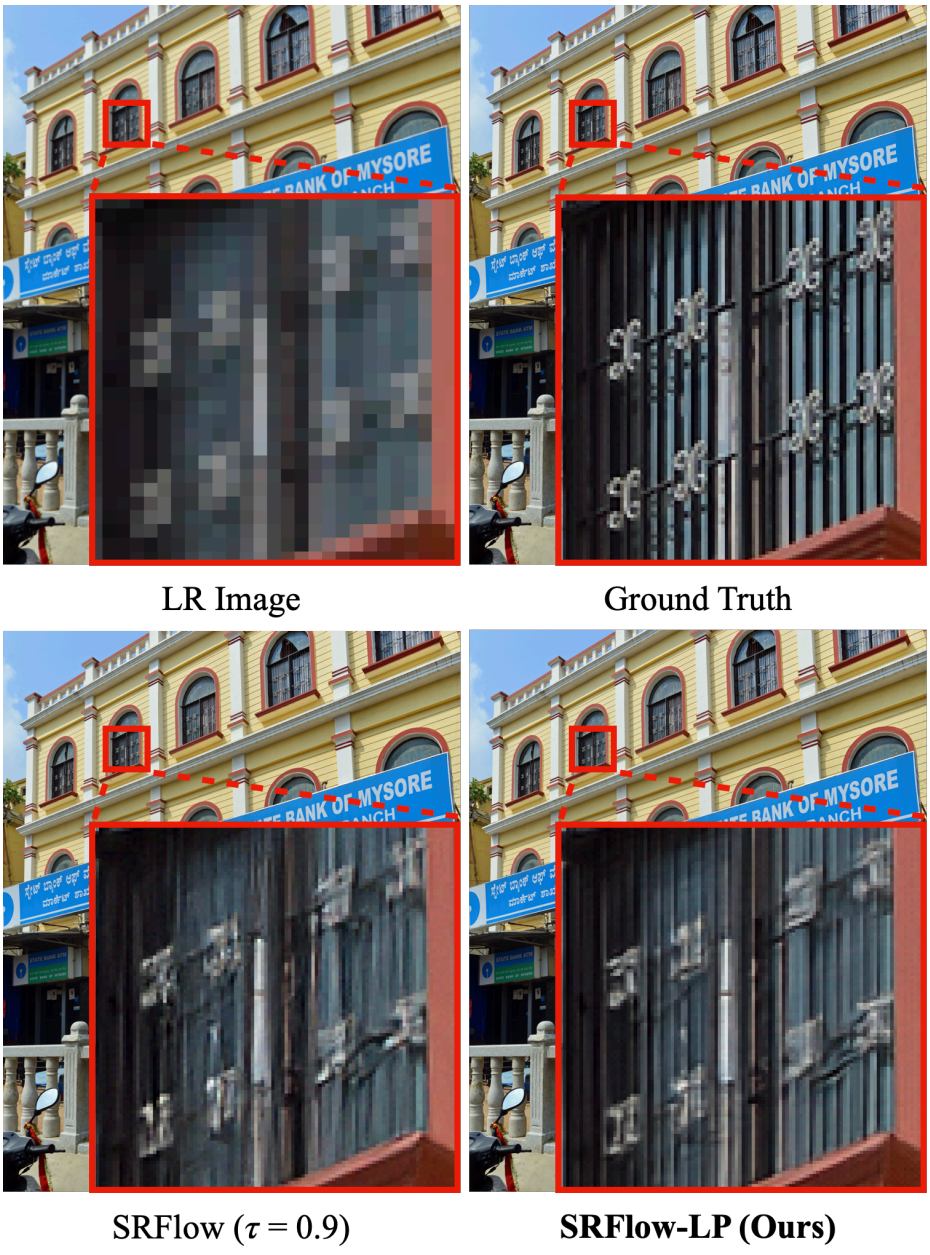

Figure 2. A qualitative comparison between the 4× SR results of SRFlow [10] and our SRFlow-LP.

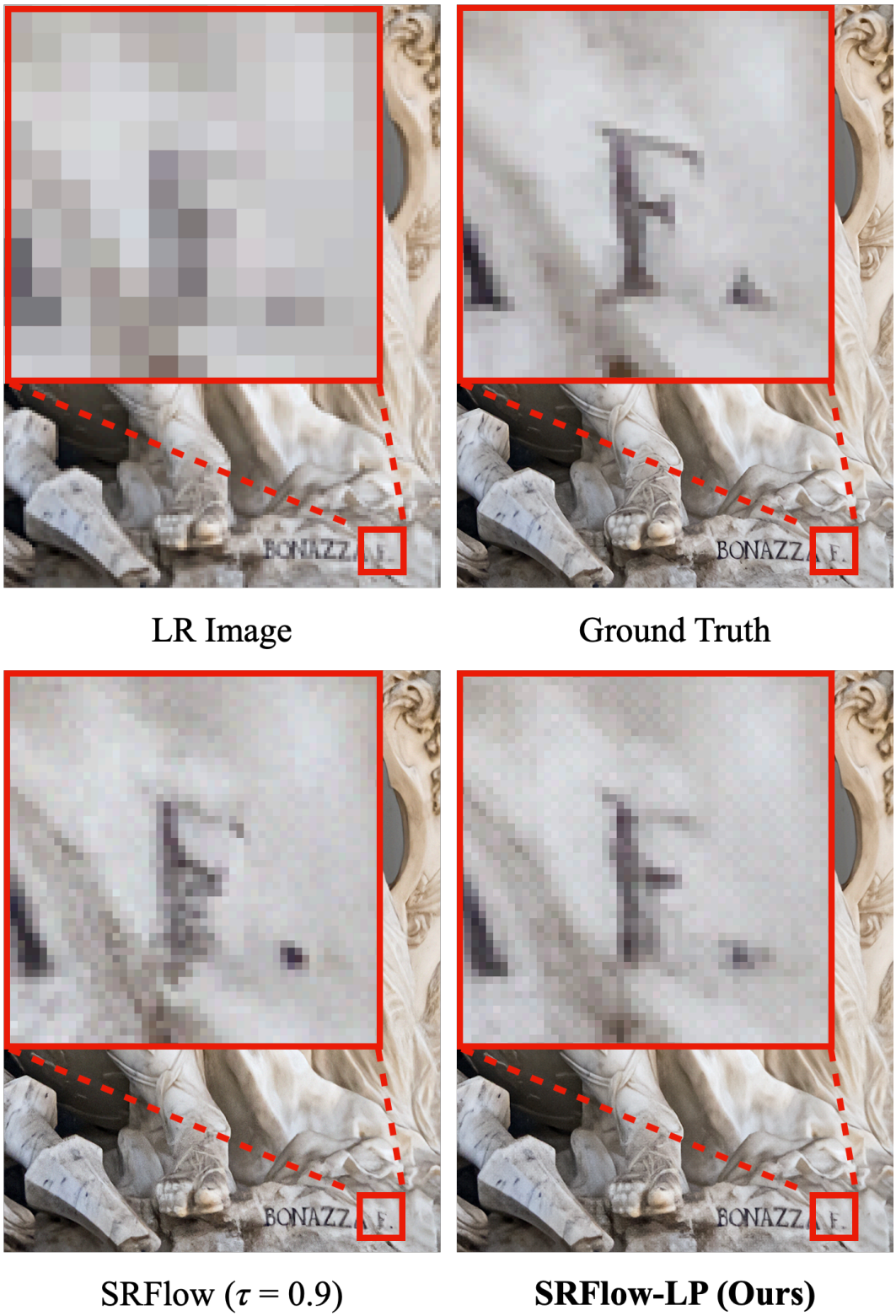

Figure 3. A qualitative comparison between the 4× SR results of SRFlow [10] and our SRFlow-LP.

Figure 4. Our framework SRFlow-LP effectively prevents SRFlow [10] from encountering exploding inverses.

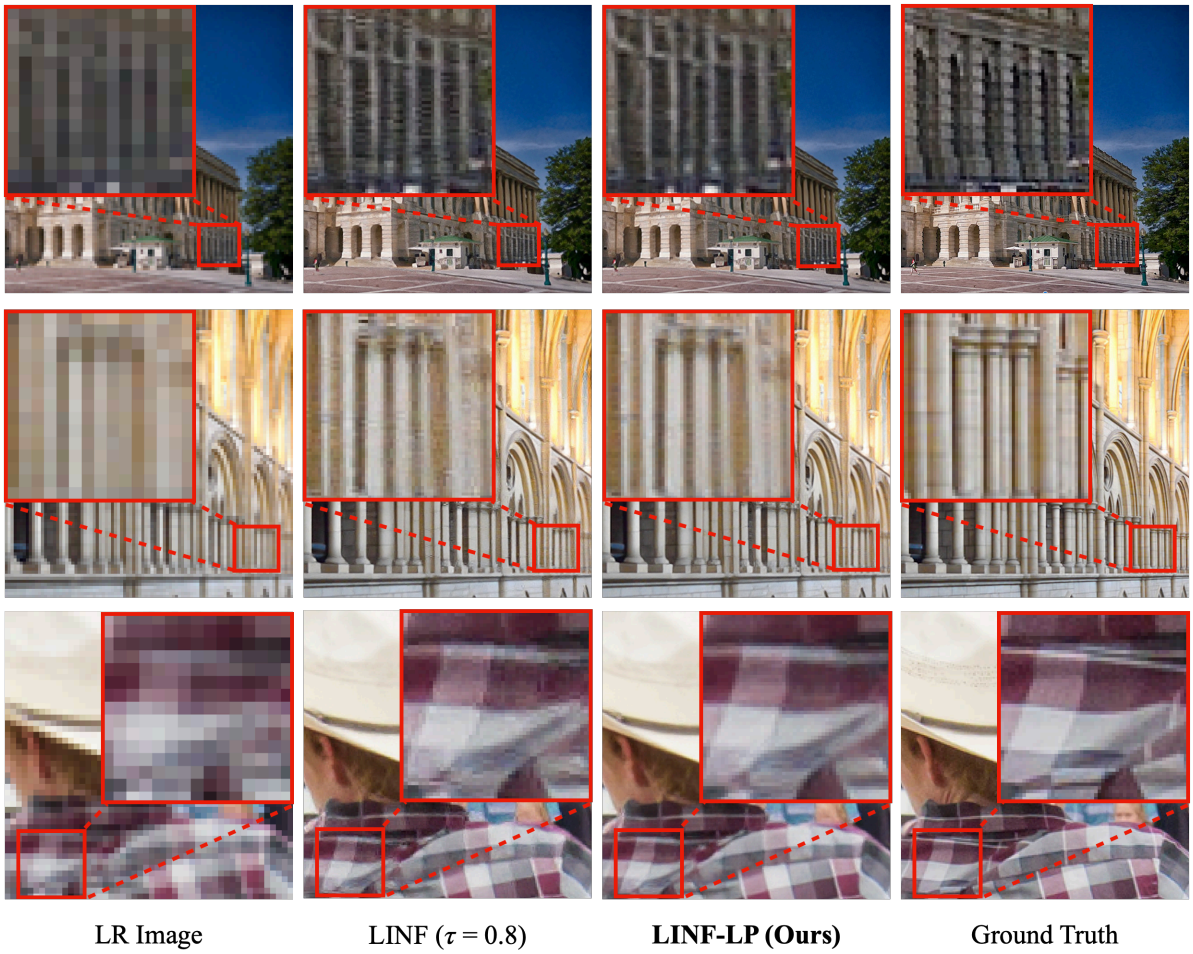

Figure 5. A qualitative comparison between images generated by LINF [14] and LINF-LP.