Supplementary File for DiffBIR: Toward Blind Image Restoration with Generative Diffusion Prior

Xinqi Lin[1,2,⋆], Jingwen He[3,4,⋆], Ziyan Chen[1,2,3], Zhaoyang Lyu[3], Bo Dai[3], Fanghua Yu[1], Yu Qiao[3], Wanli Ouyang[3,4], and Chao Dong[1,3,5,†]

[1]Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences
[2]University of Chinese Academy of Sciences [3]Shanghai AI Laboratory
[4]The Chinese University of Hong Kong
[5]Shenzhen University of Advanced Technology

Abstract. In section 1, we compare IRControlNet with more variants to prove its effectiveness for IR tasks. Then, we provide more details about the restoration module and the restoration guidance in section 2, 3, respectively. In section 4, we present more quantitative and qualitative comparisons for BSR task on synthetic datasets. We also provide a comparison regarding inference speed and model complexity among different BSR methods in section 5. Finally, we provide additional visual results in different real-world scenarios in section 6.

1 Comparison of IRControlNet and More Variants

More Variants for IRControlNet. For more comprehensive analysis, we construct another two variants. The architecture is illustrated in Fig. 1.

Variant 5. Regarding feature modulation, we simultaneously control the middle block features, decoder features and skipped features. We use concat features for simplified denotation.

Variant 6. Regarding feature modulation, we use SFT layer [11] to modulate the intermediate features. Specifically as follows:

$$SFT(\mathbf{F}|\boldsymbol{\gamma},\boldsymbol{\beta}) = \mathbf{F} \odot (1+\boldsymbol{\gamma}) + \boldsymbol{\beta} \tag{1}$$

where $\mathbf{F}$ denotes feature maps, $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ denotes the element-wise scale and shift transformation. Both $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are produced by zero-conv, thus they are initialized to zero at the beginning of training.

Table 1 presents the quantitative results. We can observe that both Variant 5 and 6 achieve better performance in terms of PSNR and SSIM while their MANIQA scores are worse than IRControlNet. Variant 5 applies more control on the pretrained model, which enhances the fidelity but damages generation quality. As for Variant 6, it utilizes SFT layer to modulate the skipped features. As SFT layer brings more precise control, which also improves the fidelity. In

⋆ Equal contribution † Corresponding author

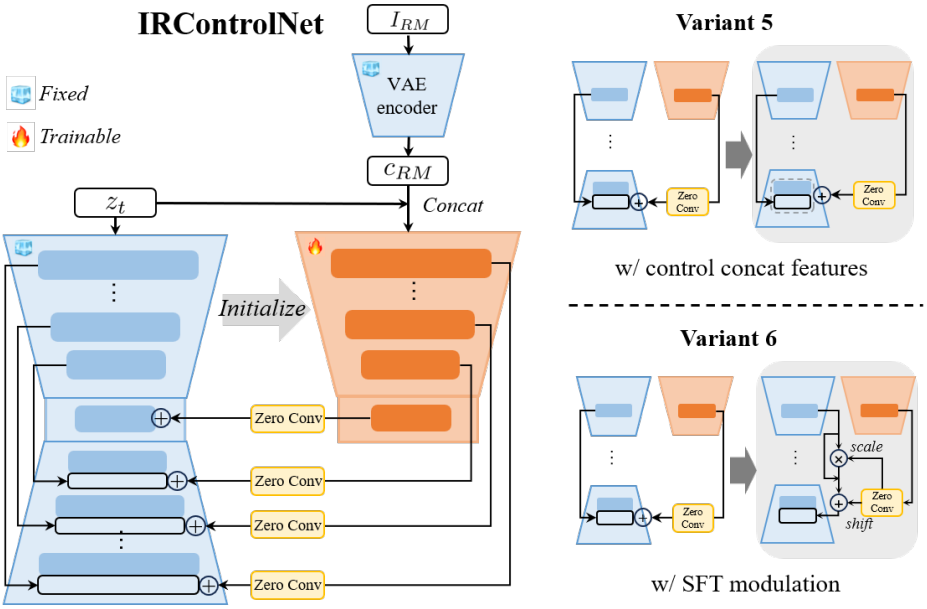

Fig. 1: Architectures of our IRControlNet and two model variants.

conclusion, both Variant 5,6 trade the quality for fidelity. IRControlNet achieves such a trade-off through restoration guidance and utilizes the add-on control to preserve most of the generation capability.

Variants	PSNR↑	SSIM↑	MANIQA↑
IRControlNet	22.9865	0.5200	0.2689
Variant 5: w/ control concat features	23.0449	0.5261	0.2567
Variant 6: w/ SFT modulation	22.9974	0.5292	0.2622

Table 1: Quantitative comparisons of IRControlNet, Variant 5 and 6 on ImageNet1k-Val with Real-ESRGAN [10] degradation.

Qualitative Comparisons for Variant 2. We present the visual comparisons for Variant 2 in Fig. 2. It can be observed that IRControlNet can generate more vivid textures while Variant 2 tends to produce over-smoothed results.

2 More Details of Training RM

During the training of generation module, we follow [14] and modify a widely-used IR backbone, SwinIR [5], as our restoration module. Specifically, we utilize the pixel unshuffle [8] operation to downsample the original low-quality input

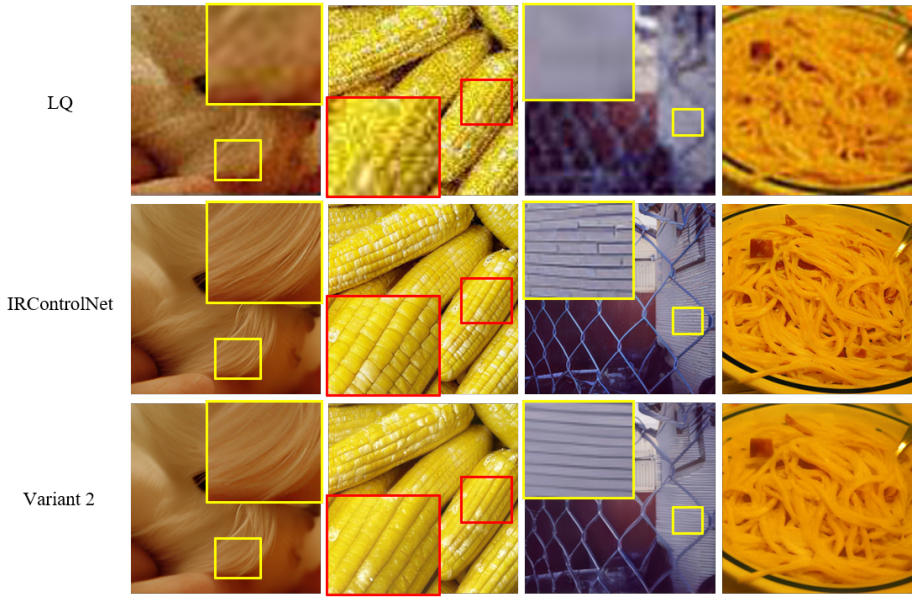

Fig. 2: Visual comparisons of Variant 2 and IRControlNet.

I_{lq} with a scale factor of 8. For upsampling the deep features back to the original image space, we perform the nearest interpolation three times, and each interpolation is followed by one convolutional layer as well as one Leaky ReLU activation layer. This modified SwinIR will be trained on synthetic LQ-HQ image pairs. Here we adopt a classic first-order degradation model to synthesize the LQ images.

$$I_{lq} = \{[(I_{hq} \otimes k_\sigma)_{\downarrow_r} + n_\delta]_{\texttt{JPEG}_q}\}_{\uparrow_r}, \tag{2}$$

where the HQ image I_{hq} is first convolved with a Gaussian kernel k_σ, followed by a downsampling of scale r. After that, additive Gaussian noise n_δ is added to the images, and then JPEG compression with quality factor q is applied. Finally, the LQ image is resized back to the original size. Note that the downsampling and blurring contribute most to the information loss, thus we expand the degradation ranges of these two operations. Specifically, we randomly sample σ, r, δ and q from $\{0.1{:}12\}$, $\{1{:}12\}$, $\{0{:}15\}$, $\{30{:}100\}$, respectively.

3 More Details about Restoration Guidance

We provide a detailed explanation for our proposed restoration guidance in this section. Restoration guidance aims to achieve a trade-off between *quality* and *fidelity* through guiding the denoising process towards the high-fidelity I_{RM} obtained in the first stage. At time t, the UNet denoiser ϵ_θ first predicts the

noise ϵ_t of the noisy latent z_t. Then the predicted noise ϵ_t is removed from z_t to obatin the clean latent $\tilde{z}_0$ through the following equations:

$$\epsilon_t = \epsilon_\theta(z_t, c, t, c_{RM}), \tag{3}$$

$$\tilde{z}_0 = \frac{z_t - \sqrt{1-\bar{\alpha}_t}\epsilon_t}{\sqrt{\bar{\alpha}_t}}. \tag{4}$$

This indicates that we could modify the clean latent $\tilde{z}_0$ in each time step, and then sample z_{t-1} according to the predefined distribution $q(z_{t-1}|z_t, \tilde{z}_0)$. In this way, we are able to achieve preferred restoration results without additional training. To modify $\tilde{z}_0$, we define a region-adaptive MSE loss in image space:

$$\mathcal{L}(\tilde{z}_0) = \frac{1}{HWC}||\mathcal{W} \odot (\mathcal{D}(\tilde{z}_0) - I_{RM})||_2^2, \tag{5}$$

$$\mathcal{W} = 1 - \mathcal{G}(I_{RM}), \tag{6}$$

where H, W, C denotes the spatial size of I_{RM}, and $\mathcal{W}$ is a weight map. $\mathcal{G}(I_{RM})$ is the normalized gradient magnitude of I_{RM}, which represents the gradient intensity of each pixel in I_{RM}. To obtain $\mathcal{G}(I_{RM})$, we first calculate the gradient magnitude for each pixel in I_{RM}:

$$M(I_{RM}) = \sqrt{G_x(I_{RM})^2 + G_y(I_{RM})^2} \tag{7}$$

where G_x and G_y denotes the sobel operator in x and y axis, respectively. As pixels with strong gradient signals are very rare in an image, we then use patch-level gradient signals for better estimate the gradient intensity. We divide I_{RM} into multiple equal-sized non-overlapping patches as follows:

$$\begin{aligned} &\{I_{RM}^{(1)}, I_{RM}^{(2)}, ..., I_{RM}^{(k)}, ...\} \\ &\forall i, j, I_{RM}^{(i)} \cap I_{RM}^{(j)} = \emptyset, \bigcup_i I_{RM}^{(i)} = I_{RM} \end{aligned} \tag{8}$$

For patch $I_{RM}^{(k)}$, we calculate the sum of the gradient magnitudes of all pixels, and use the tanh function to map them into the range of $[0, 1)$:

$$S(I_{RM}^{(k)}) = \tanh\left(\sum_{i,j} M_{i,j}(I_{RM})\right), (i,j) \in I_{RM}^{(k)} \tag{9}$$

where (i, j) denotes a pixel in patch $I_{RM}^{(k)}$. As $S(I_{RM}^{(k)})$ is closer to 1, the corresponding gradient signal is stronger, and vice versa. The final gradient magnitude can be formulated as below:

$$\mathcal{G}_{i,j}(I_{RM}) = \sum_k \mathbb{I}\left[(i,j) \in I_{RM}^{(k)}\right] S(I_{RM}^{(k)}), \tag{10}$$

where $\mathbb{I}\left[(i,j) \in P^{(k)}\right]$ is an indicator function, denoting whether the pixel (i, j) is located in the patch $I_{RM}^{(k)}$. The whole algorithm is illustrated in Algorithm 1.

Algorithm 1 Restoration guidance, given a diffusion model ϵ_θ, and the VAE's encoder $\mathcal{E}$ and decoder $\mathcal{D}$

Input: Guidance image I_{RM}, text description c (set to empty), diffusion steps T, gradient scale s
Output: Output image $\mathcal{D}(z_0)$
Sample z_T from $\mathcal{N}(0, \mathbf{I})$
for t from T to 1 **do**
  $\tilde{z}_0 \leftarrow \frac{z_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(z_t, c, t, \mathcal{E}(I_{RM}))}{\sqrt{\bar{\alpha}_t}}$
  $\mathcal{W} = 1 - \mathcal{G}(I_{RM})$
  $\mathcal{L}(\tilde{z}_0) = \frac{1}{HWC} ||\mathcal{W} \odot (\mathcal{D}(\tilde{z}_0) - I_{RM})||_2^2$
  Sample z_{t-1} from $q(z_{t-1}|z_t, \tilde{z}_0 - s\nabla_{\tilde{z}_0}\mathcal{L}(\tilde{z}_0))$
end for
return $\mathcal{D}(z_0)$

4 More Quantitative and Qualitative Comparisons for BSR on Synthetic Datasets

The quantitative results on DRealSR [12] and RealSR [2] are presented in Table 2. The comparisons on these two datasets lead to similar observations. When the guidance scale s is set to 0, DiffBIR significantly outperforms baseline methods in terms of all IQA metrics. When the guidance scale s is set to 1, DiffBIR still surpasses the baseline methods in MANIQA and CLIP-IQA. As for evaluation in PSNR, DiffBIR performs better than diffusion-based methods and shows comparable performance to GAN-based methods, indicating that DiffBIR can achieve a good balance between *quality* and *fidelity*. Visual comparisons on DIV2K-Val [1] are presented in Figure 3. We can observe that only DiffBIR is able to produce restored results with correct semantic information. For example, it correctly recovers details such as the eyes behind the helmet, the lines of fireworks, and the wings of the penguin. GAN-based methods shows a lack of generation capability, thus producing over-smoothed results. In comparison, diffusion-based baseline methods are usually affected by the severe degradation and fail to generate correct semantics.

5 Quantitative Comparisons for Model Efficiency

We present a quantitative comparison regarding inference speed and model complexity for both diffusion-based and GAN-based methods in Table 3. This comparison is performed on a super-resolution task with an input size of 128×128 and a scale factor of 4. We conduct multiple inferences and calculate the average inference time. It can be observed that DiffBIR is the most efficient among DM-based baselines. It's about 1.8x faster than StableSR and about 1.6x faster than PASD. Although GAN-based methods are more efficient, they perform significantly worse than DM-based methods. The development of diffusion models

Datasets	Metrics	FeMaSR [3]	DASR [4]	Real-ESRGAN+ [10]	BSRGAN [15]	SwinIR-GAN [5]	StableSR [9]	PASD [13]	DiffBIR (s=0)	DiffBIR (s=0.5)	DiffBIR (s=1)
DRealSR [12]	PSNR↑	23.1977	26.3844	24.6878	25.6903	25.3898	23.8669	24.8735	24.2037	24.9891	25.6238
	SSIM↑	0.6239	0.7271	0.6705	0.6765	0.6962	0.6400	0.6529	0.5874	0.6246	0.6544
	LPIPS↓	0.2190	0.1793	0.2290	0.2308	0.2057	0.2355	0.2016	0.2448	0.2328	0.2350
	MUSIQ↑	68.7458	66.0651	67.4608	68.9388	68.1393	69.2621	70.7670	72.3514	71.5339	69.8821
	MANIQA↑	0.3073	0.2048	0.2315	0.2309	0.2375	0.2565	0.2889	0.3915	0.3847	0.3530
	CLIP-IQA↑	0.6327	0.5086	0.5022	0.5328	0.5244	0.5988	0.6151	0.6878	0.6761	0.6440
RealSR [2]	PSNR↑	23.1627	25.5503	24.2400	24.9717	24.6244	23.5627	24.5385	23.5237	24.2216	24.7531
	SSIM↑	0.6534	0.7183	0.6793	0.6839	0.7051	0.6549	0.6694	0.5989	0.6346	0.6615
	LPIPS↓	0.2520	0.2397	0.2556	0.2545	0.2340	0.2429	0.2317	0.2646	0.2544	0.2565
	MUSIQ↑	66.1208	59.5565	66.7333	68.0673	67.0964	68.4594	70.0043	72.3909	71.3969	69.5167
	MANIQA↑	0.2652	0.1713	0.2243	0.2329	0.2281	0.2407	0.2746	0.3820	0.3792	0.3504
	CLIP-IQA↑	0.5925	0.4300	0.4787	0.5233	0.4920	0.5852	0.5822	0.6868	0.6817	0.6478

Table 2: Quantitative comparisons on synthetic datasets (DRealSR [12] and RealSR [2]) for BSR task. **Red** and blue indicate the best and second best performance. The top 3 results are marked as gray.

is extremely fast. There're works [6,7] that can already achieve satisfactory generation performance with only 1∼4 steps, thus the time-consuming problem can be solved in the future.

Metrics	Real-ESRGAN+ [10]	BSRGAN [15]	SwinIR-GAN [5]	FeMaSR [3]	DASR [4]	StableSR [9]	PASD [13]	DiffBIR
Inference Time (ms)	46.19	46.42	126.44	89.01	12.69	19278.46	16951.08	10906.51
Model Size (M)	16.69	16.69	11.71	34.05	8.06	1409.11	1675.76	1716.7

Table 3: Quantitative comparisons of inference speed and model complexity.

6 More Visual Comparisons on Real-world Datasets

We provide more visual comparisons for BSR, BID, BFR tasks in Figure 4, Figure 5 and Figure 6, respectively.

References

1. Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: Dataset and study. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 126–135 (2017)
2. Cai, J., Zeng, H., Yong, H., Cao, Z., Zhang, L.: Toward real-world single image super-resolution: A new benchmark and a new model. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3086–3095 (2019)
3. Chen, C., Shi, X., Qin, Y., Li, X., Han, X., Yang, T., Guo, S.: Real-world blind super-resolution via feature matching with implicit high-resolution priors. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 1329–1338 (2022)
4. Liang, J., Zeng, H., Zhang, L.: Efficient and degradation-adaptive network for real-world image super-resolution. In: European Conference on Computer Vision. pp. 574–591. Springer (2022)

5. Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Swinir: Image restoration using swin transformer. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1833–1844 (2021)
6. Luo, S., Tan, Y., Huang, L., Li, J., Zhao, H.: Latent consistency models: Synthesizing high-resolution images with few-step inference. arXiv preprint arXiv:2310.04378 (2023)
7. Sauer, A., Lorenz, D., Blattmann, A., Rombach, R.: Adversarial diffusion distillation. arXiv preprint arXiv:2311.17042 (2023)
8. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1874–1883 (2016)
9. Wang, J., Yue, Z., Zhou, S., Chan, K.C., Loy, C.C.: Exploiting diffusion prior for real-world image super-resolution. arXiv preprint arXiv:2305.07015 (2023)
10. Wang, X., Xie, L., Dong, C., Shan, Y.: Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1905–1914 (2021)
11. Wang, X., Yu, K., Dong, C., Loy, C.C.: Recovering realistic texture in image super-resolution by deep spatial feature transform. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 606–615 (2018)
12. Wei, P., Xie, Z., Lu, H., Zhan, Z., Ye, Q., Zuo, W., Lin, L.: Component divide-and-conquer for real-world image super-resolution. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16. pp. 101–117. Springer (2020)
13. Yang, T., Ren, P., Xie, X., Zhang, L.: Pixel-aware stable diffusion for realistic image super-resolution and personalized stylization. arXiv preprint arXiv:2308.14469 (2023)
14. Yue, Z., Loy, C.C.: Diffface: Blind face restoration with diffused error contraction. arXiv preprint arXiv:2212.06512 (2022)
15. Zhang, K., Liang, J., Van Gool, L., Timofte, R.: Designing a practical degradation model for deep blind image super-resolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4791–4800 (2021)

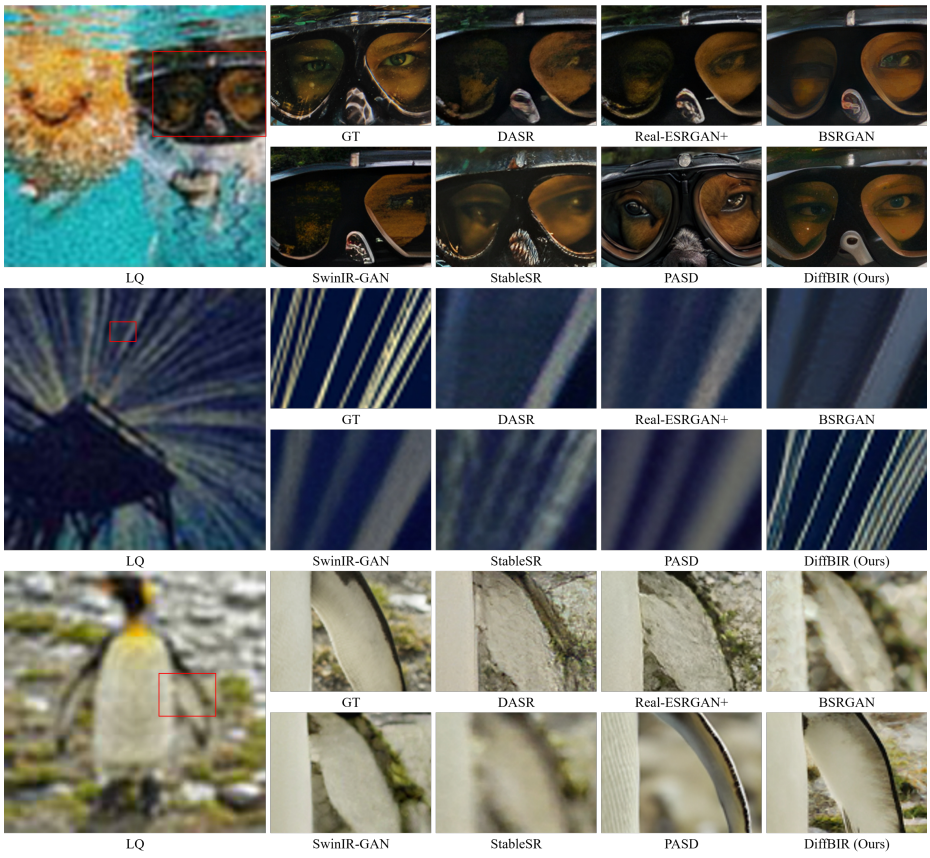

Fig. 3: Visual comparisons of BSR methods on synthetic dataset (DIV2K-Val [1]).

Fig. 4: More visual comparisons for BSR on real-world datasets.

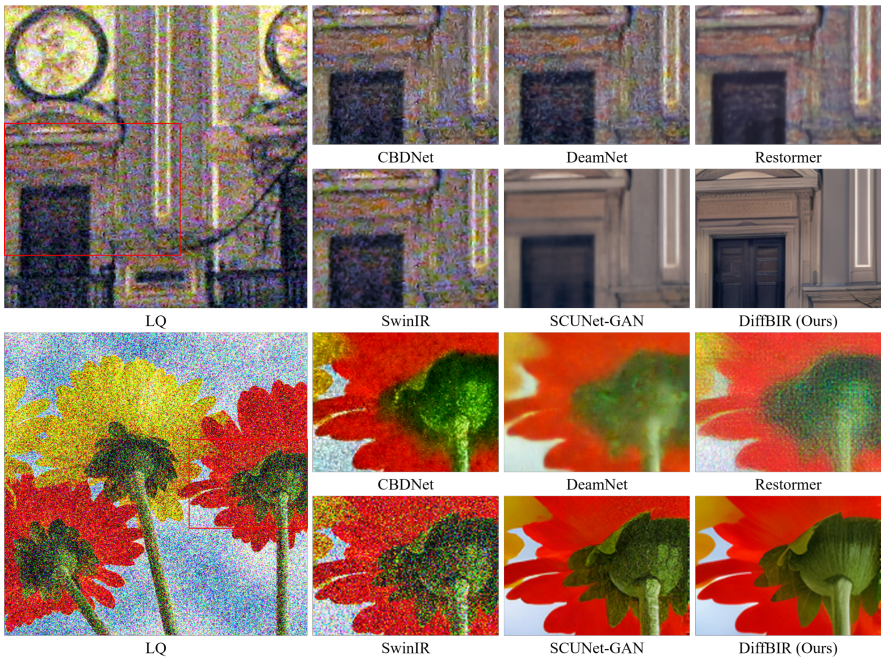

Fig. 5: More visual comparisons for BID on real-world datasets.

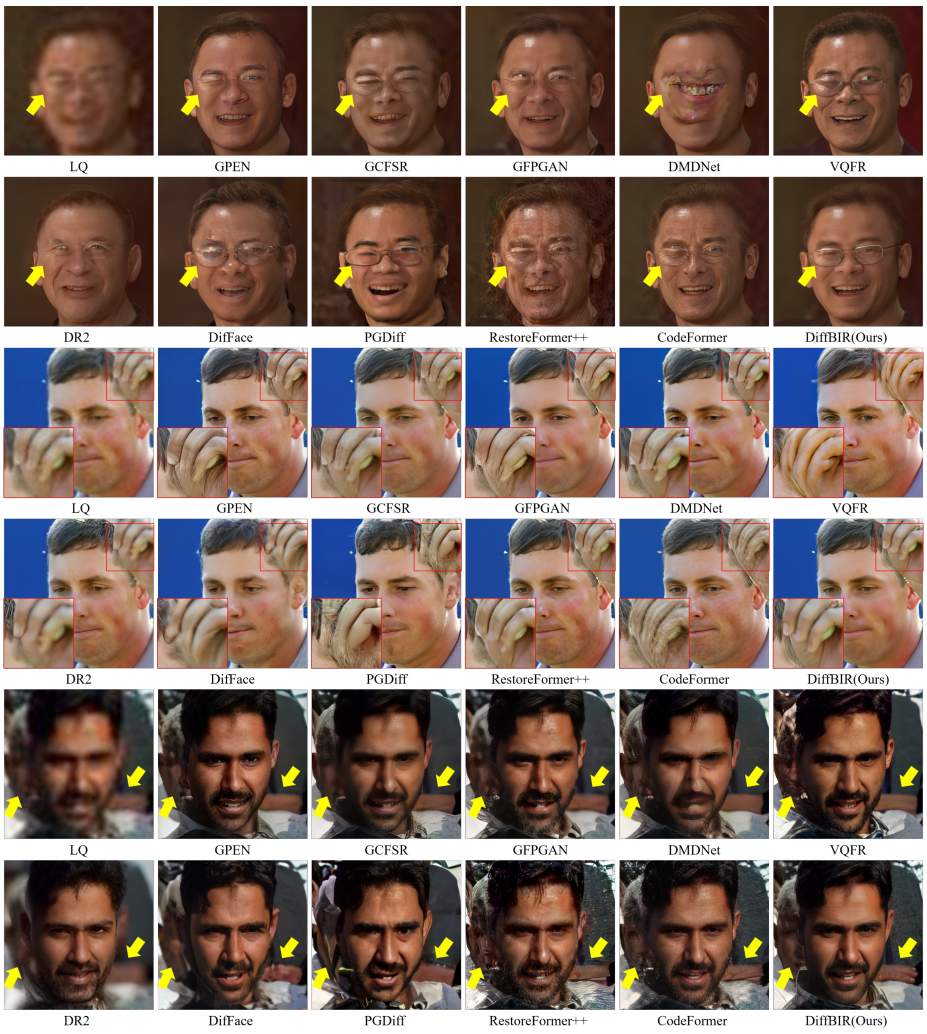

Fig. 6: More visual comparisons for BFR on real-world datasets.