Navigating Beyond Dropout: An Intriguing Solution Towards Generalizable Image Super Resolution

Supplementary Material

1. Experiment Settings

In order to simulate real-world degradations better, most state-of-the-art Blind SR researches examine their methods with the multi-degradations settings. However, since there is no unified standards for how the multi-degradations should be generated, different works usually employ it in their own ways. In this paper, for the purpose of a fair and credible validation of our method, we choose the widely adopted "second-order" degradation generation settings of Wang et al. [68] to verify our effectiveness for Blind SR. Note that the Dropout [37], which will be compared with our method in experiments, also adopt the same setting.

For our training, we leverage the high-resolution (HR) images from the DIV2K [1] dataset. During the training process, the L1 loss function is employed in combination with the Adam optimizer. The values of β_1 and β_2 of the Adam optimizer are set to 0.9 and 0.999 respectively. The batch size is set to 16, and the low-resolution (LR) images have dimensions of 32×32 pixels. To fine-tune the learning rate, we implement a cosine annealing learning strategy. Initially, the learning rate is set to 2×10^{-4}. The cosine annealing period for adjusting the learning rate spans 500,000 iterations. We have built all our models using the PyTorch framework and conducted the training on 4×NVIDIA A800 GPUs. For our testing phase, we utilize several benchmark datasets, including Set5 [4], Set14 [71], BSD100 [52], Manga109 [53], Test2k [36], and Urban100 [28]. In addition, we also test our method on a realistic NTIRE 2018 SR challenge data [61] to further show our general applicability. For evaluation, we primarily evaluate the model's performance using the Peak Signal-to-Noise Ratio (PSNR), a commonly used metric for image quality assessment [26].

In our method, all the alignment operations are conducted before the last convolutional layer (i.e., the output layer) of the model. This setting holds true throughout all the experiments and baseline models used in this paper. We do this because we think aligning features at the end of the model propagation can most effectively regularize its behaviors to generate similar outputs for input images with the same content but different degradations. In addition, the Dropout ratios used for different baseline models in this paper follow the best setting of Kong et al. [37] (i.e., SRResNet:0.7, RRDB:0.5, MSRN:0.5, SwinIR:0.5). More details of our implementation can be found in our codes.

Table 1. **Ablation Studies.**

Models	PSNR ↑
	Set5 / Set14 / BSD / Urban / Manga / Test2k
SRResNet	23.53 / 22.23 / 22.34 / 20.49 / 18.40 / 22.95
+brute-force	23.49 / 22.28 / 21.94 / 20.27 / 18.97 / 22.93
+w.o non-linear	24.01 / 22.54 / 22.76 / 20.78 / 19.05 / 23.30
+*Ours*	**24.20 / 22.83 / 22.82 / 20.96 / 19.12 / 23.41**
RRDB	23.62 / 22.45 / 22.48 / 20.66 / 18.50 / 23.02
+brute-force	23.98 / 22.69 / 22.70 / 19.81 / 18.78 / 23.18
+w.o non-linear	24.44 / 22.94 / 23.45 / 20.97 / 19.02 / 23.39
+*Ours*	**24.56 / 23.08 / 23.48 / 21.11 / 19.28 / 23.55**

2. Ablation Studies

In this section, we show the ablation studies that verify the significance of our design. To be specific, we (1) review the design of brutly forcing the intermediate features of two images with identical contents but different degradations to be exactly the same, as discussed in Sec. 4, and (2) justify the non-linear alignment design of our method. We run the experiments with SRResNet and RRDB on six benchmark datasets and use PSNR as the evaluation metric. The results are shown in Table 1. As we could observed, brutly forcing the features to be exactly the same, although theoretically the best, might put too much constraint on the model, limiting its ability to effectively reach a local minimum, thus yielding very unstable and unsatisfactory performances. On the other hand, experiments run with only linear alignment (i.e., w.o non-linear) show certain improvements, but its potential can be further excavated with the knowledge of higher dimension provided by the non-linear alignment.

3. Detailed Comparisons with Dropout

As we mentioned in Sec. 5, we provide the detailed quantitative comparison results of Fig. 6 in Table 2. Our method outperforms Dropout in almost all cases, which is not surprising and in line with our previous theoretical analyses.

4. Deep Degradation Representation

Following Kong et al. [37], we also adopt the deep degradation representation (DDR) introduced by [47] and visualized it in Fig. 1. In the figure, each point represents an input image and different colors indicate different degradations. DDR provides us a way to assess the network's generalization ability by peeking into the model behaviors.

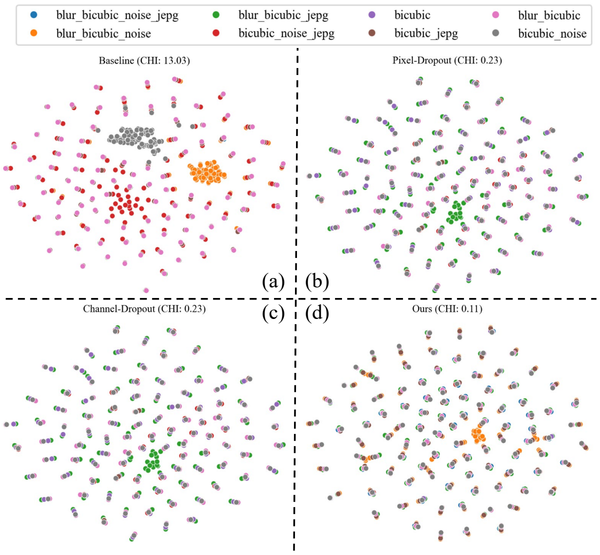


Figure 1. **The visualization of the DDR clusters of SRResNet trained with different regularizations. The CHI results are also provided to measure the separation degree of clusters.**

For example, in Fig. 1 (a) we can observe that images with the same degradations are clustered together, which means the model has learned to encode degradation-specific information, leading to its poor generalization ability. On the other hand, in Fig. 1 (d) images are clustered relying more on their contents instead of degradations, which means the model has become more degradation-invariant. Liu et al. [47] further introduce the Calinski-Harabaz Index (CHI) [9] for quantitative analysis, with a lower value indicating better cluster separation, and thus better generalization ability.

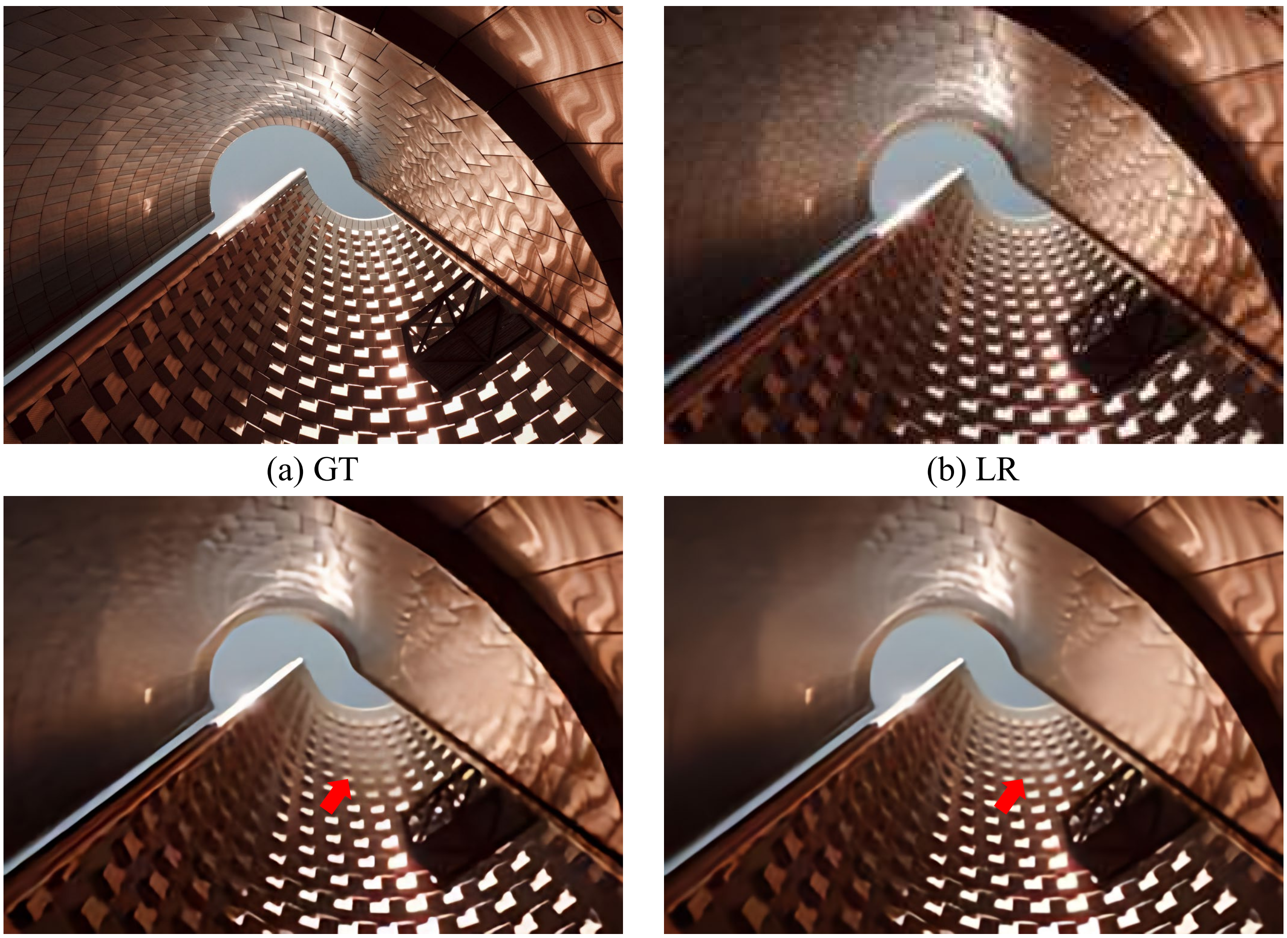
5. More Visual Results

We provide more visual comparison results in Fig. 2, Fig. 3, Fig. 4, and Fig. 5. They are examples of different degradation restoration results (see the captions of the figures), and the red arrows in the figures highlight the main improvements of our method from human visual perspective.

Table 2. **Six datasets with eight types of degradations (clean, noise, blur, jpeg, blur+noise, blur+jpeg, noise+jpeg, and blur+noise+jpeg) are used to evaluate the PSNR (dB) results of models with ×4 resolution.** The Dropout used in the experiments refers to the one in Kong et al. [37].

Models	Set5 [4]				Set14 [71]				BSD100 [52]			
	clean	blur	noise	jpeg	clean	blur	noise	jpeg	clean	blur	noise	jpeg
SRResNet [38]	24.85	24.73	22.52	23.67	23.25	23.05	21.18	22.32	23.06	22.99	21.34	22.47
+ Dropout ($p = 0.7$)	25.63	25.23	22.79	24.05	23.73	23.45	21.23	22.62	23.31	23.26	21.30	22.69
+ Ours	**25.93**	**25.62**	**23.15**	**24.38**	**24.12**	**23.80**	**21.67**	**22.99**	**23.83**	**23.64**	**21.77**	**23.04**
RRDB [66]	25.18	25.12	21.79	23.82	23.74	23.36	21.02	22.59	23.38	23.32	21.00	22.73
+ Dropout ($p = 0.5$)	26.02	26.07	22.23	24.15	24.02	23.87	21.54	22.83	23.59	23.66	21.68	22.86
+ Ours	**26.78**	**26.55**	**23.02**	**24.70**	**24.70**	**24.35**	**21.91**	**23.21**	**24.59**	**24.54**	**23.47**	**23.67**
MSRN [39]	25.25	24.89	22.57	24.08	23.38	23.10	21.80	22.53	23.38	23.30	21.92	22.76
+ Dropout ($p = 0.5$)	25.36	25.02	22.71	24.00	23.40	23.18	21.76	22.61	23.45	23.36	21.91	22.77
+ Ours	**25.81**	**25.52**	**22.84**	**24.46**	**23.93**	**23.64**	**21.86**	**22.83**	**23.72**	**23.58**	**22.01**	**22.98**
SwinIR [43]	26.25	26.03	22.96	24.37	24.53	24.25	22.08	23.14	23.91	23.83	22.12	23.04
+ Dropout ($p = 0.5$)	26.32	26.08	23.12	24.41	24.57	24.19	22.13	23.18	23.90	23.87	22.10	23.08
+ Ours	**26.49**	**26.23**	**24.61**	**24.68**	**24.65**	**24.28**	**22.23**	**23.29**	**24.04**	**23.96**	**22.21**	**23.15**
	b+n	b+j	n+j	b+n+j	b+n	b+j	n+j	b+n+j	b+n	b+j	n+j	b+n+j
SRResNet [38]	23.27	23.40	23.05	22.73	22.23	22.06	21.99	21.77	22.25	22.33	22.22	22.04
+ Dropout ($p = 0.7$)	23.47	23.64	23.46	23.01	22.28	22.39	22.28	21.98	22.25	22.50	22.41	22.16
+ Ours	**23.79**	**23.86**	**23.71**	**23.19**	**22.65**	**22.63**	**22.55**	**22.16**	**22.53**	**22.79**	**22.62**	**22.32**
RRDB [66]	23.44	23.45	23.32	22.81	22.47	22.17	22.29	21.95	22.39	22.47	22.42	22.15
+ Dropout ($p = 0.5$)	23.73	23.88	23.68	23.18	22.58	22.59	22.45	22.10	22.53	22.71	22.52	22.28
+ Ours	**24.12**	**24.14**	**23.93**	**23.26**	**22.80**	**22.76**	**22.71**	**22.21**	**22.85**	**23.21**	**22.97**	**22.54**
MSRN [39]	23.55	23.59	23.50	22.95	22.39	22.23	22.19	21.97	22.57	22.61	22.45	22.24
+ Dropout ($p = 0.5$)	23.73	23.61	23.52	23.04	22.43	22.26	22.24	21.96	22.59	22.64	22.44	22.20
+ Ours	**23.70**	**23.80**	**23.73**	**23.06**	**22.52**	**22.49**	**22.48**	**22.08**	**22.68**	**22.73**	**22.56**	**22.26**
SwinIR [43]	23.80	23.84	23.67	22.99	22.53	22.73	22.59	22.20	22.61	22.82	22.61	22.34
+ Dropout ($p = 0.5$)	24.00	23.93	23.65	23.09	22.73	22.71	22.65	22.22	22.68	22.80	22.64	22.33
+ Ours	**24.13**	**24.17**	**23.89**	**23.09**	**22.87**	**22.79**	**22.81**	**22.28**	**22.77**	**22.98**	**22.76**	**22.40**

Models	Test2k [36]				Urban100 [28]				Manga109 [53]			
	clean	blur	noise	jpeg	clean	blur	noise	jpeg	clean	blur	noise	jpeg
SRResNet [38]	23.91	23.71	21.77	23.11	21.23	21.06	19.74	20.60	18.42	18.75	18.08	18.27
+ Dropout ($p = 0.7$)	24.26	23.98	21.75	23.27	21.57	21.25	19.75	20.90	18.98	19.12	18.42	18.66
+ Ours	**24.58**	**24.43**	**22.17**	**23.65**	**21.94**	**21.65**	**20.19**	**21.20**	**19.18**	**19.46**	**18.90**	**19.02**
RRDB [66]	24.16	23.64	21.34	23.36	21.57	21.18	19.61	20.93	18.59	18.64	18.30	18.41
+ Dropout ($p = 0.5$)	24.55	24.39	21.92	23.53	21.89	21.75	19.92	21.12	18.73	19.03	18.72	18.60
+ Ours	**24.97**	**24.76**	**22.15**	**23.86**	**22.29**	**21.95**	**20.21**	**21.40**	**19.40**	**19.61**	**18.96**	**19.24**
MSRN [39]	22.99	23.83	22.30	23.22	21.35	21.14	20.19	20.75	19.12	19.31	18.72	18.89
+ Dropout ($p = 0.5$)	23.94	23.97	22.31	23.33	21.46	21.25	20.18	20.81	19.16	19.31	18.78	18.94
+ Ours	**24.52**	**24.23**	**22.38**	**23.56**	**21.88**	**21.54**	**20.22**	**21.14**	**19.23**	**19.35**	**18.84**	**19.01**
SwinIR [43]	24.78	24.57	22.71	23.63	22.18	21.90	20.56	21.32	19.10	19.27	18.71	18.95
+ Dropout ($p = 0.5$)	24.81	24.54	22.76	23.69	22.27	21.99	20.67	21.38	19.15	19.30	18.73	19.03
+ Ours	**24.98**	**24.76**	**22.84**	**23.80**	**22.34**	**22.07**	**20.69**	**21.48**	**19.24**	**19.45**	**18.98**	**19.28**
	b+n	b+j	n+j	b+n+j	b+n	b+j	n+j	b+n+j	b+n	b+j	n+j	b+n+j
SRResNet [38]	22.81	22.87	22.85	22.59	20.46	20.30	20.42	20.09	18.59	18.50	18.21	18.39
+ Dropout ($p = 0.7$)	22.89	23.11	23.04	22.70	20.48	20.49	20.66	20.22	18.94	18.85	18.66	18.72
+ Ours	**23.11**	**23.27**	**23.22**	**22.82**	**20.73**	**20.72**	**20.91**	**20.37**	**19.27**	**19.17**	**18.98**	**19.01**
RRDB [66]	22.93	22.87	23.12	22.73	20.57	20.40	20.74	20.24	18.83	18.43	18.38	18.41
+ Dropout ($p = 0.5$)	23.02	23.26	23.17	22.79	20.53	20.70	20.84	20.33	19.15	18.81	18.59	18.71
+ Ours	**23.14**	**23.37**	**23.34**	**22.82**	**20.76**	**20.85**	**21.03**	**20.38**	**19.43**	**19.31**	**19.12**	**19.15**
MSRN [39]	23.03	23.01	22.94	22.66	20.65	20.43	20.56	20.19	19.16	19.02	18.80	18.88
+ Dropout ($p = 0.5$)	23.05	23.09	22.96	22.68	20.69	20.45	20.62	20.22	19.21	19.18	18.87	18.89
+ Ours	**23.21**	**23.21**	**23.18**	**22.78**	**20.76**	**20.64**	**20.89**	**20.26**	**19.19**	**19.18**	**18.92**	**18.93**
SwinIR [43]	23.15	23.27	23.21	22.81	20.89	20.79	20.98	20.45	19.07	19.02	18.79	18.80
+ Dropout ($p = 0.5$)	23.23	23.31	23.26	22.82	20.92	20.91	20.96	20.55	19.12	18.98	18.75	18.84
+ Ours	**23.35**	**23.47**	**23.45**	**22.93**	**21.02**	**20.98**	**21.12**	**20.53**	**19.37**	**19.35**	**19.15**	**19.12**



(a) GT (b) LR

(c) SRResNet
22.87 dB

(d) SRResNet + Ours
23.21 dB

Figure 2. **Visual comparison with and without our approach in "bicubic+noise20+jepg50". (Zoom in for best view)**

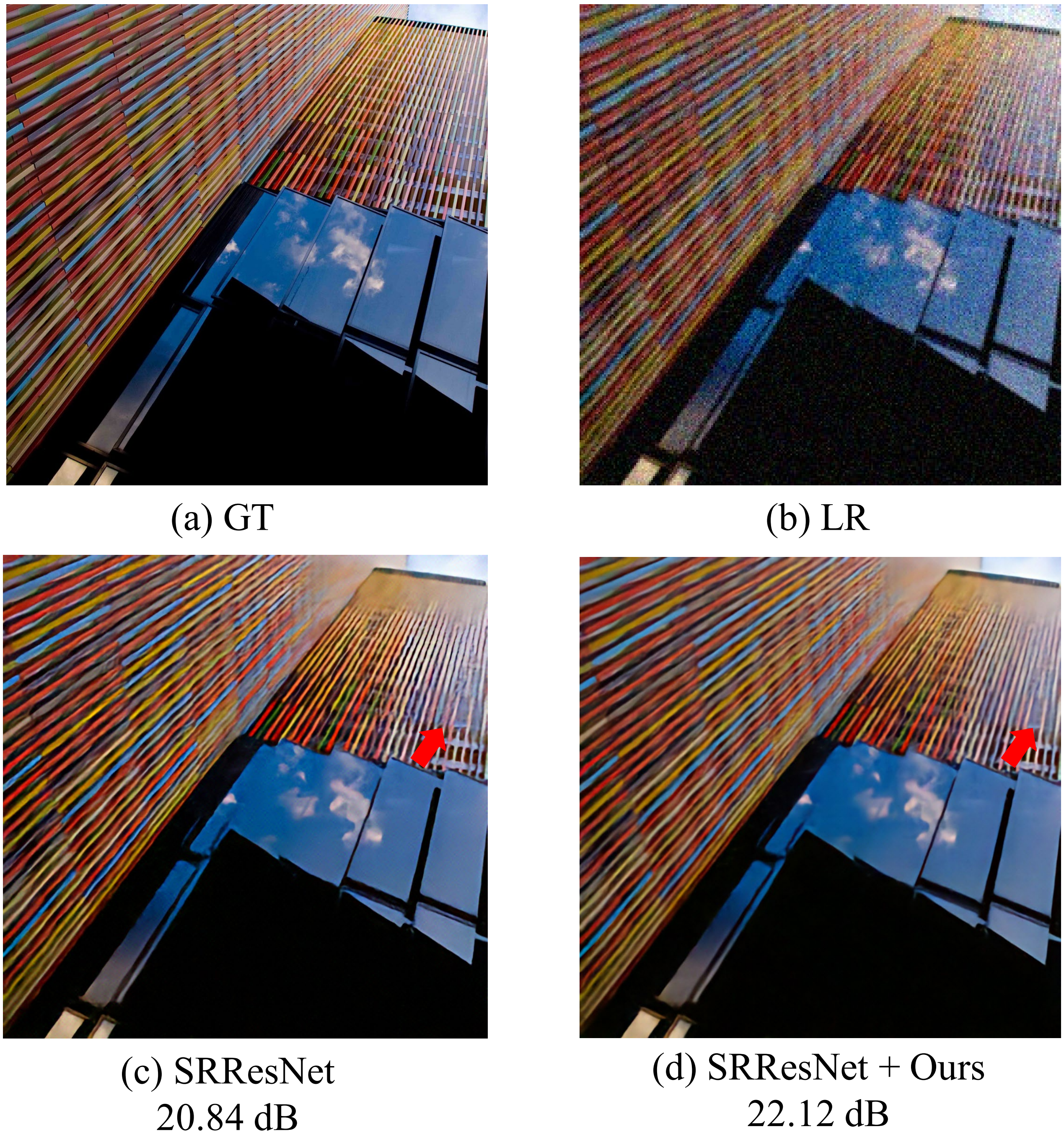


Figure 3. **Visual comparison with and without our approach in "blur2+bicubic+jepg50". (Zoom in for best view)**

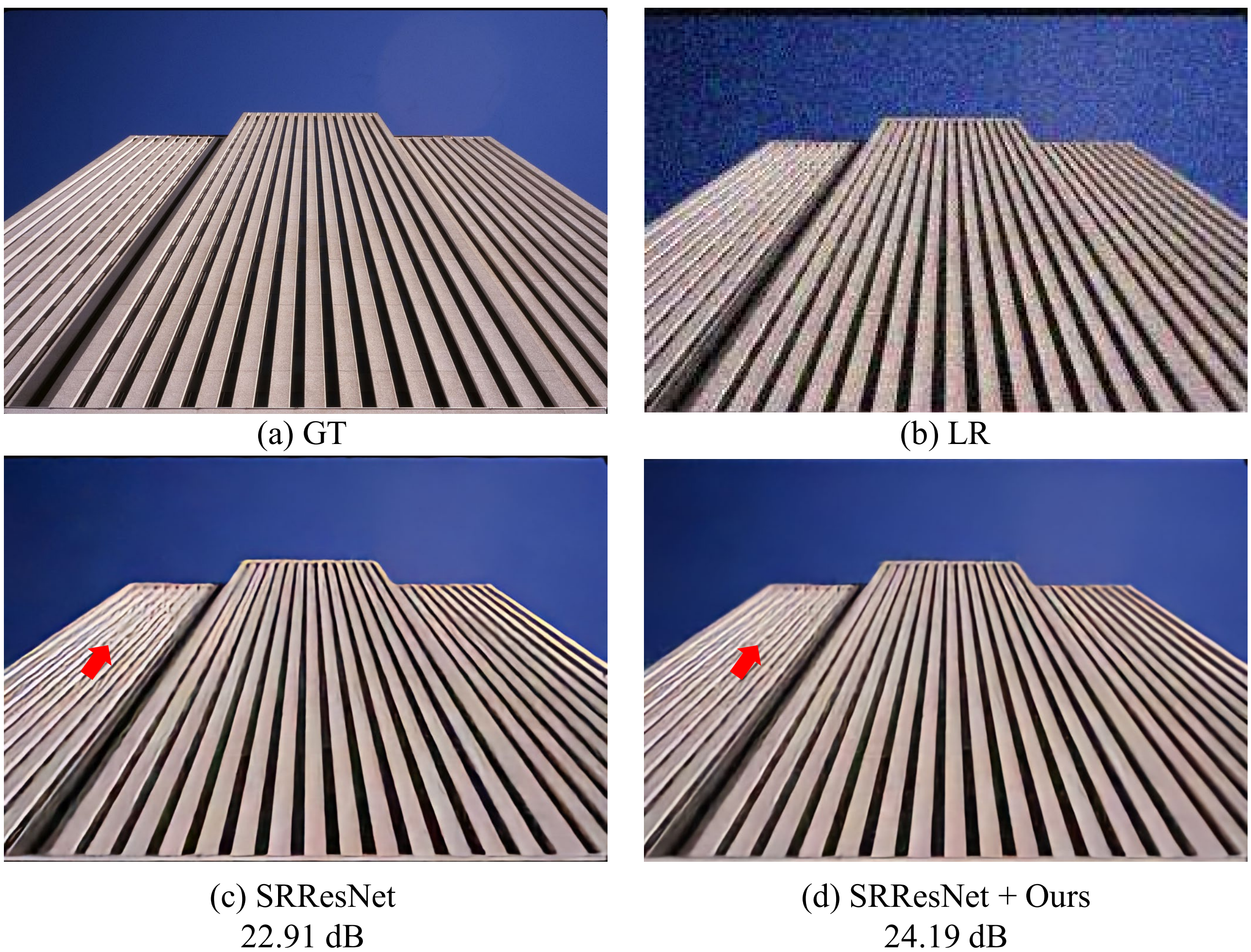
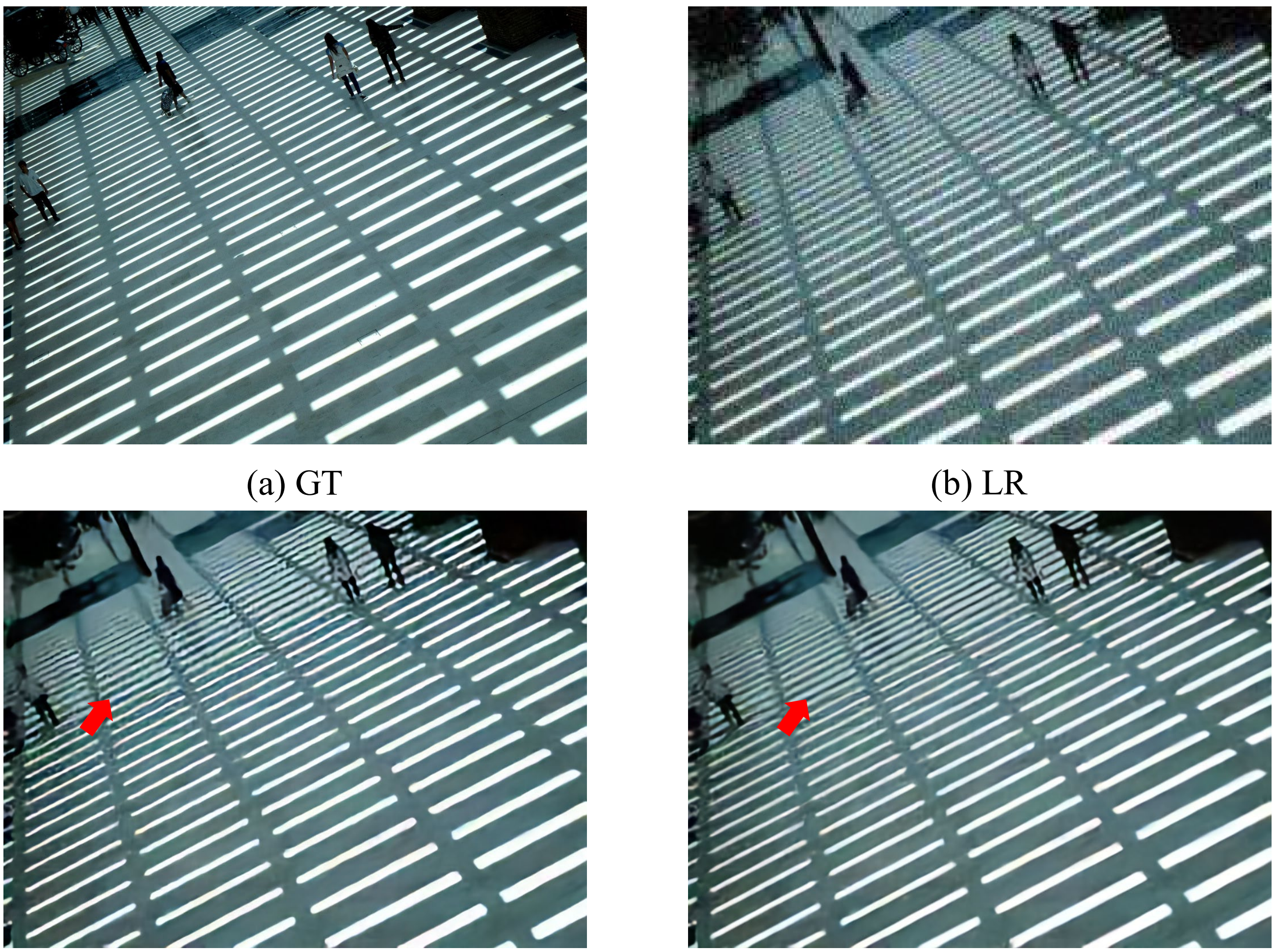


Figure 4. **Visual comparison with and without our approach in "bicubic+noise20+jepg50". (Zoom in for best view)**



(a) GT

(b) LR

(c) SRResNet
20.38 dB

(d) SRResNet + Ours
20.78 dB

Figure 5. **Visual comparison with and without our approach in “blur2+bicubic+noise20+jepg50”. (Zoom in for best view)**