Improving Image Restoration through Removing Degradations in Textual Representations

Supplementary Material

The content of the supplementary material involves:

- Experimental details in Sec. A.
- Effect of textual restoration in Sec. B.
- Explicit textual representation v.s. implicit textual representation in Sec. C.
- Guidance visualization in Sec. D.
- More visual comparisons in Sec. E.

A. Experimental Details

In this section, we list experimental details for different image restoration task (*i.e.*, all-in-one image restoration [?], image deblurring [? ?], image dehazing [?], image deraining [?], and image denoising [?]), the proposed degradation-free guidance generation process (*i.e.*, image-to-text mapping and textual restoration), and guided-restoration.

All-in-One Image Restoration. We adopt PromptIR [?] as our backbone in all-in-one image restoration. Following [?], network has 8 stages (the first 7 stages as main network, the last stage as refinement), the number of blocks for each stages is [4, 6, 6, 8, 6, 6, 4, 4], network width is 48, the number of heads for each stages is [1, 2, 4, 8, 4, 2, 1, 1]. In perspective of training data, we adopt concatenation of 400 images from BSD [?] and 4,744 images from WED [?] dataset as denoising training data, 200 images from Rain100L [?] for deraining task, 72,135 images from SOTS for dehazing task. Considering dataset size gap among different tasks, we properly enlarge deraining data and denoising data as [?]. To train, we adopt AdamW optimizer with CosineAnnealing learning rate scheduler, the initial learning rate of the main restoration network and dynamic aggregation is set to 2e-4 and 1e-4, respectively. We train all-in-one image restoration on 4 Tesla-V100 GPUs with training patch size 128, batch size 48. Performance reported on Table 1 is referred to [?]. PSNR and SSIM scores are calculated on RGB channels, except which of deraining task are calculated on Y-channel in YCbCr color space.

Image Deblurring. We adopt NAFNet [?] as our backbone in single-image motion deblurring, Restormer [?] as backbone in defocus deblurring. In single-image motion deblurring task, we follow [?], main restoration network has 9 stages, and the number of blocks for each stage is [1, 1, 1, 28, 1, 1, 1, 1, 1], network width is 64. We adopt GoPro [?] as our training data and directly evaluated the trained model on GoPro validation set, HIDE [?] testing set, and Realblur [?] dataset. GoPro dataset has 4,214 blur-sharp paris of data (2,103 for training and 1,111 for validation), testset of HIDE dataset consist of 2,025 images, and two subsets of Realblur both have 980 images. To train single-image motion deblurring, we adopt AdamW optimizer with CosineAnnealing learning rate scheduler, the initial learning rate of the main restoration network and dynamic aggregation module is 1e-4 and 5e-5, respectively. We train single-image motion deblurring on 8 Tesla-V100 GPUs with training patch size of 256, batch size of 16. In defocus deblurring task, we follow [?], main restoration network has 8 stages (the last stage as refinement stage), the number of blocks of each stage is [4, 6, 6, 8, 6, 6, 4, 4], network width is 48, the number of heads of each stage is [1, 2, 4, 8, 4, 2, 1, 1]. Note that input channels is 6 in dual-pixel defocus deblurring, the output channels is 3 in both single-image defocus deblurring and dual-pixel defocus deblurring. We adopt DPDD [?] dataset as our training data. DPDD dataset contains 500 indoor & outdoor scenes captured by DSLR camera. Each scene includes three defocus input images and a corresponding all-in-focus ground-truth image. Three input images are labeled as left, right and center views. The left and right defocused sub-aperture views are acquired with a wide camera aperture setting, and the corresponding all-in-focus ground-truth image captured with a narrow aperture. Following Restormer [?], we use sub-aperture data to train dual-pixel defocus deblurring, and we use center input image to train single-image defocus deblurring. To perform evaluation, we separately evaluate trained model in indoor & outdoor scene testing data and the average performance is calculated by weighted combination. To train defocus deblurring, we maintain progressive learning in official implementation, we adopt AdamW optimizer with CosineAnnealing learning rate scheduler, the initial learning rate of main restoration network and dynamic aggregation module is set to 3e-4 and 1e-4, respectively. Performance reported in Table 2 and Table 3 is referred to [?] and [?]. PSNR and SSIM scores are calculated on RGB channels.

Image Dehazing. We adopt SFNet [?] as our backbone in image dehazing. Following [?], network width is set to 32, the number of resblocks is 16. We train image dehazing on RESIDE [?] dataset, we train and evaluate method separately on indoor scene and outdoor scene data. To train image dehazing, we use Adam optimizer with CosineAnnealing learning rate scheduler, the initial learning rate of main network and dynamic aggregation is set to 1e-4 and 5e-5, respectively. Performance in Table 4 is referred to [?]. PSNR and SSIM scores are calculated on RGB channels.

Image Deraining. We adopt DRSformer [?] as our backbone in image deraining. Following [?], we adopt 7 stages for main restoration network, the number of blocks for each stage is [4, 6, 6 ,8, 6, 6, 4], network width is set to 48, the number of heads for each stage is [1, 2, 4, 8, 4, 2, 1]. For training, we separately train and evaluate the proposed method on four datasets with synthetic rainstreak degradation, including Rain200L [?], Rain200H [?], DID-Data [?], and DDN-Data [?]. Rain200H and Rain200L dataset contain 1,800 pairs of rainy-clean images for training and 200 pairs images for testing. In DID-Data and DDN-Data, the synthetic rainstreak has different directions and different levels. DID-Data contains 12,000 pairs of images for training and 1,200 pairs of images for testing. DDN-Data contains 12,600 pairs of images for training and 1,400 images for testing. We employ MEFC [?] module for Rain200H, DID-Data, and DDN-Data. During training, we adopt AdamW optimizer with CosineAnnealing learning rate scheduler, patch size and batch size is set to 128 and 16 with 4 Tesla-V100 GPUs, the initial learning rate of main network and dynamic aggregation is set to 1e-4 and 5e-5, respectively. Performance in Table 5 is referred to [?]. PSNR and SSIM scores are calculated on Y channel in YCbCr color space.

Image Denoising. We adopt Restormer [?] as our backbone in image denoising. Following [?], we employ the bias-free network with 8 stages, the number of blocks for each stage is [4, 6, 6, 8, 6, 6, 4, 4], network width is set to 48, the number of heads of each stage is [1, 2, 4, 8, 4, 2, 1, 1]. We adopt concatenation data of Div2k [?] (800 images for training), Flickr2k (2,650 images for training), BSD [?] (400 images for training), and WED [?] (4,744 images for training) to train Gaussian grayscale denoising and Gaussian color denoising. We adopt 320 high-resolution images in SIDD [?] dataset for real-world denoising. During training, we also maintain the progressive learning strategy, we adopt AdamW optimizer with CosinAnnealing learning rate scheduler, the initial learning rate of main restoration network and dynamic aggregation is set to 3e-4 and 1e-4, respectively. Performance in Table 6, Table 7, and Table 8 is referred to [?]. PSNR and SSIM scores are calculated on RGB channels.

Image-to-Text Mapping. To enable our image-to-text mapping network can project both clean images and degraded images into textual space, we use the collection of high-quality data, degraded data from different image restoration tasks as our training data. High-quality data includes LSDIR [?] dataset and HQ-50K [?] dataset, LSDIR dataset contains 84,991 high-quality images for training, HQ-50K dataset contains 50,000 high-quality images for training. Degraded data includes GoPro, RESIDE, Rain200H, Rain200L, DID-Data, DDN-Data, and DFBW data with synthetic Gaussian noise. During training, we crop high-quality high-resolution data (LSDIR and HQ-50K) into 512×512 as input, for others we centerly crop images along shorter side and resize them to 512×512 as input. The mapping network is implemented as four-layer MLP network, and we adopt N=20 words to control representation capability of textual word embedding. To encode image concepts into textual space, feature from the last layer of CLIP image encoder is selected as input to image-to-text mapping network. The learning rate is set to 1e-6 and batch size is set to 4.

Textual Restoration. To train textual restoration network, we use concatenation of training dataset used in different image restoration tasks as our training data. We adopt the same strategy to preprocess pairs of degraded-clean data to 512×512 patches as input. The same with image-to-text mapping network, the textual restoration network is also implemented by four-layer MLP network. The learning rate is set to 1e-6 and training batch size is set to 4. During guidance generation, we use 200 steps of DDIM scheduler with scale of 5.

Guided-Restoration. Following [?], the dynamic aggregation includes two steps: feature matching and feature aggregation. In feature matching, we adopt a shared n-stages encoder to extract multi-scale feature from degraded input and clean guidance, n depends on total downsampling ratio of the main restoration network, each stage is with 4 residual blocks, the width of encoder is the same to the width of main network. We then adopt a coarse-to-fine manner to match useful information for each patch of degraded input, *e.g.*, we first match in coarse block level then match in fine patch level. In coarse matching, feature block size is set to 8, dilation ratio is set to [1, 2, 3]. In fine matching, patch size is set to 3. For feature aggregation, we employ a more general way. We simply use concatenation & residual/self-attention blocks with adaptive scaling factor α to fuse guidance information to main restoration network, *i.e.*, Eq. (4).

B. Effect of Textual Restoration

In this section, we demonstrate the effectiveness of our textual restoration. We discard textual restoration and directly use the output of image-to-text mapping network as conditional input for diffusion model, and the visual results of the synthetic guidance images are shown in Fig. A, denoted as **w/o. textual restoration**.

C. Explicit Textual Representation v.s. Implicit Textual Representation

In this section, we compare synthetic guidance images conditioned on explicit text representation and implicit textual

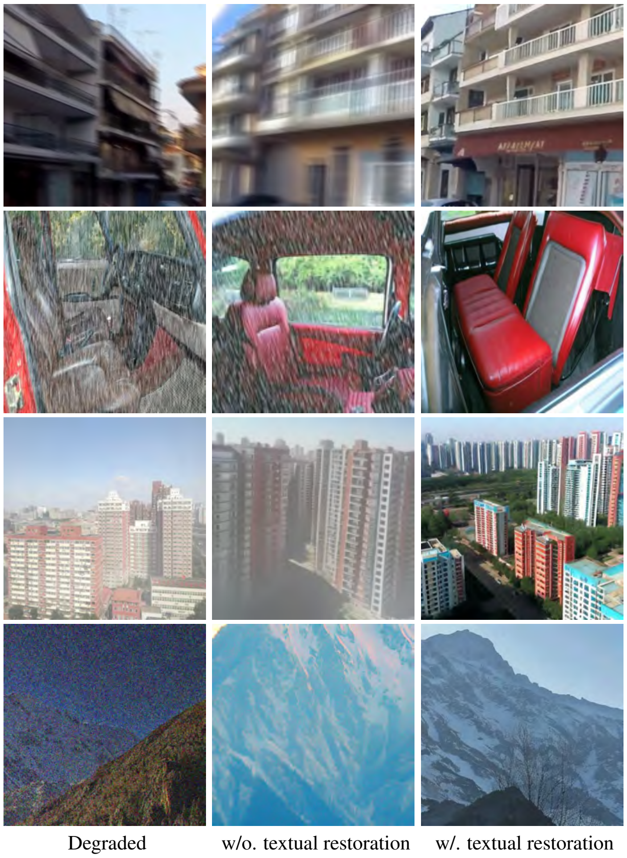


Figure A. Visual comparison of w/o. textual restoration and w/. textual restoration.

representation, 1) explicit text representation: we first convert degraded images into image caption by BLIPv2 [**?**], then we manually discarding degradation-related text in image caption, finally we use the processed image caption as text prompt input to StableDiffusion to get synthetic guidance images. Denoted as **Explicit**. 2) implicit textual representation: our method, which is denoted as **Ours**. As shown in Fig. B, Fig. C, Fig. D, and Fig. E, we illustrate visual comparison for image deblurring, image deraining, image dehazing, and image denoising tasks. We can found though explicit text representation can describe content of degraded image properly, the synthetic results cannot maintain style, details and texture of original content. And in image denoising task, explicitly converting degraded noise image into image caption usually leads to wrong captions and thus cannot provide useful guidance image for restoration.

D. Guidance Visualization

We illustrate synthesized guidance images for each image restoration:image deblurring shows in Fig. F, image deraining shows in Fig. G, image dehazing shows in Fig. H, image denoising shows in Fig. I.

E. More Visual Comparisons

We provide visual comparison for different image restoration tasks:

- All-in-one image restoration: image deraining results show in Fig. J and Fig. K, image dehazing results show in Fig. L, image denoising results show in Fig. M.
- Image deblurring results: single-image motion deblurring results show in Fig. N and Fig. O, defocus deblurring results show in Fig. P and Fig. Q.
- Image dehazing results: Fig. R.
- Image deraining results: Fig. S, Fig. T, and Fig. U.
- Image denoising results: Fig. V.

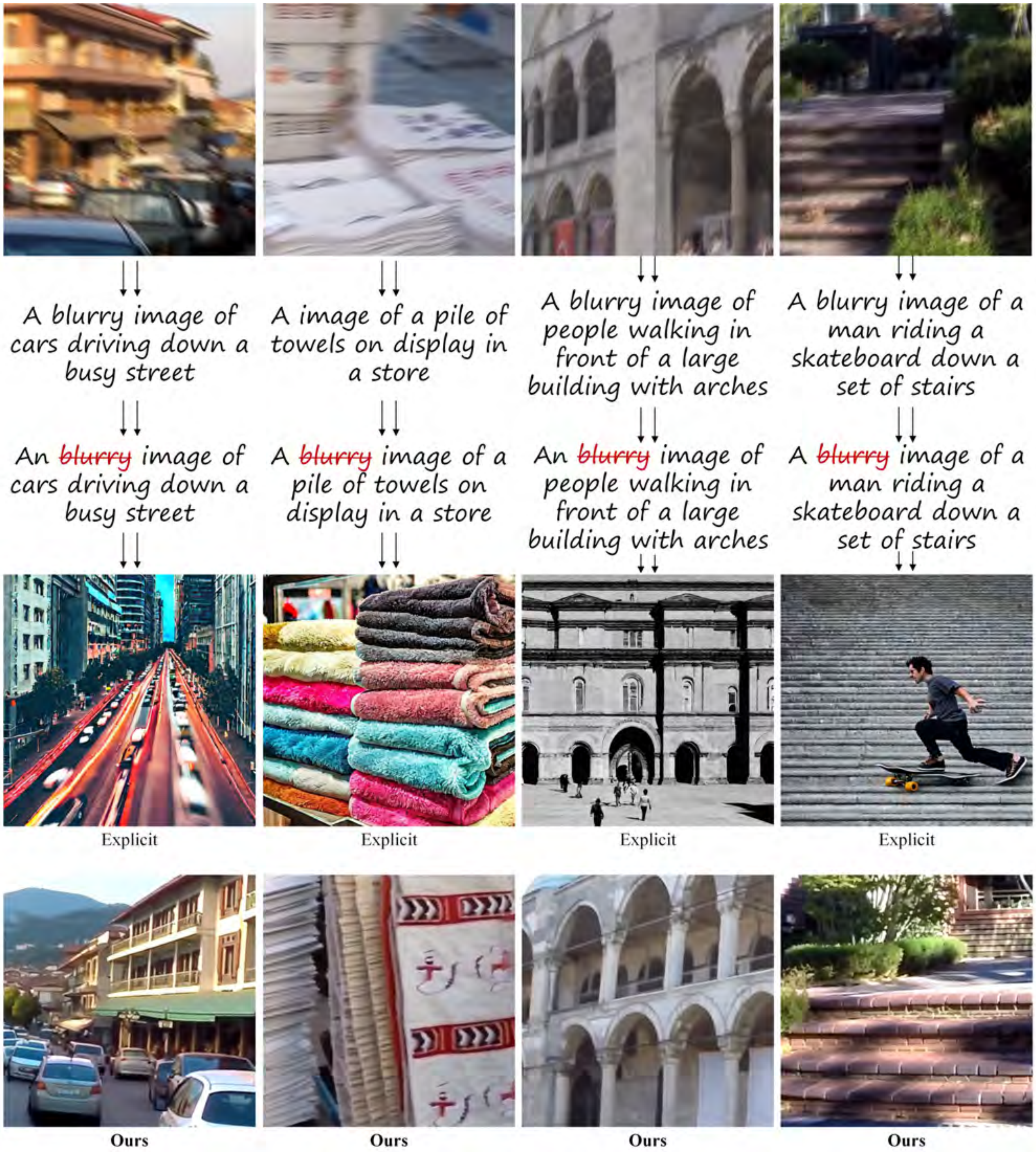

A blurry image of cars driving down a busy street
A image of a pile of towels on display in a store
A blurry image of people walking in front of a large building with arches
A blurry image of a man riding a skateboard down a set of stairs
An ~~blurry~~ image of cars driving down a busy street
A ~~blurry~~ image of a pile of towels on display in a store
An ~~blurry~~ image of people walking in front of a large building with arches
A ~~blurry~~ image of a man riding a skateboard down a set of stairs
Explicit
Explicit
Explicit
Explicit
Ours
Ours
Ours
Ours

Figure B. Visual comparison of synthetic guidance by explicit and implicit textual representation on image deblurring task.

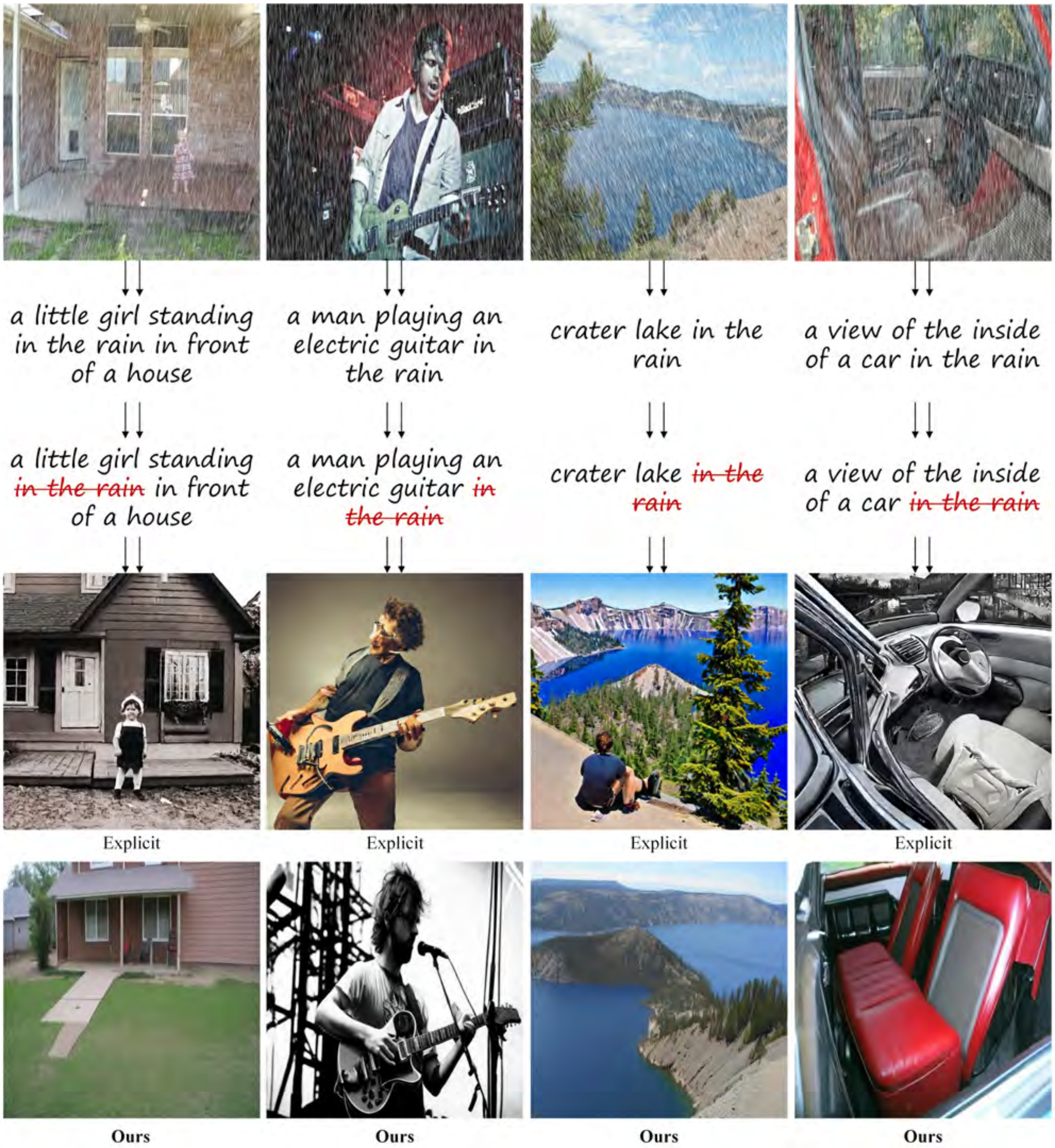

a little girl standing in the rain in front of a house
a man playing an electric guitar in the rain
crater lake in the rain
a view of the inside of a car in the rain
a little girl standing ~~in the rain~~ in front of a house
a man playing an electric guitar ~~in the rain~~
crater lake ~~in the rain~~
a view of the inside of a car ~~in the rain~~
Explicit
Explicit
Explicit
Explicit
Ours
Ours
Ours
Ours

Figure C. Visual comparison of synthetic guidance by explicit and implicit textual representation on image deraining task.

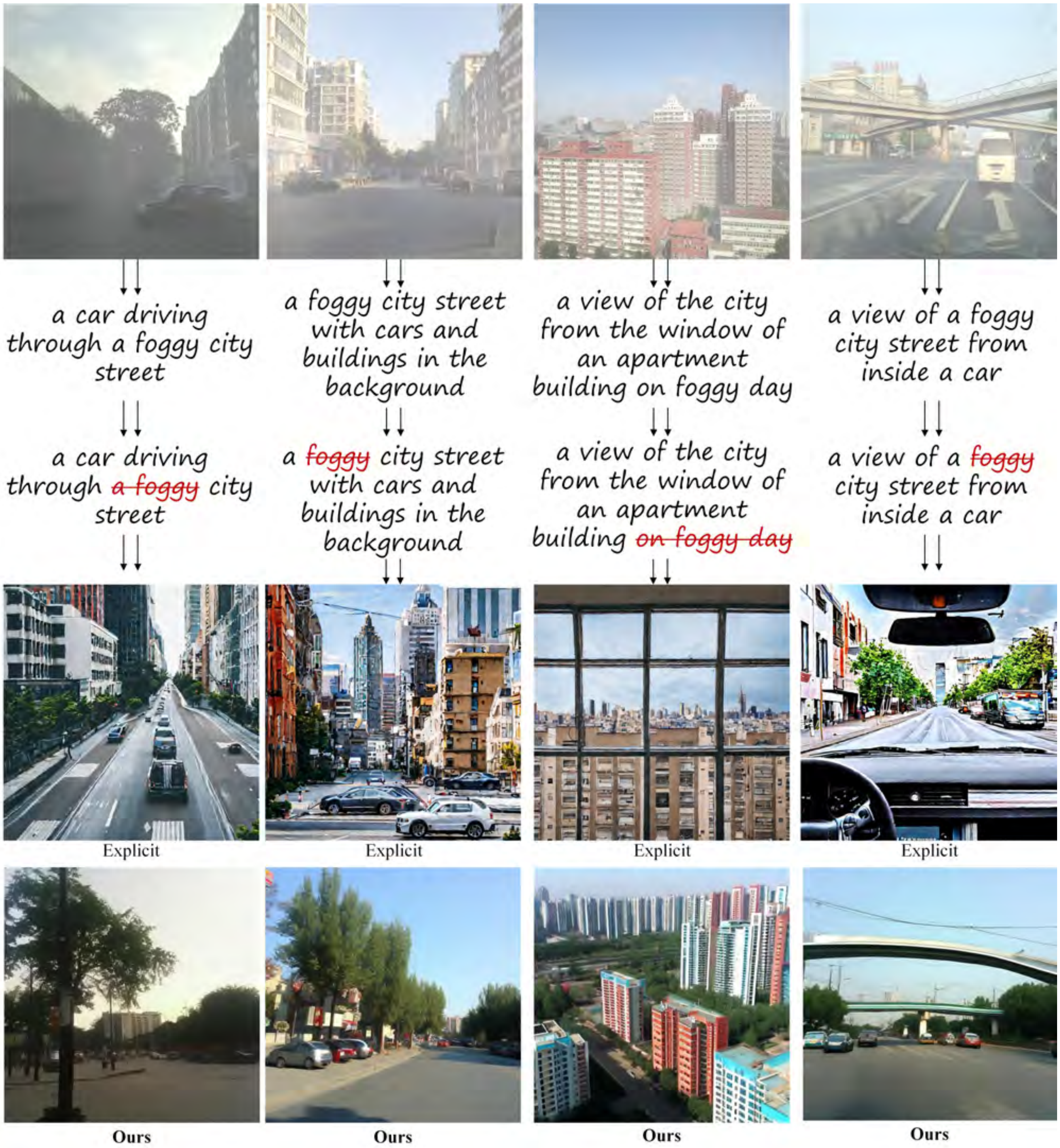

a car driving through a foggy city street
a foggy city street with cars and buildings in the background
a view of the city from the window of an apartment building on foggy day
a view of a foggy city street from inside a car
a car driving through ~~a foggy~~ city street
a ~~foggy~~ city street with cars and buildings in the background
a view of the city from the window of an apartment building ~~on foggy day~~
a view of a ~~foggy~~ city street from inside a car
Explicit
Explicit
Explicit
Explicit
Ours
Ours
Ours
Ours

Figure D. Visual comparison of synthetic guidance by explicit and implicit textual representation on image dehazing task.

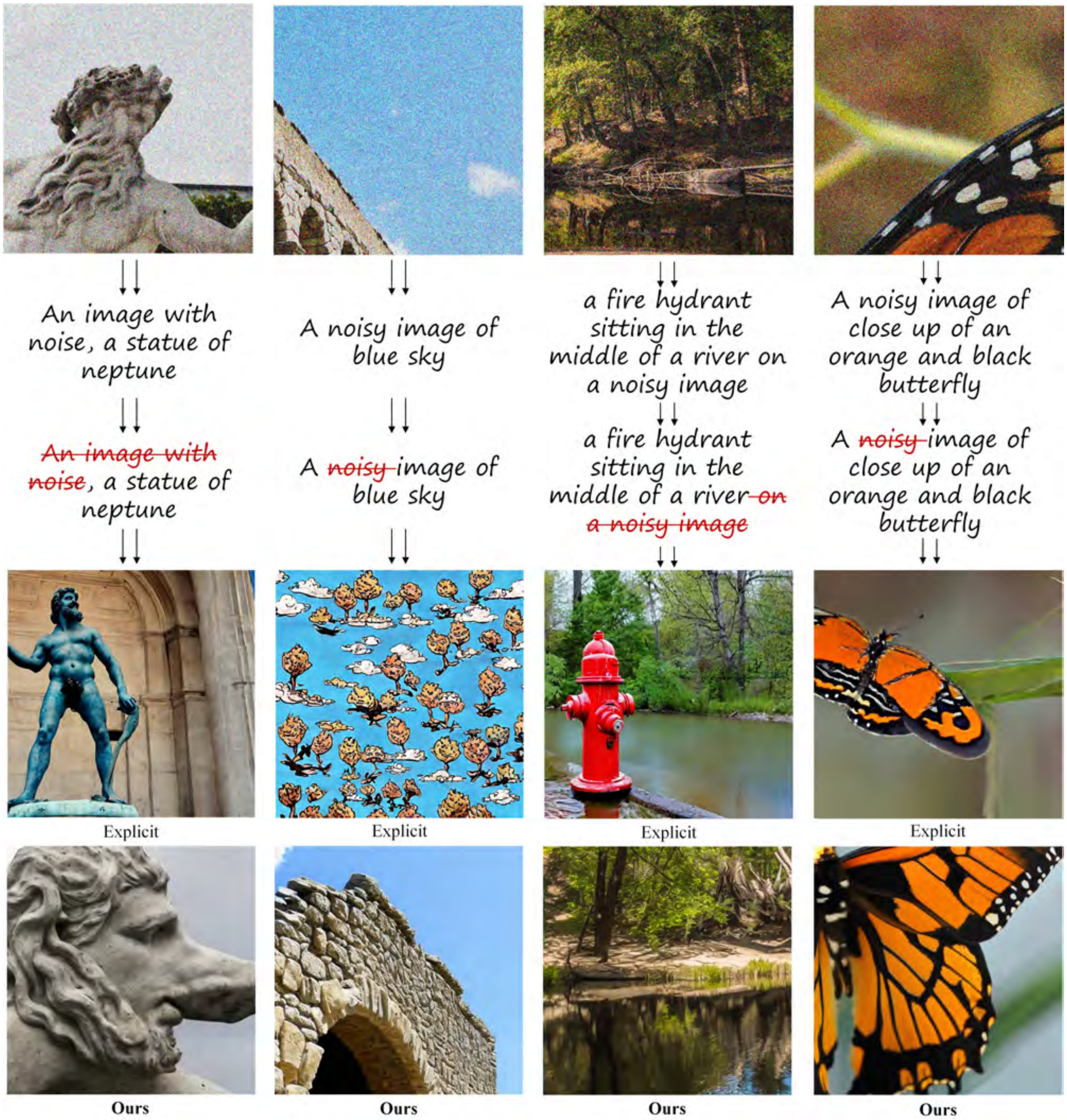
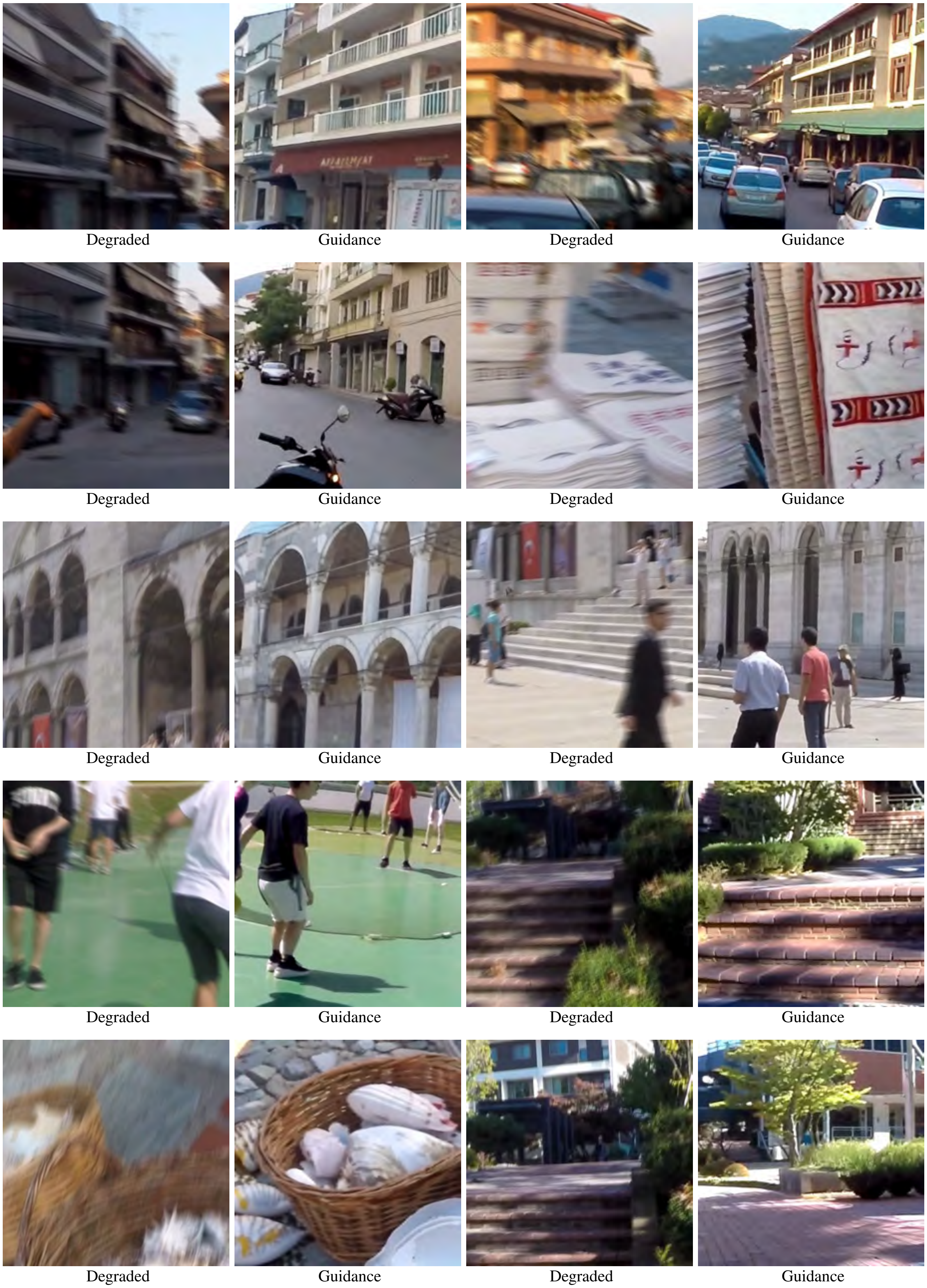

An image with noise, a statue of neptune
~~An image with noise~~, a statue of neptune
Explicit
Ours
A noisy image of blue sky
A ~~noisy~~ image of blue sky
Explicit
Ours
a fire hydrant sitting in the middle of a river on a noisy image
a fire hydrant sitting in the middle of a river ~~on a noisy image~~
Explicit
Ours
A noisy image of close up of an orange and black butterfly
A ~~noisy~~ image of close up of an orange and black butterfly
Explicit
Ours

Figure E. Visual comparison of synthetic guidance by explicit and implicit textual representation on image denoising task.



Degraded Guidance Degraded Guidance

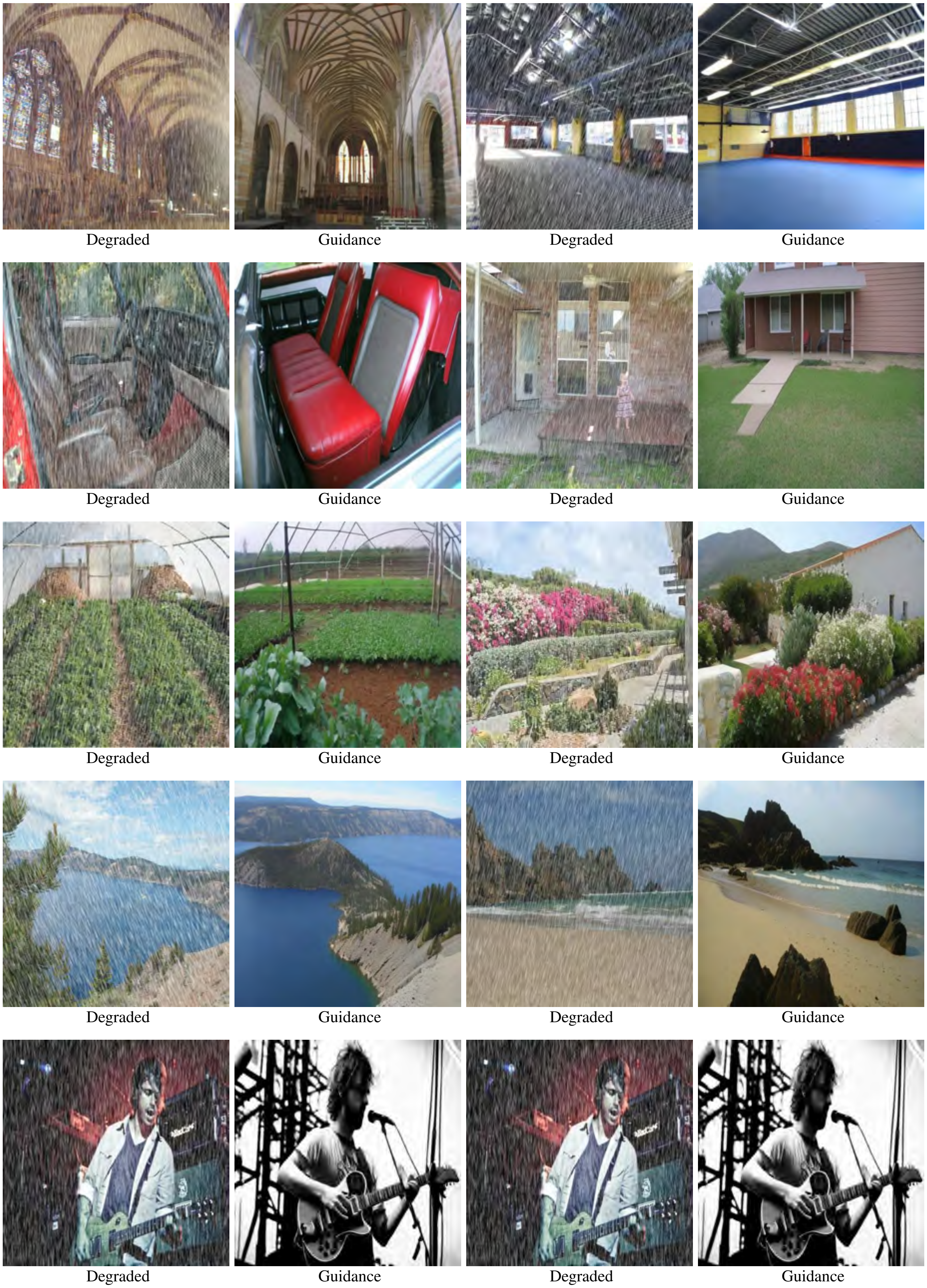
Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Figure F. Illustration of guidance images for image deblurring task.



Degraded Guidance Degraded Guidance

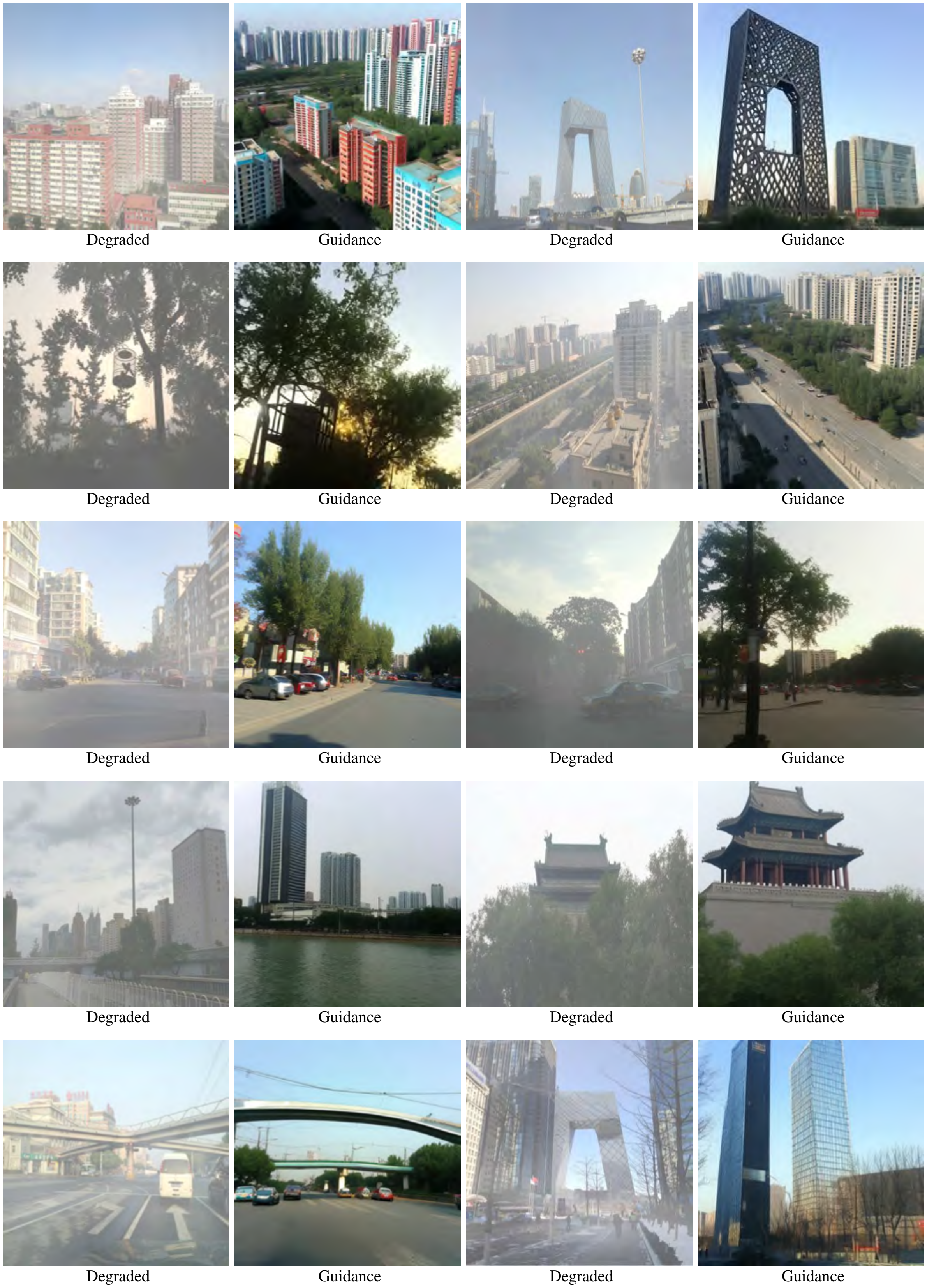
Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Figure G. Illustration of guidance images for image deraining task.



Degraded Guidance Degraded Guidance

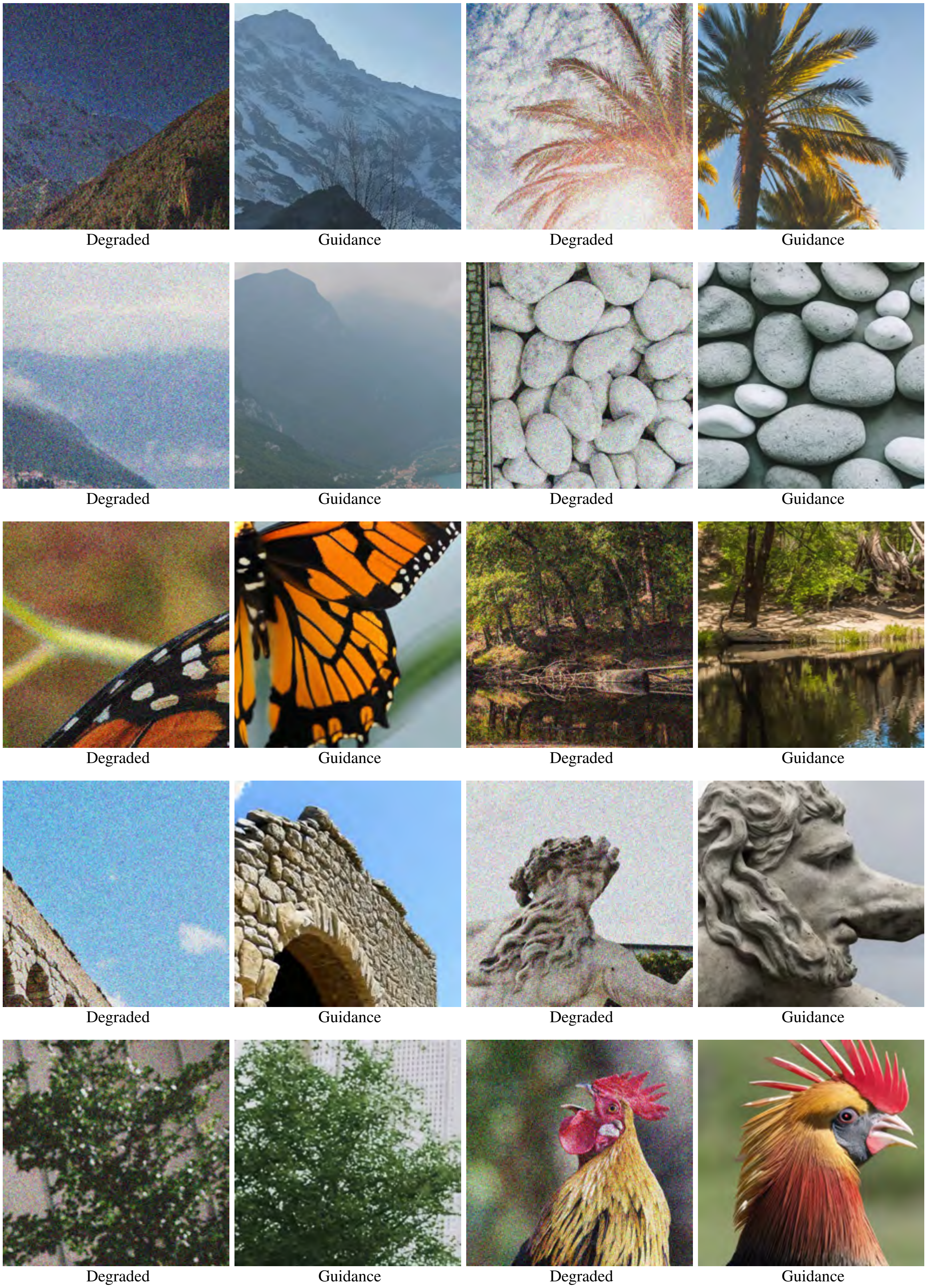
Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Figure H. Illustration of guidance images for image dehazing task.



Degraded Guidance Degraded Guidance

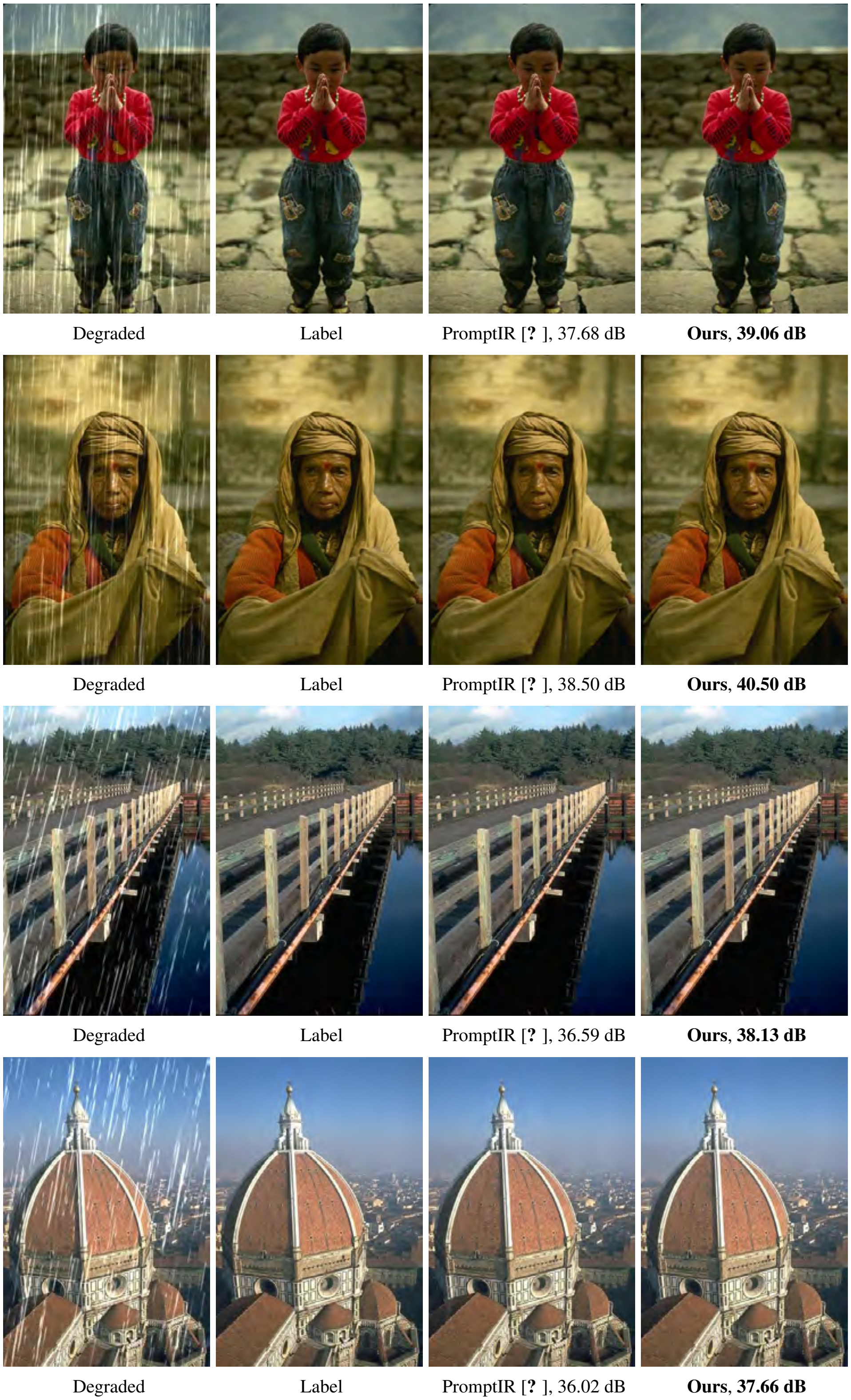
Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Degraded Guidance Degraded Guidance

Figure I. Illustration of guidance images for image denoising task.



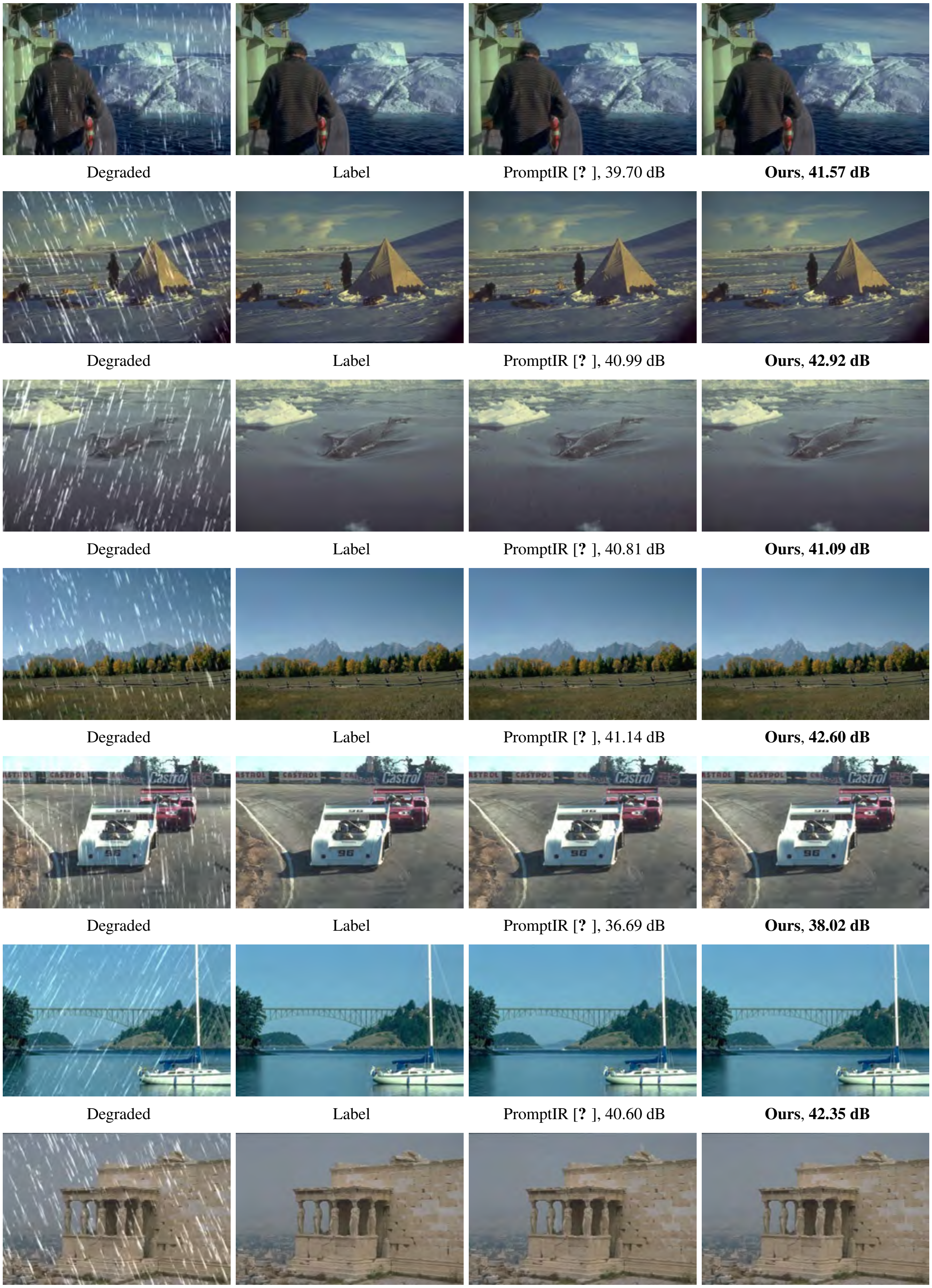
Degraded | Label | PromptIR [?], 37.68 dB | **Ours, 39.06 dB**

Degraded | Label | PromptIR [?], 38.50 dB | **Ours, 40.50 dB**

Degraded | Label | PromptIR [?], 36.59 dB | **Ours, 38.13 dB**

Degraded | Label | PromptIR [?], 36.02 dB | **Ours, 37.66 dB**

Figure J. Image Deraining on Rain100L [**?**].



Degraded Label PromptIR [?], 39.70 dB **Ours**, **41.57 dB**

Degraded Label PromptIR [?], 40.99 dB **Ours**, **42.92 dB**

Degraded Label PromptIR [?], 40.81 dB **Ours**, **41.09 dB**

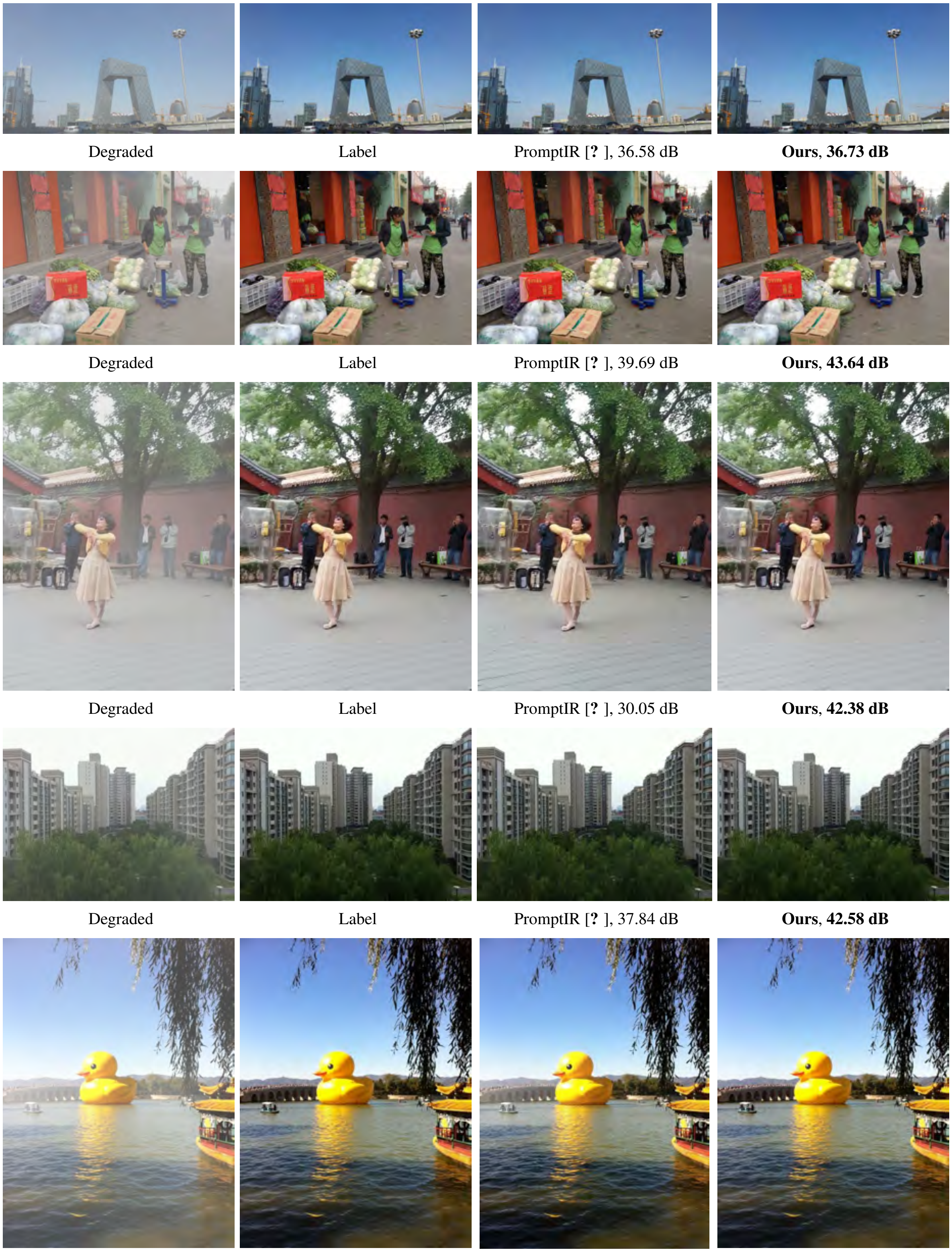
Degraded Label PromptIR [?], 41.14 dB **Ours**, **42.60 dB**

Degraded Label PromptIR [?], 36.69 dB **Ours**, **38.02 dB**

Degraded Label PromptIR [?], 40.60 dB **Ours**, **42.35 dB**

Degraded Label PromptIR [?], 40.11 dB **Ours**, **40.95 dB**

Figure K. Image Deraining on Rain100L [?].



Degraded | Label | PromptIR [?], 36.58 dB | **Ours**, **36.73 dB**

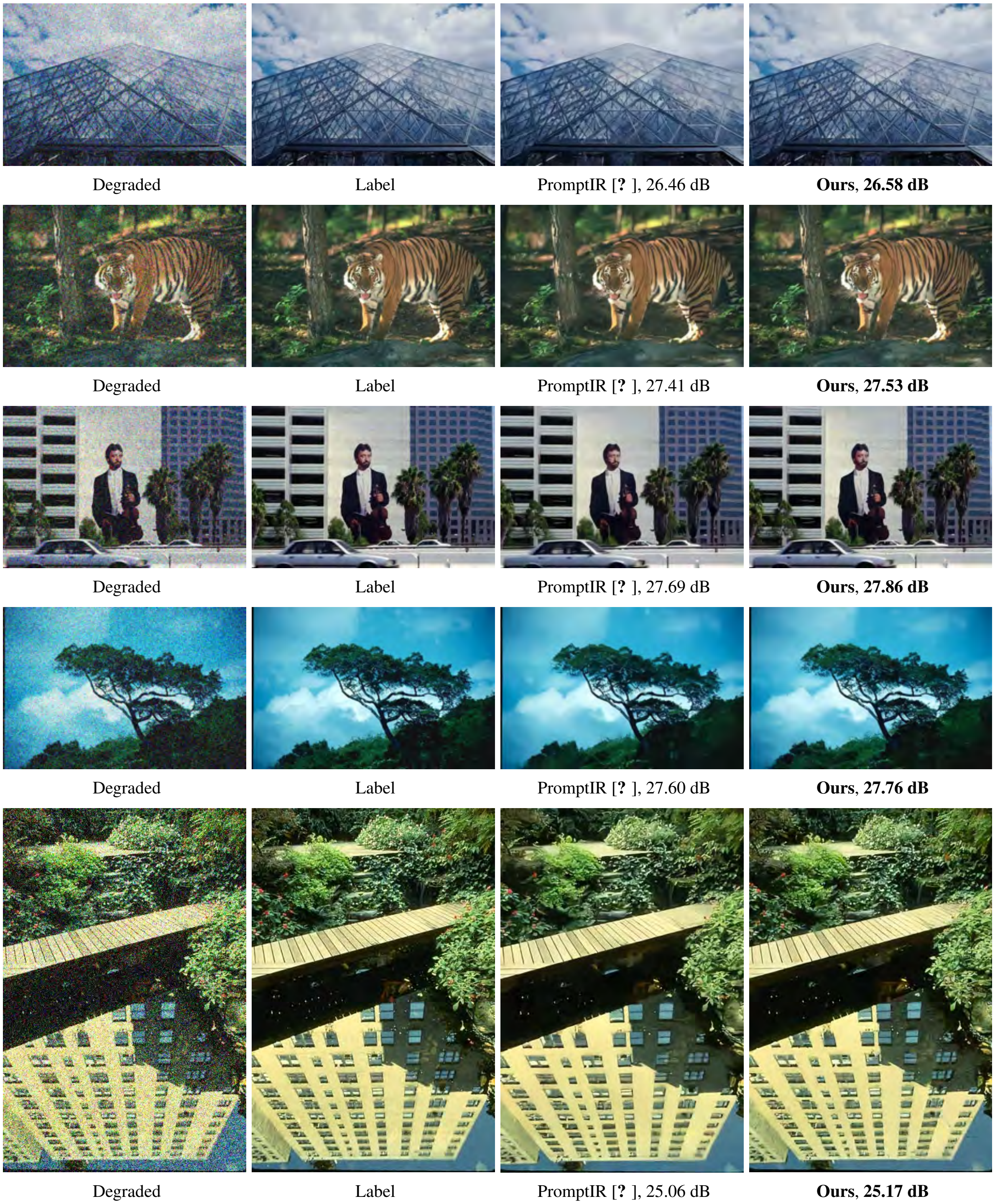
Degraded | Label | PromptIR [?], 39.69 dB | **Ours**, **43.64 dB**

Degraded | Label | PromptIR [?], 30.05 dB | **Ours**, **42.38 dB**

Degraded | Label | PromptIR [?], 37.84 dB | **Ours**, **42.58 dB**

Degraded | Label | PromptIR [?], 26.53 dB | **Ours**, **36.72 dB**

Figure L. Image Dehazing on SOTS-outdoor [?].



Degraded

Label

PromptIR [?], 26.46 dB

Ours, **26.58 dB**

Degraded

Label

PromptIR [?], 27.41 dB

Ours, **27.53 dB**

Degraded

Label

PromptIR [?], 27.69 dB

Ours, **27.86 dB**

Degraded

Label

PromptIR [?], 27.60 dB

Ours, **27.76 dB**

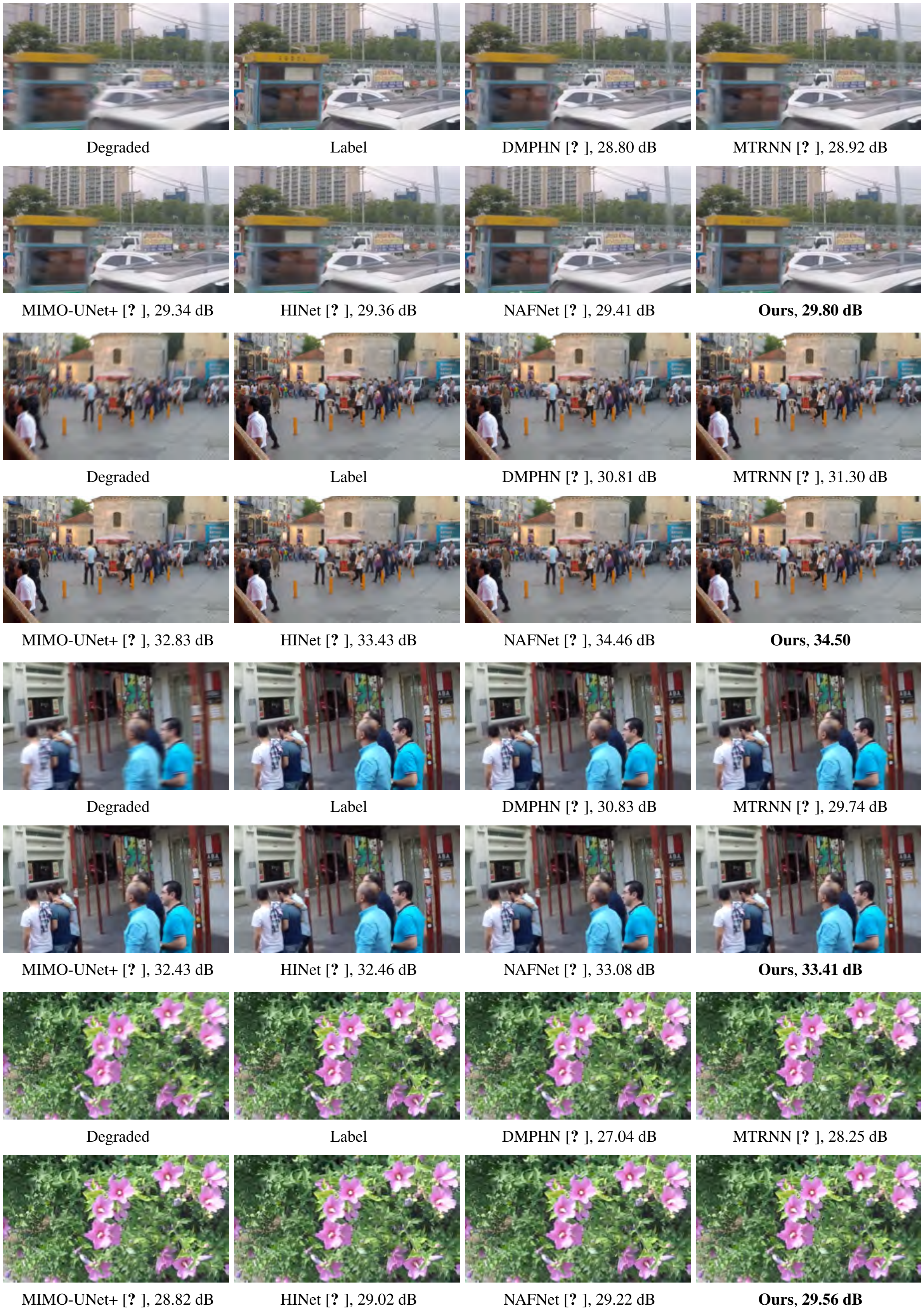
Degraded

Label

PromptIR [?], 25.06 dB

Ours, **25.17 dB**

Figure M. Image Denoising on CBSD68 [?].



Degraded

Label

DMPHN [?], 28.80 dB

MTRNN [?], 28.92 dB

MIMO-UNet+ [?], 29.34 dB

HINet [?], 29.36 dB

NAFNet [?], 29.41 dB

Ours, **29.80 dB**

Degraded

Label

DMPHN [?], 30.81 dB

MTRNN [?], 31.30 dB

MIMO-UNet+ [?], 32.83 dB

HINet [?], 33.43 dB

NAFNet [?], 34.46 dB

Ours, **34.50**

Degraded

Label

DMPHN [?], 30.83 dB

MTRNN [?], 29.74 dB

MIMO-UNet+ [?], 32.43 dB

HINet [?], 32.46 dB

NAFNet [?], 33.08 dB

Ours, **33.41 dB**

Degraded

Label

DMPHN [?], 27.04 dB

MTRNN [?], 28.25 dB

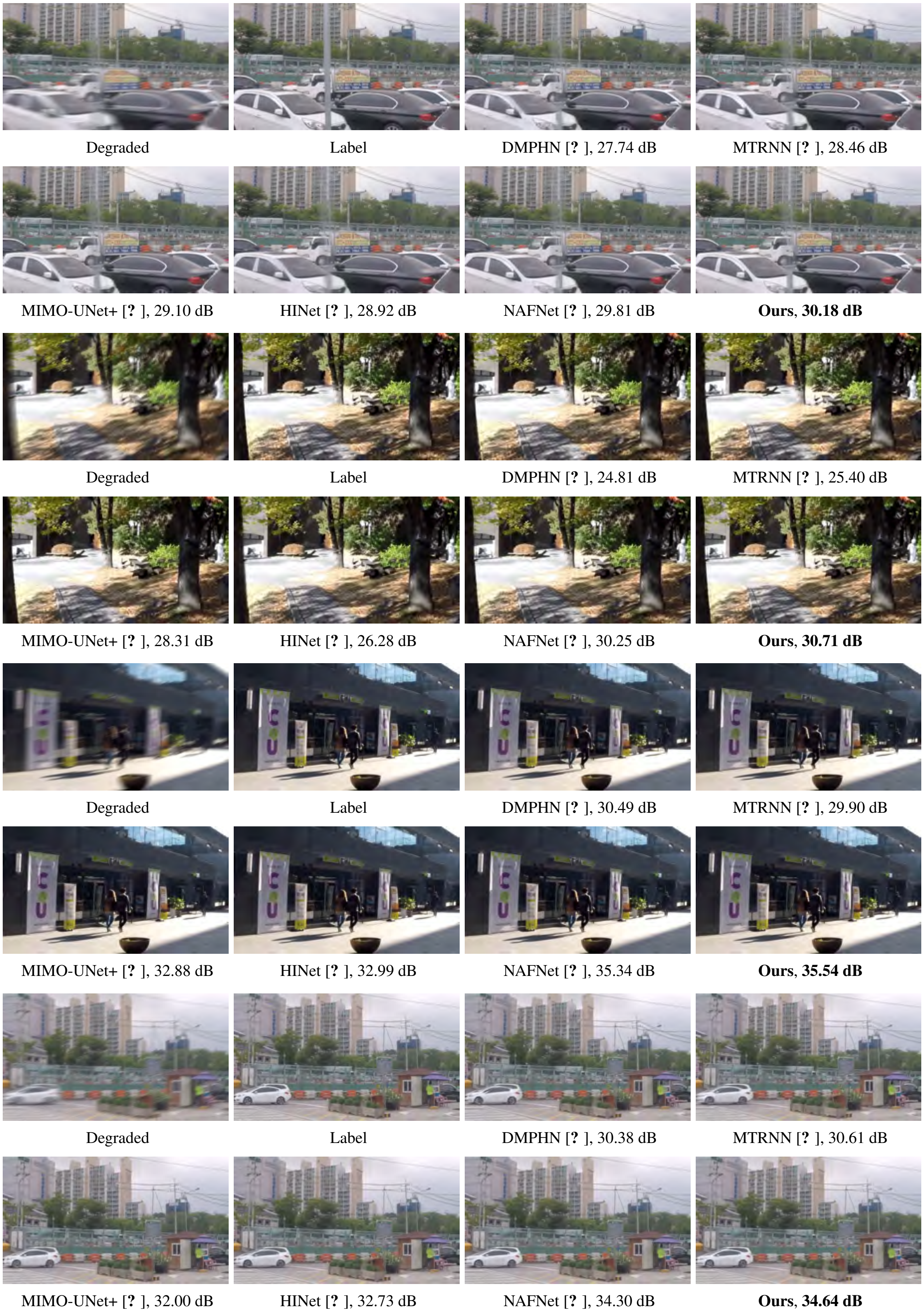
MIMO-UNet+ [?], 28.82 dB

HINet [?], 29.02 dB

NAFNet [?], 29.22 dB

Ours, **29.56 dB**

Figure N. Single-image motion deblurring on GoPro [?].



Degraded

Label

DMPHN [?], 27.74 dB

MTRNN [?], 28.46 dB

MIMO-UNet+ [?], 29.10 dB

HINet [?], 28.92 dB

NAFNet [?], 29.81 dB

Ours, **30.18 dB**

Degraded

Label

DMPHN [?], 24.81 dB

MTRNN [?], 25.40 dB

MIMO-UNet+ [?], 28.31 dB

HINet [?], 26.28 dB

NAFNet [?], 30.25 dB

Ours, **30.71 dB**

Degraded

Label

DMPHN [?], 30.49 dB

MTRNN [?], 29.90 dB

MIMO-UNet+ [?], 32.88 dB

HINet [?], 32.99 dB

NAFNet [?], 35.34 dB

Ours, **35.54 dB**

Degraded

Label

DMPHN [?], 30.38 dB

MTRNN [?], 30.61 dB

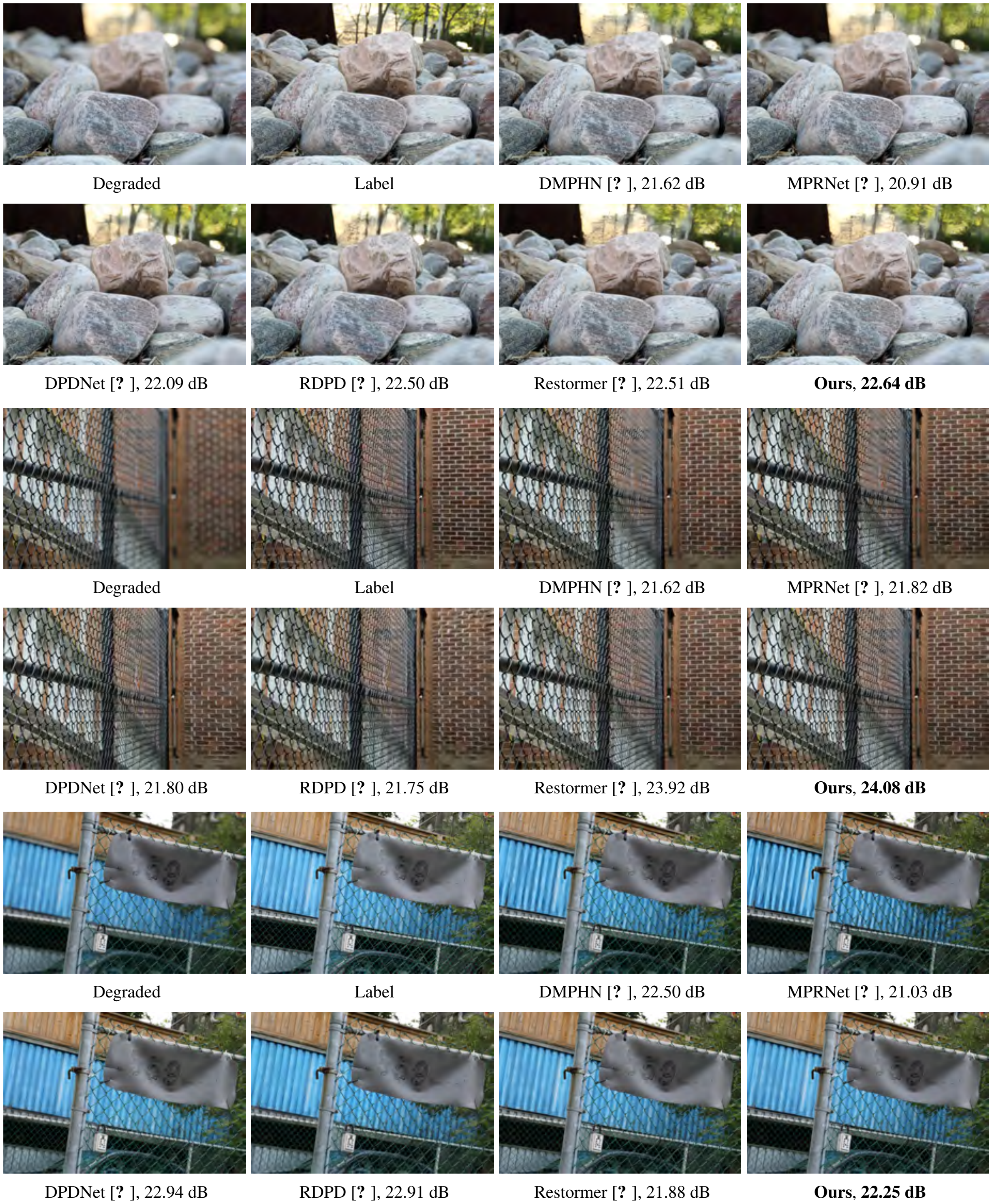
MIMO-UNet+ [?], 32.00 dB

HINet [?], 32.73 dB

NAFNet [?], 34.30 dB

Ours, **34.64 dB**

Figure O. Single-image motion deblurring on GoPro [?].



Degraded

Label

DMPHN [?], 21.62 dB

MPRNet [?], 20.91 dB

DPDNet [?], 22.09 dB

RDPD [?], 22.50 dB

Restormer [?], 22.51 dB

Ours, **22.64 dB**

Degraded

Label

DMPHN [?], 21.62 dB

MPRNet [?], 21.82 dB

DPDNet [?], 21.80 dB

RDPD [?], 21.75 dB

Restormer [?], 23.92 dB

Ours, **24.08 dB**

Degraded

Label

DMPHN [?], 22.50 dB

MPRNet [?], 21.03 dB

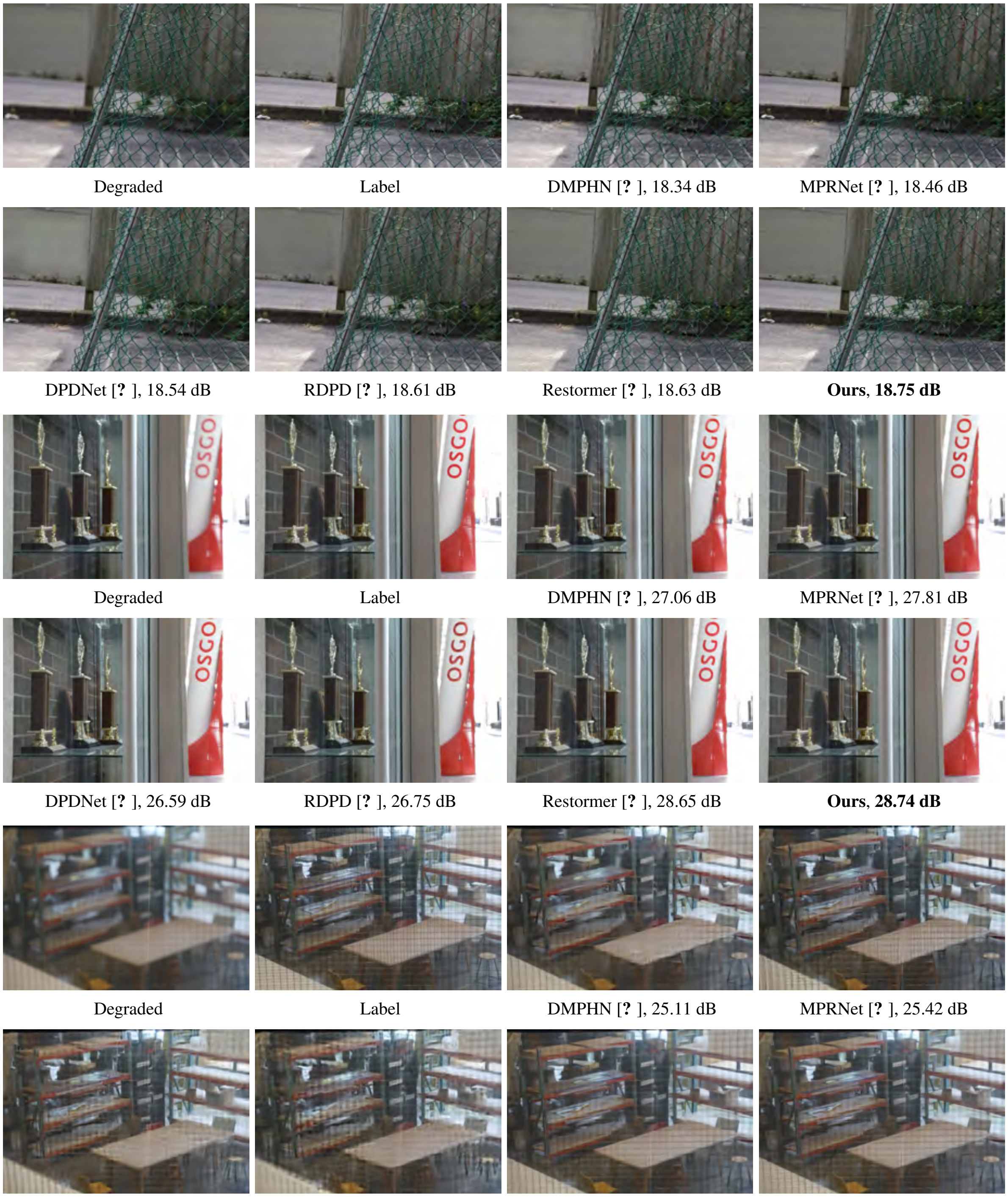
DPDNet [?], 22.94 dB

RDPD [?], 22.91 dB

Restormer [?], 21.88 dB

Ours, **22.25 dB**

Figure P. Defocus deblurring on DPDD [?].



Degraded

Label

DMPHN [?], 18.34 dB

MPRNet [?], 18.46 dB

DPDNet [?], 18.54 dB

RDPD [?], 18.61 dB

Restormer [?], 18.63 dB

Ours, **18.75 dB**

Degraded

Label

DMPHN [?], 27.06 dB

MPRNet [?], 27.81 dB

DPDNet [?], 26.59 dB

RDPD [?], 26.75 dB

Restormer [?], 28.65 dB

Ours, **28.74 dB**

Degraded

Label

DMPHN [?], 25.11 dB

MPRNet [?], 25.42 dB

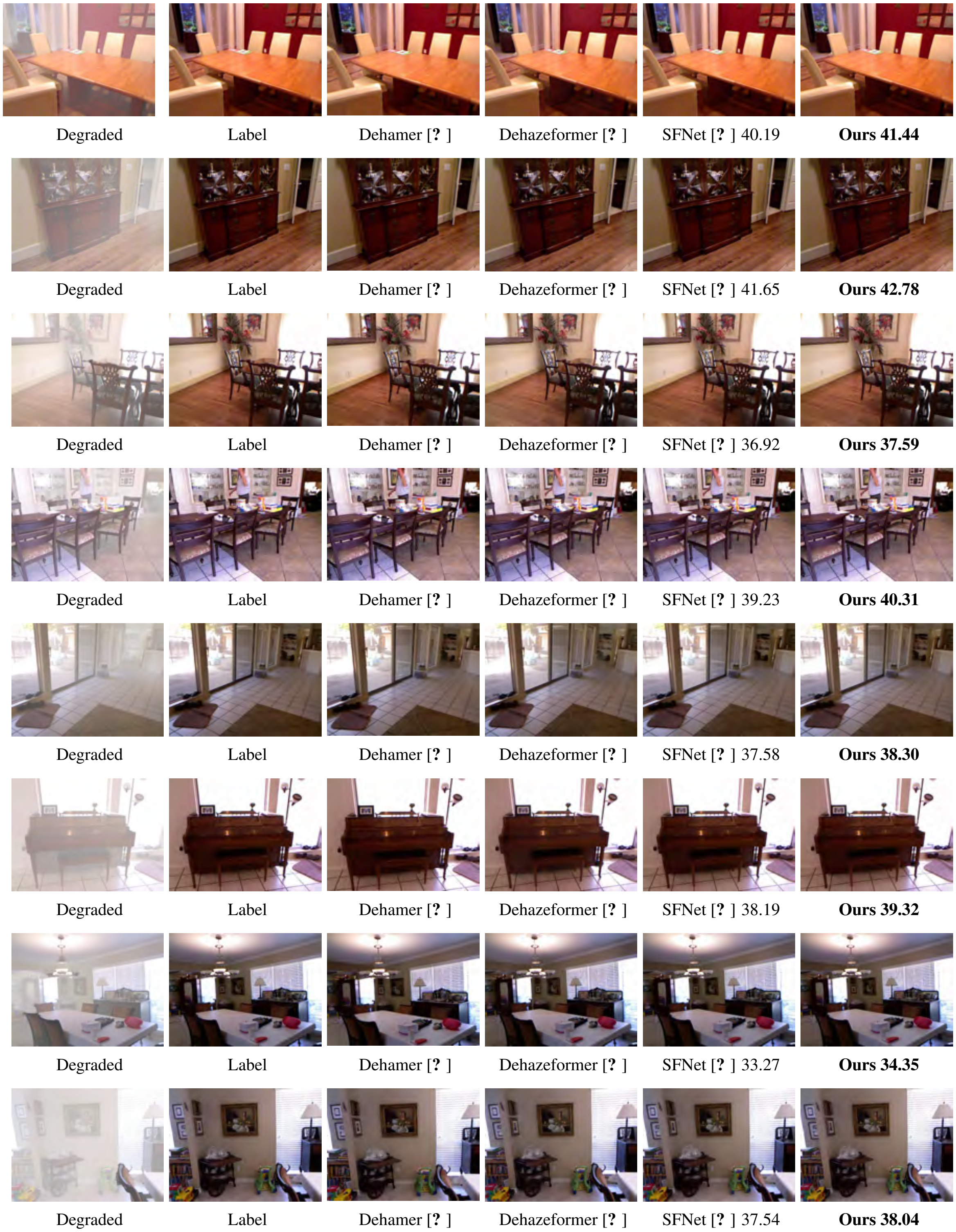
DPDNet [?], 24.01 dB

RDPD [?], 23.09 dB

Restormer [?], 26.43 dB

Ours, **26.58 dB**

Figure Q. Defocus deblurring on DPDD [?].



Degraded Label Dehamer [?] Dehazeformer [?] SFNet [?] 40.19 **Ours 41.44**

Degraded Label Dehamer [?] Dehazeformer [?] SFNet [?] 41.65 **Ours 42.78**

Degraded Label Dehamer [?] Dehazeformer [?] SFNet [?] 36.92 **Ours 37.59**

Degraded Label Dehamer [?] Dehazeformer [?] SFNet [?] 39.23 **Ours 40.31**

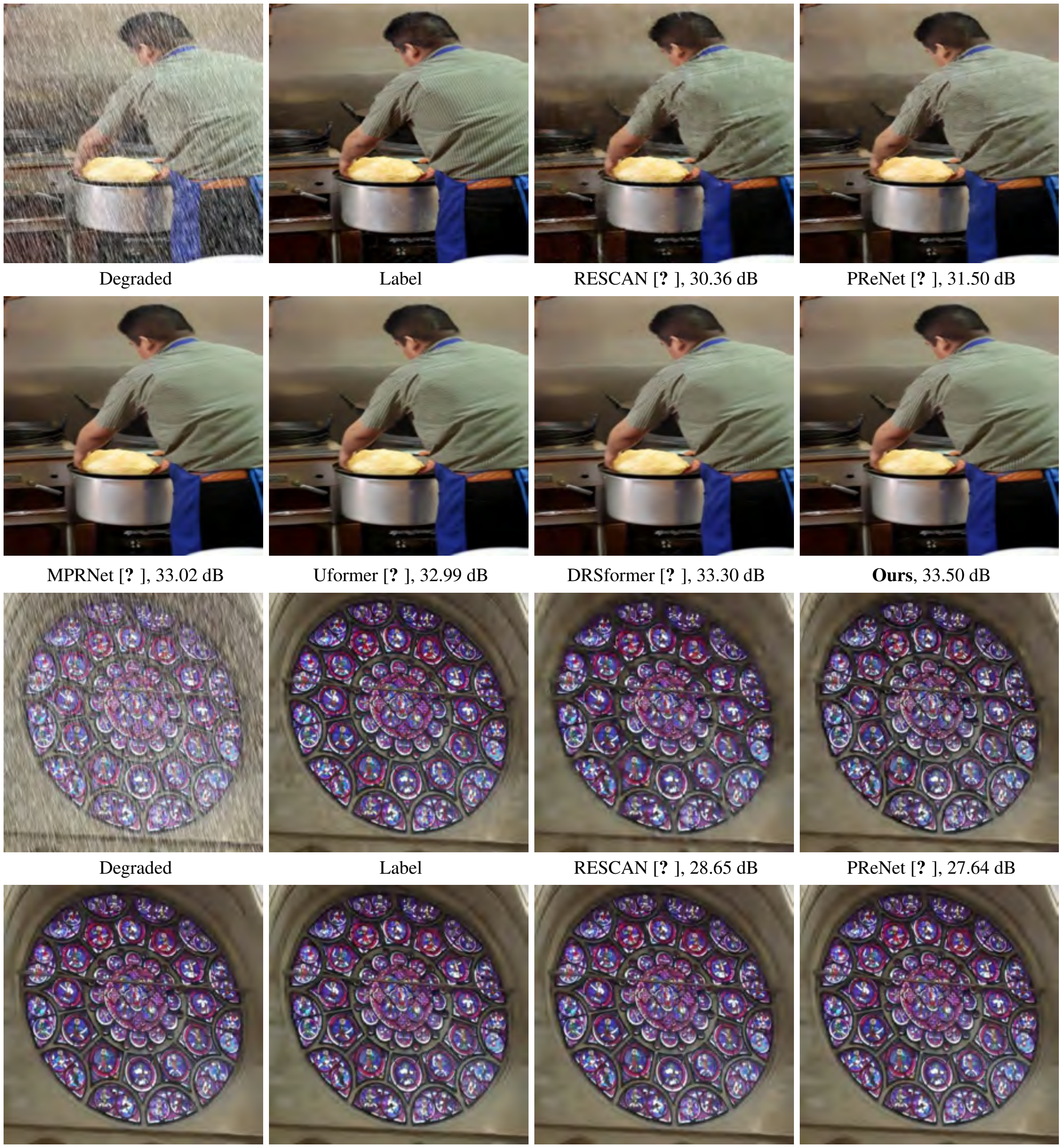
Degraded Label Dehamer [?] Dehazeformer [?] SFNet [?] 37.58 **Ours 38.30**

Degraded Label Dehamer [?] Dehazeformer [?] SFNet [?] 38.19 **Ours 39.32**

Degraded Label Dehamer [?] Dehazeformer [?] SFNet [?] 33.27 **Ours 34.35**

Degraded Label Dehamer [?] Dehazeformer [?] SFNet [?] 37.54 **Ours 38.04**

Figure R. Image dehazing results on SOTS [?].



Degraded

Label

RESCAN [?], 30.36 dB

PReNet [?], 31.50 dB

MPRNet [?], 33.02 dB

Uformer [?], 32.99 dB

DRSformer [?], 33.30 dB

Ours, 33.50 dB

Degraded

Label

RESCAN [?], 28.65 dB

PReNet [?], 27.64 dB

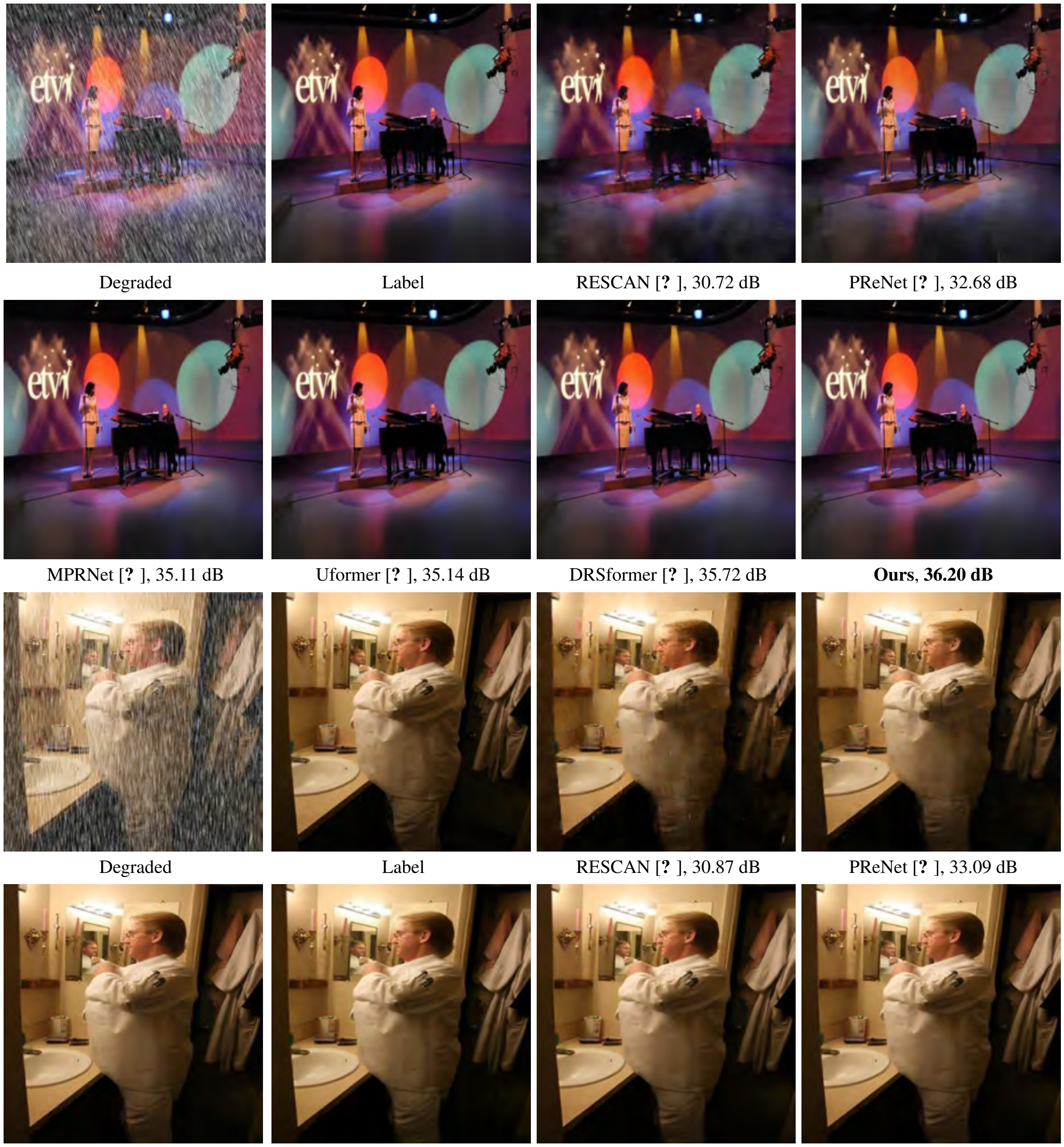
MPRNet [?], 29.35 dB

Uformer [?], 29.26 dB

DRSformer [?], 29.82 dB

Ours, 30.07 dB

Figure S. Image deraining results on DID-Data [?]



Degraded

Label

RESCAN [**?**], 30.72 dB

PReNet [**?**], 32.68 dB

MPRNet [**?**], 35.11 dB

Uformer [**?**], 35.14 dB

DRSformer [**?**], 35.72 dB

Ours, **36.20 dB**

Degraded

Label

RESCAN [**?**], 30.87 dB

PReNet [**?**], 33.09 dB

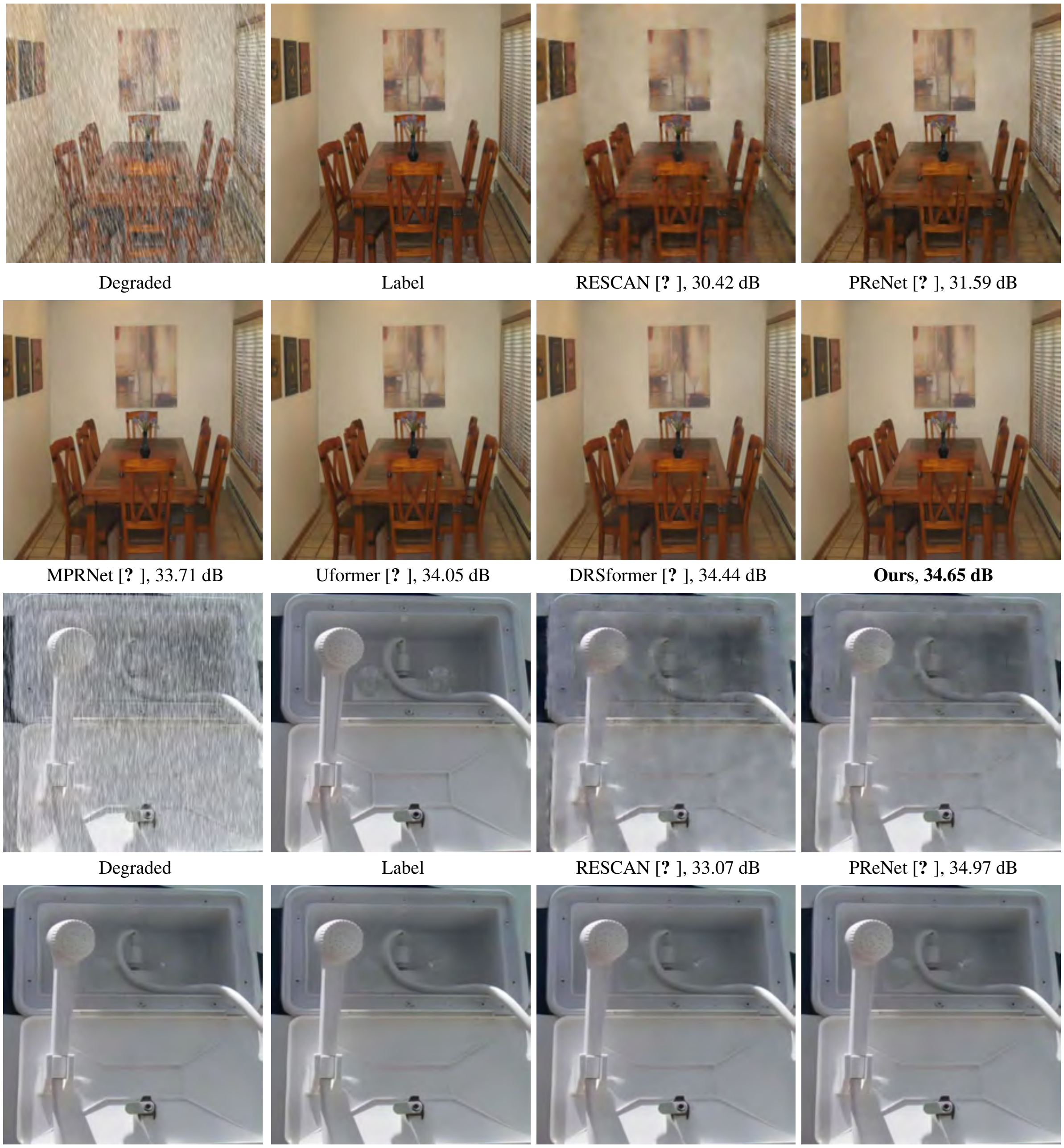
MPRNet [**?**], 35.08 dB

Uformer [**?**], 34.98 dB

DRSformer [**?**], 35.60 dB

Ours, **35.93 dB**

Figure T. Image deraining results on DID-Data [**?**]



Degraded

Label

RESCAN [**?**], 30.42 dB

PReNet [**?**], 31.59 dB

MPRNet [**?**], 33.71 dB

Uformer [**?**], 34.05 dB

DRSformer [**?**], 34.44 dB

Ours, **34.65 dB**

Degraded

Label

RESCAN [**?**], 33.07 dB

PReNet [**?**], 34.97 dB

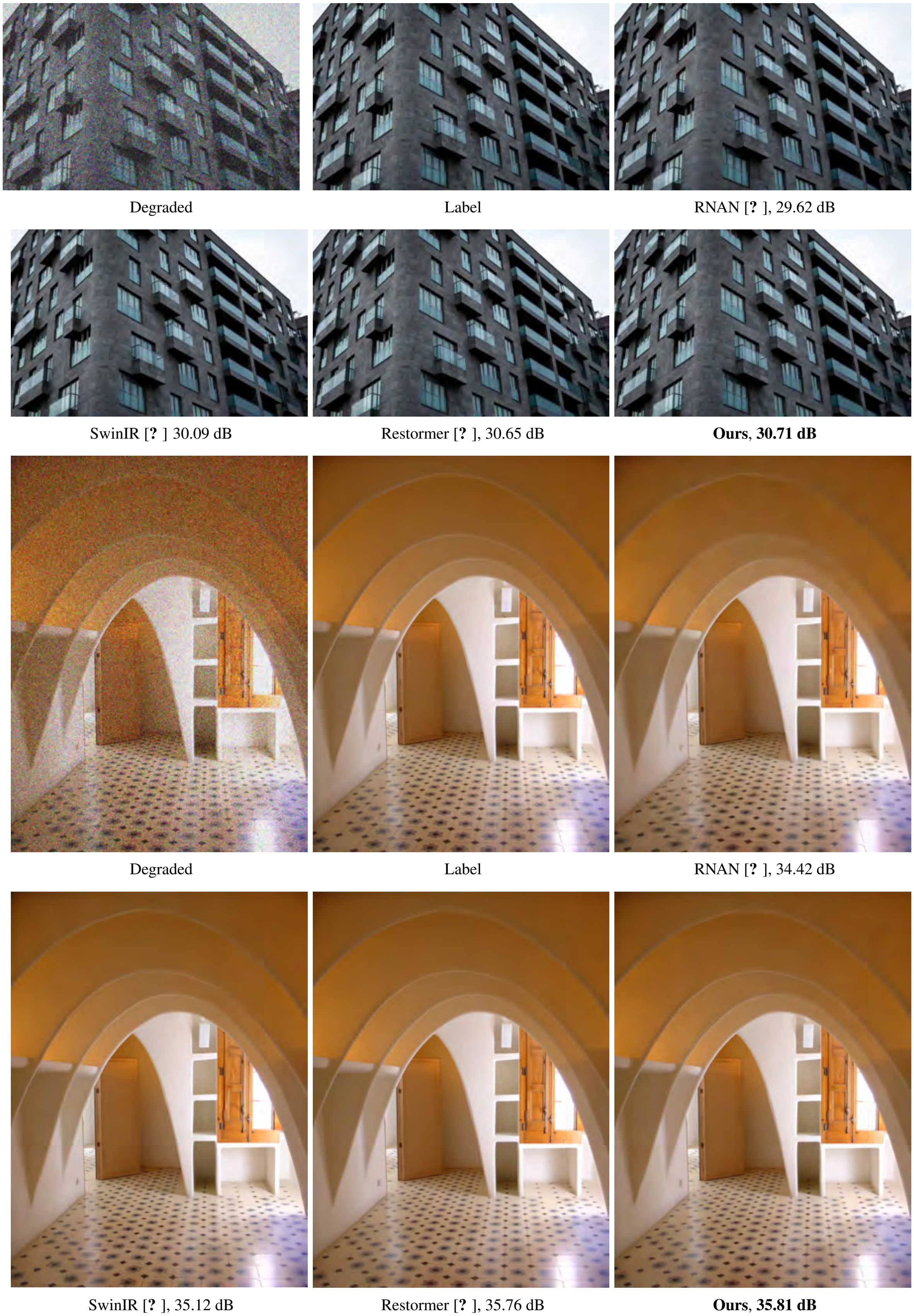
MPRNet [**?**] , 37.27 dB

Uformer [**?**], 37.05 dB

DRSformer [**?**], 37.77 dB

Ours, **38.91 dB**

Figure U. Image deraining results on DID-Data [**?**]



Degraded

Label

RNAN [?], 29.62 dB

SwinIR [?] 30.09 dB

Restormer [?], 30.65 dB

Ours, **30.71 dB**

Degraded

Label

RNAN [?], 34.42 dB

SwinIR [?], 35.12 dB

Restormer [?], 35.76 dB

Ours, **35.81 dB**

Figure V. Gaussian color denoising results on Urban100 [?].