Generative Diffusion Prior for Unified Image Restoration and Enhancement

Ben Fei[1,2,*], Zhaoyang Lyu[2,*], Liang Pan[3], Junzhe Zhang[3],
Weidong Yang[1,†], Tianyue Luo[1], Bo Zhang[2], Bo Dai[2,†]
[1] Fudan University, [2]Shanghai AI Laboratory, [3]S-Lab, Nanyang Technological University
bfei21@m.fudan.edu.cn, wdyang@fudan.edu.cn, (lvzhaoyang,daibo)@pjlab.org.cn

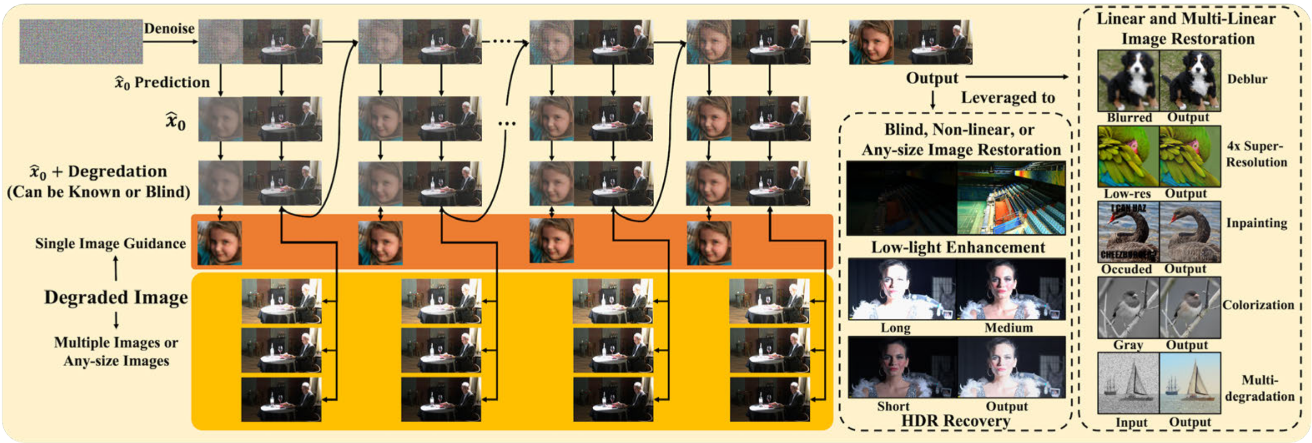


Figure 1. **Illustration of our GDP method for unified image recovery**, including linear inverse problems (Deblurring, 4× super-resolution, inpainting, and colorization), multi-degradation (*i.e.* Colorization + inpainting), non-linear and blind problems (Low-light enhancement and HDR recovery). Note that GDP can restore images of arbitrary sizes, and can accept multiple low-quality images as guidance as in the case of HDR recovery. GDP fulfills all the tasks using a single unconditional DDPM pre-trained on ImageNet.

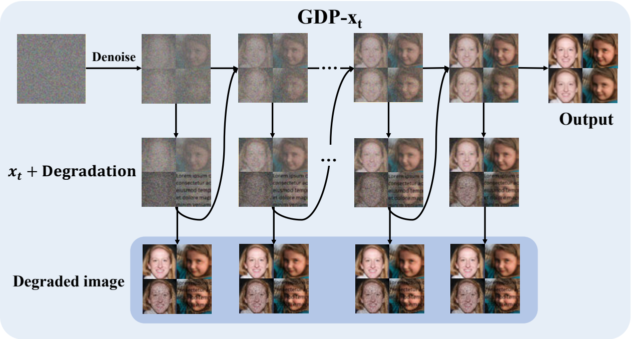


Figure 2. **Overview of the GDP-$\boldsymbol{x}_t$.** The guidance will be added on the noisy image $\boldsymbol{x}_t$ in every time step.

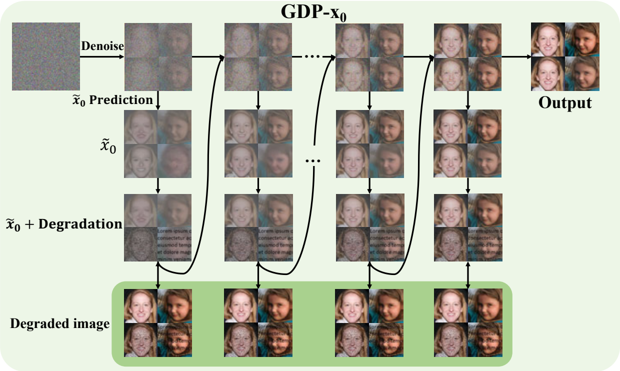


Figure 3. **Overview of the GDP-$\boldsymbol{x}_0$.** The guidance will be applied to a clean image $\tilde{\boldsymbol{x}}_0$ predicted from the noisy image $\boldsymbol{x}_t$.

A. Limitations and Future works

Limitations. The main limitation of our work is its inference time. Since we might add several guidance steps in every time step t, the sampling time is extended. This limits the applicability of our method to real-time applications and weak end-user devices such as mobile devices. To address this issue, further research into accelerated diffusion sampling techniques is required.

In addition, the choice of the guidance scale is also obtained through experiments, which means that for samples

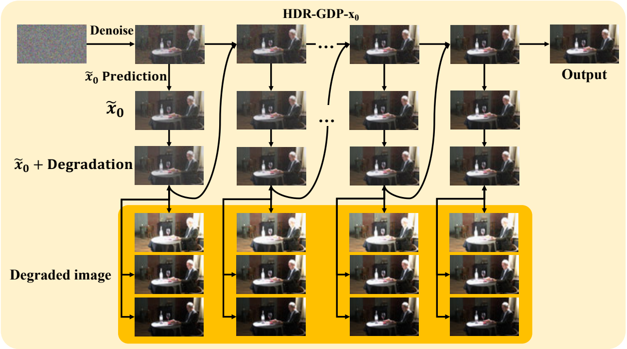


Figure 4. **Overview of the HDR-GDP-x_0.** The guidance will also be applied to a clean image $\tilde{\boldsymbol{x}}_0$. Unlike the GDP-x_0, three degraded images are utilized to guide the reverse process, and three sets of degradation models are optimized along the reverse process.

```
Algorithm 1: GDP-$\boldsymbol{x}_t$: Conditioner guided diffusion sampling on $\boldsymbol{x}_t$, given a diffusion model $(\mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t))$, corrupted image conditioner $\boldsymbol{y}$.
Input: Corrupted image $\boldsymbol{y}$, gradient scale $s$,
       degradation model $\mathcal{D}_\phi$ with randomly
       initiated parameters $\phi$, learning rate $l$ for
       optimizable degradation model, distance
       measure $\mathcal{L}$.
Output: Output image $\boldsymbol{x}_0$ conditioned on $\boldsymbol{y}$
Sample $\boldsymbol{x}_T$ from $\mathcal{N}(0, \mathbf{I})$
for t from T to 1 do
    $\mu, \Sigma = \mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t)$
    $\mathcal{L}^{total}_{\phi,\boldsymbol{x}_t} = \mathcal{L}(\boldsymbol{y}, \mathcal{D}_\phi(\boldsymbol{x}_t)) + \mathcal{Q}(\boldsymbol{x}_t)$
    $\phi \leftarrow \phi - l\nabla_\phi \mathcal{L}^{total}_{\phi,\boldsymbol{x}_t}$
    Sample $\boldsymbol{x}_{t-1}$ by $\mathcal{N}\left(\mu + s\nabla_{\boldsymbol{x}_t}\mathcal{L}^{total}_{\phi,\boldsymbol{x}_t}, \Sigma\right)$
end
return $\boldsymbol{x}_0$
```

with different distributions, it is necessary to manually select the optimal guidance scale. However, we found that for the same distribution of data, an approximate degradation model may lead to close guidance scales. This phenomenon may be proved mathematically in future work.

Future works. In future work, in addition to further optimizing the time step and variance schedules, it would be interesting to investigate the following:

(i) The Guided Diffusion Prior can also theoretically be applied to 3D data restoration. For instance, point cloud completion and upsampling can be regarded as linear inverse problems in 3D vision. Shapeinversion [27] tackles the point cloud completion by GAN inversion, where the GDP can hopefully be integrated.

(ii) Moreover, since LiDAR is affected by various kinds of weather in the real world and also produces various nonlinear degradations, GDP should also be explored for the recovery of these point clouds.

(iii) Self-supervised training techniques inspired by our GDP and techniques used in supervised techniques [17]

```
Algorithm 2: GDP-$\boldsymbol{x}_0$: Conditioner guided diffusion sampling on $\tilde{\boldsymbol{x}}_0$, given a diffusion model $(\mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t))$, corrupted image conditioner $\boldsymbol{y}$.
Input: Corrupted image $\boldsymbol{y}$, gradient scale $s$,
       degradation model $\mathcal{D}$, distance measure $\mathcal{L}$.
Output: Output image $\boldsymbol{x}_0$ conditioned on $\boldsymbol{y}$
Sample $\boldsymbol{x}_T$ from $\mathcal{N}(0, \mathbf{I})$
for t from T to 1 do
    $\mu, \Sigma = \mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t)$
    $\tilde{\boldsymbol{x}}_0 = \frac{\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\boldsymbol{x}_t,t)}{\sqrt{\bar{\alpha}_t}}$
    $\mathcal{L}^{total}_{\tilde{\boldsymbol{x}}_0} = \mathcal{L}(\boldsymbol{y}, \mathcal{D}(\tilde{\boldsymbol{x}}_0)) + \mathcal{Q}(\tilde{\boldsymbol{x}}_0)$
    Sample $\boldsymbol{x}_{t-1}$ by $\mathcal{N}\left(\mu + s\nabla_{\tilde{\boldsymbol{x}}_0}\mathcal{L}^{total}_{\tilde{\boldsymbol{x}}_0}, \Sigma\right)$
end
return $\boldsymbol{x}_0$
```

```
Algorithm 3: GDP-$\boldsymbol{x}_0$: Conditioner guided diffusion sampling on $\boldsymbol{x}_0$, given a diffusion model $(\mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t))$, corrupted images conditioner $\{\boldsymbol{y}^i \mid i = 1, 2, \ldots, n\}$.
Input: Corrupted image $\{\boldsymbol{y}^i \mid i = 1, 2, \ldots, n\}$ ($n$ =
       3 for HDR recovery (LDR-long image $\boldsymbol{y}^1$,
       LDR-medium image $\boldsymbol{y}^2$, LDR-short image
       $\boldsymbol{y}^3$) and $n$ = 1 for other tasks), gradient scale
       $s$, degradation models $\{\mathcal{D}_{\phi^i} | i = 1, 2, \ldots, n\}$
       with randomly initiated parameters
       $\{\phi^i | i = 1, 2, \ldots, n\}$, learning rate $l$ for
       optimizable degradation model, distance
       measure $\mathcal{L}$.
Output: Output image $\boldsymbol{x}_0$ conditioned on
        $\{\boldsymbol{y}^i \mid i = 1, 2, \ldots, n\}$
Sample $\boldsymbol{x}_T$ from $\mathcal{N}(0, \mathbf{I})$
for t from T to 1 do
    $\mu, \Sigma = \mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t)$
    $\tilde{\boldsymbol{x}}_0 = \frac{\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\boldsymbol{x}_t,t)}{\sqrt{\bar{\alpha}_t}}$
    $\mathcal{L}^{total}_{\phi,\tilde{\boldsymbol{x}}_0} = 0$
    for j from 1 to n do
        $\mathcal{L}_{\phi^j,\tilde{\boldsymbol{x}}_0} = \mathcal{L}(\boldsymbol{y}^j, \mathcal{D}_{\phi^j}(\tilde{\boldsymbol{x}}_0)) + \mathcal{Q}(\tilde{\boldsymbol{x}}_0)$
        $\phi^j = \phi^j - l\nabla_{\phi^j}\mathcal{L}_{\phi^j,\tilde{\boldsymbol{x}}_0}$
        $\mathcal{L}^{total}_{\phi,\tilde{\boldsymbol{x}}_0} = \mathcal{L}^{total}_{\phi,\tilde{\boldsymbol{x}}_0} + \mathcal{L}_{\phi^j,\tilde{\boldsymbol{x}}_0}$
    end
    Sample $\boldsymbol{x}_{t-1}$ by $\mathcal{N}\left(\mu + s\nabla_{\tilde{\boldsymbol{x}}_0}\mathcal{L}^{total}_{\phi,\tilde{\boldsymbol{x}}_0}, \Sigma\right)$
end
return $\boldsymbol{x}_0$
```

Algorithm 4: Restore Any-size Image

```
Input: Conditioner guided diffusion sampling on $\tilde{\boldsymbol{x}}_0$, given a
       diffusion model $(\mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t))$, corrupted image
       conditioner $\boldsymbol{y}$, degradation model $\mathcal{D}_\phi : \boldsymbol{y} = f\boldsymbol{x} + \boldsymbol{\mathcal{M}}$
       with randomly initiated parameters $\phi$, learning rate $l$ for
       optimizable degradation model. Dictionary of $K$
       overlapping patch locations, and a binary patch mask $\mathbf{P}^k$.
Output: Output image $\boldsymbol{x}_0$ conditioned on $\boldsymbol{y}$
Sample $\boldsymbol{x}_T$ from $\mathcal{N}(0, \mathbf{I})$
for $t$ from $T$ to $1$ do
    $\mu, \Sigma = \mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t)$
    Mean vector $\boldsymbol{\Omega}_t = \mathbf{0}$ and variance vector $\boldsymbol{\psi}_t =$
      $\mathbf{0}$ and weight vector $\mathbf{G} = \mathbf{0}$ and $f = \mathbf{0}$ and $\boldsymbol{\mathcal{M}} = \mathbf{0}$
    for $k = 1, \ldots, K$ do
        $\boldsymbol{x}_t^k = \mathrm{Crop}\left(\mathbf{P}^k \circ \boldsymbol{x}_t\right)$
        $\boldsymbol{y}^k = \mathrm{Crop}\left(\mathbf{P}^k \circ \boldsymbol{y}\right)$
        $\boldsymbol{\mathcal{M}}^k = \mathrm{Crop}\left(\mathbf{P}^k \circ \boldsymbol{\mathcal{M}}\right)$
        $\tilde{\boldsymbol{x}}_0^k = \frac{\boldsymbol{x}_t^k}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\epsilon_\theta\left(\boldsymbol{x}_t^k, t\right)}{\sqrt{\bar{\alpha}_t}}$
        $\mathcal{L}_{\phi, \tilde{\boldsymbol{x}}_0^k}^{total} = \mathcal{L}(\boldsymbol{y}^k, \mathcal{D}_\phi\left(\tilde{\boldsymbol{x}}_0^k\right)) + \mathcal{Q}\left(\tilde{\boldsymbol{x}}_0^k\right)$
        $f^k \leftarrow f^k - l\nabla_{f^k}\mathcal{L}_{f^k, \tilde{\boldsymbol{x}}_0^k}^{total}$
        $\boldsymbol{\mathcal{M}}^k \leftarrow \boldsymbol{\mathcal{M}}^k - l\nabla_{\boldsymbol{\mathcal{M}}^k}\mathcal{L}_{\boldsymbol{\mathcal{M}}^k, \tilde{\boldsymbol{x}}_0^k}^{total}$
        $\mu^k = \mu + s\nabla_{\tilde{\boldsymbol{x}}_0^k}\mathcal{L}_{\phi, \tilde{\boldsymbol{x}}_0^k}^{total}$
        $f = f + f^k$
        $\boldsymbol{\Omega}_t = \boldsymbol{\Omega}_t + \mathbf{P}_k \cdot \mu^k$
        $\boldsymbol{\psi}_t = \psi_t + \mathbf{P}^k \cdot \sigma^k$
        $\boldsymbol{\mathcal{M}} = \boldsymbol{\mathcal{M}} + \mathbf{P}^k \cdot \boldsymbol{\mathcal{M}}^k$
        $\mathbf{G} = \mathbf{G} + \mathbf{P}^k$
    end
    $\boldsymbol{\Omega}_t = \boldsymbol{\Omega}_t \oslash \mathbf{G}$        $//\oslash :$ element-wise division
    $\psi_t = \psi_t \oslash \mathbf{G}$
    $\boldsymbol{\mathcal{M}} = \boldsymbol{\mathcal{M}} \oslash \mathbf{G}$
    $f = f/K$
    Sample $\boldsymbol{x}_{t-1}$ by $\mathcal{N}(\boldsymbol{\Omega}_t, \psi_t)$
end
return Restored any-size image $\boldsymbol{x}_0$
```

that further improve the performance of unsupervised image restoration models.

B. Implementation Details

We apply GDP to a suite of challenging image restoration tasks: (1) **Colorization** transforms an input gray-scale image to a plausible color image. (2) **Inpainting** fills in user-specified masked regions of an image with realistic content. (3) **Super-resolution** extends a low-resolution image into a higher one. (4) **Deblurring** corrects the blurred images, restoring plausible image detail. (5) **Enlighting** enables the dark images turned into normal images. (6) **HDR image recovery** aims to obtain HDR images with the aid of three LDR images. Inputs and outputs of the first four tasks are represented as 256×256 RGB images, while the last two tasks are various (1900×1060 for HDR image recovery and 600×400 for image enlightening, respectively). We do so without task-specific hyperparameter tuning and architecture customization.

Colorization requires the representation of objects, segmentation, and layouts with long-range image dependencies. Inpainting is challenging due to large masks, image diversity, and cluttered scenes. Super-resolution and deblurring are also not trivial because the degradation might damage the content of the images. While the other tasks are linear in nature, low-light enhancement and HDR recovery are non-linear inverse problems; they require a good model of natural image statistics to detect and correct over-exposed and under-exposed areas. Although previous works have studied these problems extensively, it is rare that a model with no task-specific engineering achieves strong performance in all tasks, beating strong task-specific GAN and regression baselines. Our GDP is devised to achieve this goal.

B.1. Dataset briefs

ImageNet, LSUN, CelebA, and USC-SIPI Datasets. To quantitatively evaluate GDP on linear image restoration tasks, we test on 1k images from the ImageNet validation set following [15]. The CelebA-HQ [7] dataset is a high-quality subset of the Large-Scale CelebFaces Attributes (CelebA) dataset [10]. LSUN dataset [26] contains around one million labeled images for each of 10 scene categories and 20 object categories. And the USC-SIPI dataset [22] is a collection of various digitized images. We utilize the images from CelebA, LSUN, and USC-SIPI provided by [6].

LOL Dataset. The LOL dataset [23] is composed of 500 low-light and normal-light image pairs and divided into 485 training pairs and 15 testing pairs. The low-light images contain noise produced during the photo capture process. Most of the images are indoor scenes. All the images have a resolution of 400 × 600.

VE-LOL-L Dataset. For underexposure correction experiments, we use the paired data of the VE-LOL-L dataset [9], in which each captured well-exposed image has its underexposed version with different underexposure levels. Note that the VE-LOL-L dataset, consisting of VE-LOL-Cap and VE-LOL-Syn, is also carried out. Due to the different distribution of the two sub-set, we solve them under different guidance scales.

LoLi-Phone Dataset. LoLi-Phone [8] is a large-scale low-light image and video dataset for low-light image enhancement. The images and videos are taken by different mobile phone cameras under diverse illumination conditions.

NTIRE Dataset [16]. In the NTIRE dataset, there are 1494 LDRs/HDR for training, 60 images for validation, and 201 images for testing. The 1494 frames consist of 26 long shots. Each scene contains three LDR images, their corresponding exposure and alignment information, and HDR ground truth. The size of an image is 1060×1900. Since the ground truth of the validation and test sets are not avail-

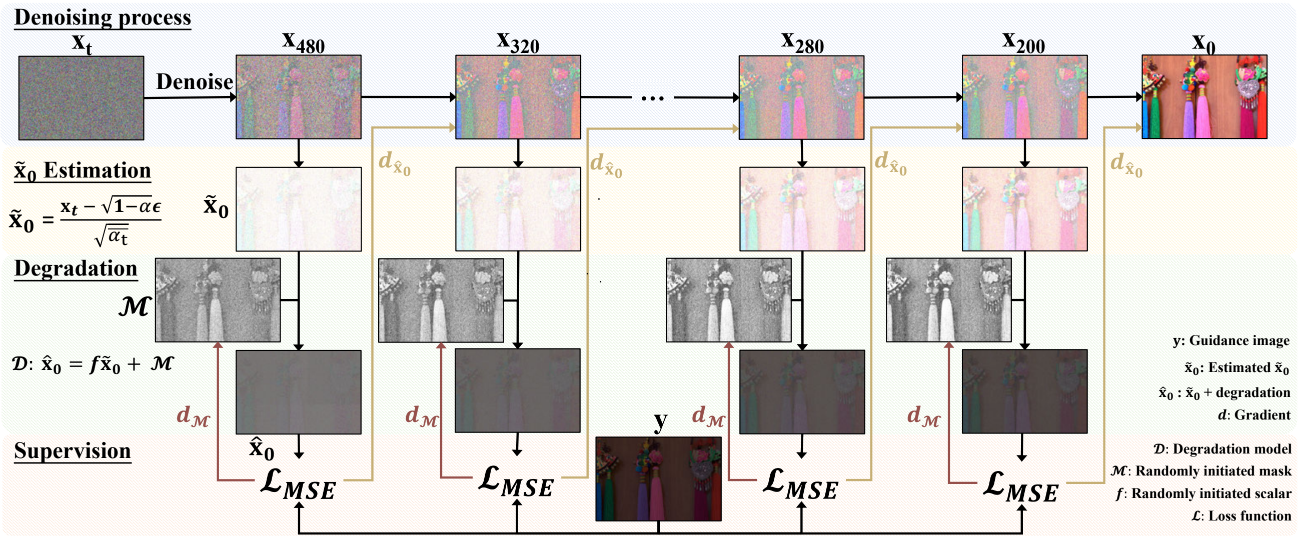
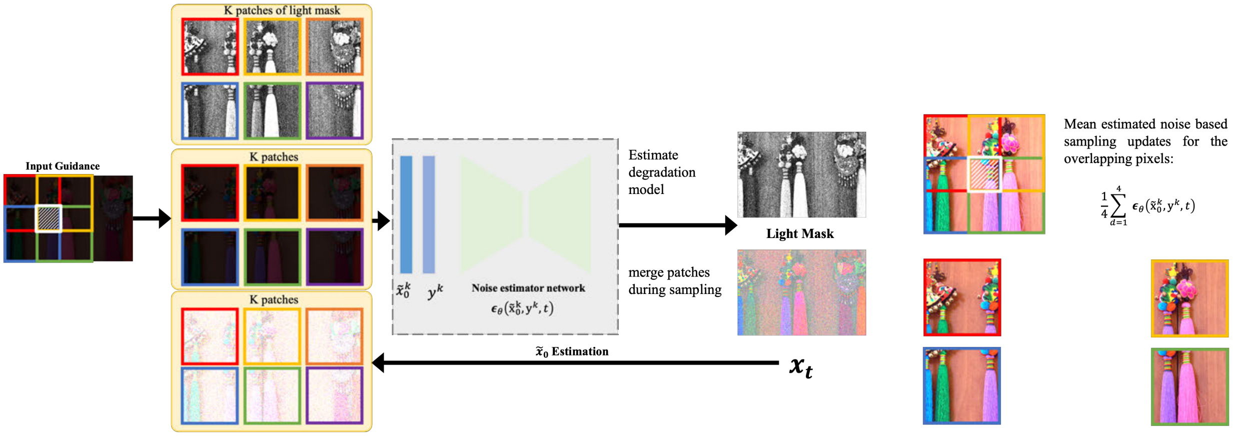

Denoising process
x_t
Denoise
x_{480}
x_{320}
x_{280}
x_{200}
x_0
$\tilde{x}_0$ Estimation
$\tilde{x}_0 = \frac{x_t - \sqrt{1-\alpha\epsilon}}{\sqrt{\alpha_t}}$
$\tilde{x}_0$
$d_{\hat{x}_0}$
Degradation
$\mathcal{M}$
$\mathcal{D}$: $\hat{x}_0 = f\tilde{x}_0 + \mathcal{M}$
$d_{\mathcal{M}}$
y
Supervision
$\hat{x}_0$
$\mathcal{L}_{MSE}$
y: Guidance image
$\tilde{x}_0$: Estimated $\tilde{x}_0$
$\hat{x}_0$: $\tilde{x}_0$ + degradation
d: Gradient
$\mathcal{D}$: Degradation model
$\mathcal{M}$: Randomly initiated mask
f: Randomly initiated scalar
$\mathcal{L}$: Loss function

Figure 5. Illustration of the patch-based method for any-size image restoration.


K patches of light mask
Input Guidance
K patches
K patches
$\tilde{x}_0^k$
y^k
Noise estimator network
$\epsilon_\theta(\tilde{x}_0^k, y^k, t)$
Estimate degradation model
Light Mask
merge patches during sampling
$\tilde{x}_0$ Estimation
x_t
Mean estimated noise based sampling updates for the overlapping pixels:
$\frac{1}{4}\sum_{d=1}^{4} \epsilon_\theta(\tilde{x}_0^k, y^k, t)$

(a) Patch-based diffusive image restoration

(b) Illustrating sampling for overlapping patches

Figure 6. (a) Illustration of the patch-based image restoration pipeline detailed in Algorithm 4. (b) Illustrating mean estimated noise-guided sampling updates for overlapping pixels across patches. We demonstrate a simplified example where $r = p/2$, r is the stride and p is the patch size of images. And there are only four overlapping patches sharing the grid cell marked with the white border and gratings. The pixels in this region would be updated at each denoising step t using the mean estimated noise over the four overlapping patches.

able, we only do experiments on the training set. We select 100 images as the test set.

B.2. Experimental Setup

In each inverse problem, the pixel values are in the range [0,1], and the resulting degradation measures are as follows: (i) For super-resolution, a block averaging filter is utilized to downscale the image on each axis 4 times; (ii) In terms of deblurring, the image is blurred by a 9×9 unified kernel. (iii) For colorization, the gray-scale image is the average of the red, green, and blue channels of the original image; (iv) For inpainting, we cover parts of the original image with text overlays or randomly delete 25% pixels.

In the non-linear and blind problem, the images from the low-light dataset and NTIRE dataset are naturally over-exposed or under-exposed. Therefore, no additional operations are required for the images.

C. Evaluation Metrics

Apart from the commonly used PSNR and SSIM, other metrics are also utilized for evaluation: (i) **FID** [4] is an objective metric used to assess the quality of synthesized images. (ii) **Consistency** [18] measures MSE between the degraded inputs and the outputs undergoing the same degradation. (iii) **Learned perceptual image patch similarity (LPIPS)** [28] is also adopted, a deep feature-based perceptual distance metric to further assess the image quality. (iv) The non-reference **perceptual index (PI)** [11] is also employed to evaluate perceptual quality. The PI metric is originally utilized to measure perceptual quality in image super-resolution. It has also been used to assess the performance of other image restoration tasks. A lower PI value indicates better perceptual quality. (v) The **lightness order error (LOE)** [21] is employed as our objective metric to measure the performance. The definition of LOE is as follows:

$$LOE=\frac{1}{m}\sum_{x=1}^{m}\sum_{y=1}^{m}\left(U(\mathbf{T}(x),\mathbf{T}(y))\oplus U\left(\mathbf{T}_r(x),\mathbf{T}_r(y)\right)\right) \tag{1}$$

where m is the pixel number. The function $U(p,q)$ returns 1 if $p >= q, 0$ otherwise. $\oplus$ stands for the exclusive-or operator. In addition, $\mathbf{T}(x)$ and $\mathbf{T}_r(x)$ are the maximum values among R, G and B channels at location x of the enhanced and reference images, respectively. The lower the LOE is, the better the enhancement preserves the naturalness of lightness.

D. Further elaboration of the models

GDP-$\boldsymbol{x}_t$. As shown in Fig. 2, the guidance is conditioned on $\boldsymbol{x}_t$ but with the absence of Σ. The noisy images are gradually denoised during the reverse process. And the $\boldsymbol{x}_t$ undergoing the degradation model is more similar to the corrupted image. The gradients ∇ of the loss function are utilized to control the mean of the conditional distribution.

GDP-$\boldsymbol{x}_0$. To make a clear comparison, we also illustrate the GDP-$\boldsymbol{x}_0$ in Fig. 3, and Algorithm 2 in the main paper. Different from the GDP-$\boldsymbol{x}_t$, GDP-$\boldsymbol{x}_0$ will predict the intermediate variance $\tilde{\boldsymbol{x}}_0$ from the noisy image $\boldsymbol{x}_t$ by estimating the noise in $\boldsymbol{x}_t$, which can be directly inferred when given the $\boldsymbol{x}_t$ in every time steps t. Then the predicted $\tilde{\boldsymbol{x}}_0$ goes through the same degradation as input to obtain $\hat{\boldsymbol{x}}_0$. Note that the degradation might be unknown. Then the loss between the $\hat{\boldsymbol{x}}_0$ and the corrupted image $\boldsymbol{y}$, the gradients will be applied to optimize the unknown degradation models and sample the next step latent $\boldsymbol{x}_{t-1}$.

HDR-GDP-$\boldsymbol{x}_0$. As depicted in Fig. 4, and Algorithm 3, there are three images to guide the reverse process. As a blind problem, we randomly initiate three sets of the parameters of the degradation models. At every time step, $\tilde{\boldsymbol{x}}_0$ will undergo the three degradation models $\mathcal{D}^i$, respectively. Unlike GDP-$\boldsymbol{x}_0$, the gradients of the three losses are used to optimize the corresponding degradation model and all leveraged to sample the next step latent $\boldsymbol{x}_{t-1}$.

Hierarchical Guidance and Patch-based Methods. As vividly illustrated in Fig. 5 and 6, we resize the corrupted images $\boldsymbol{y}\in\mathbb{R}^{3\times H\times W}$ to $\overline{\boldsymbol{y}}\in\mathbb{R}^{3\times 256\times\overline{W}\text{ or }3\times\overline{H}\times 256}$, then apply the patch-based methods [14] on the reshaped images. Following that, the light masks $\overline{\boldsymbol{\mathcal{M}}}$ are interpolated to the original image size to obtain the $\boldsymbol{\mathcal{M}}$, which can be regarded as the global light shift. After, the light factor $\boldsymbol{f}$ and the light mask $\boldsymbol{\mathcal{M}}$ will be fixed and utilized to generate the image patches of the original images, which will be finally recombined as the output images. In our experiments, low-light enhancement and HDR recovery problems can be tackled by this strategy.

E. Further Ablation Study on the Guidance

To gain insight into the way of guidance, apart from GDP-$\boldsymbol{x}_t$ and GDP-$\boldsymbol{x}_0$, two more variants GDP-$\boldsymbol{x}_t$-v1 and GDP-$\boldsymbol{x}_0$-v1 are devised for comparison.

The main difference among these four variants is the way of mean shift. The mean shift of four variants can be written as follow:

$$\begin{aligned}
&\text{GDP-}\boldsymbol{x}_0:\tilde{\boldsymbol{\mu}}_t\left(\boldsymbol{x}_t,\tilde{\boldsymbol{x}}_0\right)=\\
&\frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\tilde{\boldsymbol{x}}_0+\frac{\sqrt{\alpha_t}\left(1-\bar{\alpha}_{t-1}\right)}{1-\bar{\alpha}_t}\boldsymbol{x}_t+s\nabla_{\tilde{\boldsymbol{x}}_0}\mathcal{L}_{\tilde{\boldsymbol{x}}_0}^{total}\\
&\text{GDP-}\boldsymbol{x}_t:\tilde{\boldsymbol{\mu}}_t\left(\boldsymbol{x}_t,\tilde{\boldsymbol{x}}_0\right)=\\
&\frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\tilde{\boldsymbol{x}}_0+\frac{\sqrt{\alpha_t}\left(1-\bar{\alpha}_{t-1}\right)}{1-\bar{\alpha}_t}\boldsymbol{x}_t+s\nabla_{\boldsymbol{x}_t}\mathcal{L}_{\boldsymbol{x}_t}^{total}\\
&\text{GDP-}\boldsymbol{x}_0\text{-v1}:\tilde{\boldsymbol{\mu}}_t\left(\boldsymbol{x}_t,\tilde{\boldsymbol{x}}_0\right)=\\
&\frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}(\tilde{\boldsymbol{x}}_0+s\nabla_{\tilde{\boldsymbol{x}}_0}\mathcal{L}_{\tilde{\boldsymbol{x}}_0}^{total})+\frac{\sqrt{\alpha_t}\left(1-\bar{\alpha}_{t-1}\right)}{1-\bar{\alpha}_t}\boldsymbol{x}_t\\
&\text{GDP-}\boldsymbol{x}_t\text{-v1}:\tilde{\boldsymbol{\mu}}_t\left(\boldsymbol{x}_t,\tilde{\boldsymbol{x}}_0\right)=\\
&\frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\tilde{\boldsymbol{x}}_0+\frac{\sqrt{\alpha_t}\left(1-\bar{\alpha}_{t-1}\right)}{1-\bar{\alpha}_t}(\boldsymbol{x}_t+s\nabla_{\boldsymbol{x}_t}\mathcal{L}_{\boldsymbol{x}_t}^{total}).
\end{aligned} \tag{2}$$

where GDP-$\boldsymbol{x}_0$ directly add the mean shift $s\nabla_{\boldsymbol{x}_0}\mathcal{L}_{\boldsymbol{x}_0}^{total}$ into $\tilde{\boldsymbol{\mu}}_t\left(\boldsymbol{x}_t,\tilde{\boldsymbol{x}}_0\right)$ without the coefficient $\frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}$, compared with GDP-$\boldsymbol{x}_0$-v1.

It is experimentally found that GDP-$\boldsymbol{x}_0$ and GDP-$\boldsymbol{x}_t$ fulfills better performance on four linear tasks than GDP-$\boldsymbol{x}_0$-v1 and GDP-$\boldsymbol{x}_t$-v1 in Table. 2.

F. The ELBO objective of GDP

GDP is a Markov chain conditioned on $\boldsymbol{y}$, resulting in the following ELBO objective [20]:

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{x}_0\sim q(\boldsymbol{x}_0),\boldsymbol{y}\sim q(\boldsymbol{y}\mid\boldsymbol{x}_0)}\left[\log p_\theta\left(\boldsymbol{x}_0\mid\boldsymbol{y}\right)\right]\geq\\
&-\mathbb{E}\left[\sum_{t=1}^{T-1}\mathrm{KL}\left(q^t\left(\boldsymbol{x}_t\mid\boldsymbol{x}_{t+1},\boldsymbol{x}_0,\boldsymbol{y}\right)\|p_\theta^t\left(\boldsymbol{x}_t\mid\boldsymbol{x}_{t+1},\boldsymbol{y}\right)\right)\right]\\
&+\mathbb{E}\left[\log p_\theta^0\left(\boldsymbol{x}_0\mid\boldsymbol{x}_1,\boldsymbol{y}\right)\right]\\
&-\mathbb{E}\left[\mathrm{KL}\left(q^T\left(\boldsymbol{x}_T\mid\boldsymbol{x}_0,\boldsymbol{y}\right)\|p_\theta^T\left(\boldsymbol{x}_T\mid\boldsymbol{y}\right)\right)\right]
\end{aligned} \tag{3}$$

where $q\left(\boldsymbol{x}_0\right)$ denotes the data distribution, $q\left(\boldsymbol{y}\mid\boldsymbol{x}_0\right)$ in the main paper, the expectation on the right-hand side is

Table 1. **The guidance scales and the number of optimization per time step on the various tasks.** Note that these parameters may not be optimal due to the infinite number of possible combinations.

Tasks	Dataset	Guidance scale	The number of optimization per time step
4× Super-resolution	ImageNet [15]	2E+03	6
Deblurring	ImageNet [15]	6E+03	6
25% Inpainting	ImageNet [15]	4E+03	6
Colorization	ImageNet [15]	6E+03	6
Low-light enhancement	LOL dataset [23]	1E+05	6
HDR recovery	NTIRE dataset [16]	1E+05	1

Table 2. The performance of ablation studies on the way of guidance. We compare four ways of guidance in terms of FID.

FID	4x super-resolution	Deblur	25% Inpainting	Colorization
GDP-$\boldsymbol{x}_t$-v1	108.06	88.52	113.47	102.37
GDP-$\boldsymbol{x}_0$-v1	44.16	10.35	37.32	41.53
GDP-$\boldsymbol{x}_t$	64.67	5.00	20.24	66.43
GDP-$\boldsymbol{x}_0$	38.24	2.44	16.58	37.60

```
Algorithm 5: GDP-$\boldsymbol{x}_t$-v1 with fixed degradation model: Conditioner guided diffusion sampling on $\boldsymbol{x}_t$, given a diffusion model $(\mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t))$, corrupted image conditioner $\boldsymbol{y}$.
Input: Corrupted image $\boldsymbol{y}$, gradient scale $s$,
        degradation model $\mathcal{D}$, distance measure $\mathcal{L}$,
        optional quality enhancement loss $\mathcal{Q}$, quality
        enhancement scale $\lambda$.
Output: Output image $\boldsymbol{x}_0$ conditioned on $\boldsymbol{y}$
Sample $\boldsymbol{x}_T$ from $\mathcal{N}(0, \mathbf{I})$
for t from T to 1 do
    $\mu, \Sigma = \mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t)$
    $\tilde{\boldsymbol{x}}_0 = \frac{\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\boldsymbol{x}_t, t)}{\sqrt{\bar{\alpha}_t}}$
    $\mathcal{L}_{\boldsymbol{x}_t}^{total} = \mathcal{L}(\boldsymbol{y}, \mathcal{D}(\boldsymbol{x}_t)) + \mathcal{Q}(\boldsymbol{x}_t)$
    $\boldsymbol{x}_t \leftarrow \boldsymbol{x}_t - s\nabla_{\boldsymbol{x}_t}\mathcal{L}(\boldsymbol{y}, \mathcal{D}(\boldsymbol{x}_t))$
    Sample $\boldsymbol{x}_{t-1}$ by $q(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, \tilde{\boldsymbol{x}}_0) =$
     $\mathcal{N}\left(\boldsymbol{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t, \tilde{\boldsymbol{x}}_0), \tilde{\beta}_t\mathbf{I}\right)$,
    where
     $\tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t, \tilde{\boldsymbol{x}}_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\tilde{\boldsymbol{x}}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\boldsymbol{x}_t$
     and $\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$
end
return $\boldsymbol{x}_0$
```

given by sampling $\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0), \boldsymbol{y} \sim q(\boldsymbol{y} \mid \boldsymbol{x}_0), \boldsymbol{x}_T \sim q^T(\boldsymbol{x}_T \mid \boldsymbol{x}_0, \boldsymbol{y})$, and $\boldsymbol{x}_t \sim q^t(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}, \boldsymbol{x}_0, \boldsymbol{y})$ for $t \in [1, T-1]$.

```
Algorithm 6: GDP-$\boldsymbol{x}_0$-v1: Conditioner guided diffusion sampling on $\tilde{\boldsymbol{x}}_0$, given a diffusion model $(\mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t))$, corrupted image conditioner $\boldsymbol{y}$.
Input: Corrupted image $\boldsymbol{y}$, gradient scale $s$,
        degradation model $\mathcal{D}$, distance measure $\mathcal{L}$.
Output: Output image $\boldsymbol{x}_0$ conditioned on $\boldsymbol{y}$
Sample $\boldsymbol{x}_T$ from $\mathcal{N}(0, \mathbf{I})$
for t from T to 1 do
    $\mu, \Sigma = \mu_\theta(\boldsymbol{x}_t), \Sigma_\theta(\boldsymbol{x}_t)$
    $\tilde{\boldsymbol{x}}_0 = \frac{\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} - \frac{\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\boldsymbol{x}_t, t)}{\sqrt{\bar{\alpha}_t}}$
    $\mathcal{L}_{\tilde{\boldsymbol{x}}_0}^{total} = \mathcal{L}(\boldsymbol{y}, \mathcal{D}(\tilde{\boldsymbol{x}}_0)) + \mathcal{Q}(\tilde{\boldsymbol{x}}_0)$
    $\tilde{\boldsymbol{x}}_0 \leftarrow \tilde{\boldsymbol{x}}_0 - s\nabla_{\tilde{\boldsymbol{x}}_0}\mathcal{L}_{\tilde{\boldsymbol{x}}_0}^{total}$
    Sample $\boldsymbol{x}_{t-1}$ by $q(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, \tilde{\boldsymbol{x}}_0) =$
     $\mathcal{N}\left(\boldsymbol{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t, \tilde{\boldsymbol{x}}_0), \tilde{\beta}_t\mathbf{I}\right)$,
    where
     $\tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t, \tilde{\boldsymbol{x}}_0) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\tilde{\boldsymbol{x}}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\boldsymbol{x}_t$
     and $\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$
end
return $\boldsymbol{x}_0$
```

G. Sampling with DDIM

To accelerate the sampling strategy, GDP follows [12] to use DDIM, which skipping steps in the reverse process to speed up the DDPM generating process. We apply this method to the ImageNet dataset on the four tasks. We set the T=20 in the sampling process, while DDRM also utilizes the same time steps for a fair comparison.

As shown in Table 3, our GDP-x_0-DDIM(20) outperforms DDRM(20) on consistency and FID across four tasks. Although DDRM(20) obtains better PSNR and SSIM than our GDP-x_0-DDIM(20), the qualitative results of DDRM(20) are still worse than our GDP-x_0-DDIM(20), which can be seen from Figs. 7 and 8. Previous work [1–3, 18] demonstrated that these conventional automated evaluation measures (PSNR and SSIM) do not correlate well with human perception when the input resolution is low, and the magnification is large. This is not surprising since these metrics tend to penalize any synthesized high-frequency detail that is not perfectly aligned with the target image.

H. Image Guidance

A conditioner $p(\boldsymbol{y} \mid \boldsymbol{x})$ is exploited to improve a diffusion generator. Specifically, we can utilize a conditioner $p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t, t\right)$ on input images, and then use gradients $\nabla_{\boldsymbol{x}_t} \log p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t, t\right)$ to guide the diffusion sampling process towards a given the degraded images $\boldsymbol{y}$.

In this section, we will describe how to use such conditioners to improve the quality of sampled images. The notation is chosen as $p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t, t\right) = p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)$ and $\epsilon_\theta\left(\boldsymbol{x}_t, t\right) = \epsilon_\theta\left(\boldsymbol{x}_t\right)$ for brevity. Note that they refer to separate functions for each time step t.

H.1. Conditional Reverse Process

Assume a diffusion model with an unconditional reverse noising process $p_\theta\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right)$. In image restoration and enhancement, the corrupted inputs can be regarded as conditions. Therefore, we regard $\boldsymbol{y}$ as the input images, and $\boldsymbol{x}_t$ as the generated images in time step t. Then, the conditioner is formulated as follows:

$$p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right) = \frac{1}{K} exp\left(-\mathcal{L}\left(\boldsymbol{y}, \mathcal{D}\left(\boldsymbol{x}_t\right)\right)\right), \tag{4}$$

where $\mathcal{D}$ represents the degradation function, $\mathcal{L}$ stands for Mean Square Error together with optional Quality Enhancement Loss, and K is an arbitrary constant. In order to condition this on the input corrupted image $\boldsymbol{y}$, it is sufficient to sample each transition based on the following:

$$p_{\theta,\phi}\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}, \boldsymbol{y}\right) = C p_\theta\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right) p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right) \tag{5}$$

where C denotes a normalizing constant. It is typically intractable to sample from this distribution exactly, but Sohl-Dickstein *et al.* [19] show that it can be approximated as a perturbed Gaussian distribution. Sampling accurately from this distribution is often tricky, but Sohl-Dickstein *et al.* [19] prove that it could be approximated as a perturbed Gaussian distribution. It is formulated that the diffusion model samples the previous time step $\boldsymbol{x}_t$ from time step $\boldsymbol{x}_{t+1}$ via a Gaussian distribution:

$$p_\theta\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right) = \mathcal{N}(\mu, \Sigma) \tag{6}$$

$$\log p_\theta\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right) = -\frac{1}{2}\left(\boldsymbol{x}_t - \mu\right)_T \Sigma^{-1}\left(\boldsymbol{x}_t - \mu\right) + Z \tag{7}$$

We can assume that $\log_\phi p\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)$ owns low curvature when compared with Σ^{-1}. This assumption is reasonable under the constraint that the infinite diffusion step, where $\|\Sigma\| \to 0$. Under the circumstances, $\log p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)$ can be approximated via a Taylor expansion around $\boldsymbol{x}_t = \mu$ as:

$$\begin{aligned}\log p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right) &\approx \left.\log p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)\right|_{\boldsymbol{x}_t=\mu} \\ &\quad + \left(\boldsymbol{x}_t - \mu\right) \nabla_{\boldsymbol{x}_t} \left.\log p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)\right|_{\boldsymbol{x}_t=\mu} \\ &= \left(\boldsymbol{x}_t - \mu\right) g + Z_1\end{aligned} \tag{8}$$

Here, $g = \nabla_{\boldsymbol{x}_t} \log p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right) \|_{\boldsymbol{x}_t=\mu}$, and Z_1 is a constant. We can replace the g with Eq. 4 as follows:

$$\log p\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right) = -\mathcal{L}\left(\boldsymbol{y}, \mathcal{D}\left(\boldsymbol{x}_t\right)\right) - \log K \tag{9}$$

$$g = \nabla_{\boldsymbol{x}_t} \log p\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right) = -\nabla_{\boldsymbol{x}_t} \mathcal{L}\left(\boldsymbol{y}, \mathcal{D}\left(\boldsymbol{x}_t\right)\right) \tag{10}$$

This gives:

$$\begin{aligned}&\log\left(p_\theta\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right) p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)\right) \\ &\approx -\frac{1}{2}\left(\boldsymbol{x}_t - \mu\right)^T \Sigma^{-1}\left(\boldsymbol{x}_t - \mu\right) + \left(\boldsymbol{x}_t - \mu\right) g + Z_2 \\ &= -\frac{1}{2}\left(\boldsymbol{x}_t - \mu - \Sigma g\right)^T \Sigma^{-1}\left(\boldsymbol{x}_t - \mu - \Sigma g\right) + \frac{1}{2} g^T \Sigma g + Z_2 \\ &= -\frac{1}{2}\left(\boldsymbol{x}_t - \mu - \Sigma g\right)^T \Sigma^{-1}\left(\boldsymbol{x}_t - \mu - \Sigma g\right) + Z_3 \\ &= \log p(z) + Z_4, z \sim \mathcal{N}(\mu + \Sigma g, \Sigma)\end{aligned} \tag{11}$$

where the constant term C_4 could be safely ignored because it is equivalent to the normalizing coefficient Z in Eq. 5. Thus, we find that the conditional transition operator can be approximated by a Gaussian similar to the unconditional transition operator, but with a mean shifted by Σg. Moreover, an optional scaling factor s is included for gradients, which will be described in more detail in Sec. H.3. However, it is experimentally found that this guidance way might not be effective enough, where our GDP-x_0 is systematically studied.

H.2. Conditional Diffusion Process

Here, we figure out that conditional sampling can be fulfilled with a transition operator proportional to $p_\theta\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right) p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)$, where $p_\theta\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right)$ approximates $q\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right)$ and $p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)$ approximates the distribution of the input for a noised sample $\boldsymbol{x}_t$.

A conditional Markovian noising process $\hat{q}$ is similar to q. And $\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_0\right)$ is assumed as a known and readily avail-

Table 3. **The performances of DDRM (20) and GDM-x_0-DDIM(20) towards the four tasks on ImageNet 1k.** The DDIM sample steps are all set to 20 to make a fair comparison.

Task	4× super resolution				Deblur				25% Impainting				Colorization			
	PSNR	SSIM	Consistency	FID	PSNR	SSIM	Consistency	FID	PSNR	SSIM	Consistency	FID	PSNR	SSIM	Consistency	FID
DDRM(20) [6]	**26.53**	**0.784**	19.39	40.75	**35.64**	**0.978**	50.24	4.78	**34.28**	**0.958**	**4.08**	24.09	**22.12**	**0.924**	38.66	47.05
GDP-x_0-DDIM(20)	23.77	0.623	**9.24**	**39.46**	24.87	0.683	**44.39**	**3.66**	30.82	0.892	7.10	**19.70**	21.13	0.840	**37.33**	**41.38**

Table 4. The time comparison of GDP-x_0-DDIM(20) and GDP-x_0 on 4x super-resolution. These experiments are compared on Tesla A100.

		Guidance scale	Total steps	Guidance times per steps	Generation time per image
GDP-x_0	w.o. DDIM	2e3	1000	6	69.55
GDP-x_0-DDIM(20)	w. DDIM	22e5	20	20	1.74

able degraded images distribution for each sample.

$$\begin{aligned} \hat{q}\left(\boldsymbol{x}_0\right) &:= q\left(\boldsymbol{x}_0\right) && (12) \\ \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_0\right) &:= \text{Corrupted input image per sample} && (13) \\ \hat{q}\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t, \boldsymbol{y}\right) &:= q\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t\right) && (14) \\ \hat{q}\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{x}_0, \boldsymbol{y}\right) &:= \prod_{t=1}^{T} \hat{q}\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}, \boldsymbol{y}\right) && (15) \end{aligned}$$

Assuming that the noising process $\hat{q}$ is conditioned on $\boldsymbol{y}$, we can reveal that $\hat{q}$ behaves exactly like q when not conditioned on $\boldsymbol{y}$. According to this idea, we first derive the unconditional noising operator $\hat{q}\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t\right)$:

$$\begin{aligned} \hat{q}\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t\right) &= \int_{\boldsymbol{y}} \hat{q}\left(\boldsymbol{x}_{t+1}, \boldsymbol{y} \mid \boldsymbol{x}_t\right) d\boldsymbol{y} && (16) \\ &= \int_{\boldsymbol{y}} \hat{q}\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t, \boldsymbol{y}\right) \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right) d\boldsymbol{y} && (17) \\ &= \int_{\boldsymbol{y}} q\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t\right) \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right) d\boldsymbol{y} && (18) \\ &= q\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t\right) \int_{\boldsymbol{y}} \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right) d\boldsymbol{y} && (19) \\ &= q\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t\right) && (20) \\ &= \hat{q}\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t, \boldsymbol{y}\right) && (21) \end{aligned}$$

Similarly, the joint distribution $\hat{q}\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{x}_0\right)$ can be written as:

$$\begin{aligned} \hat{q}\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{x}_0\right) &= \int_{\boldsymbol{y}} \hat{q}\left(\boldsymbol{x}_{1:T}, \boldsymbol{y} \mid \boldsymbol{x}_0\right) d\boldsymbol{y} && (22) \\ &= \int_{\boldsymbol{y}} \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_0\right) \hat{q}\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{x}_0, \boldsymbol{y}\right) d\boldsymbol{y} && (23) \\ &= \int_{\boldsymbol{y}} \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_0\right) \prod_{t=1}^{T} \hat{q}\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}, \boldsymbol{y}\right) d\boldsymbol{y} && (24) \\ &= \int_{\boldsymbol{y}} \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_0\right) \prod_{t=1}^{T} q\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}\right) d\boldsymbol{y} && (25) \\ &= \prod_{t=1}^{T} q\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}\right) \int_{\boldsymbol{y}} \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_0\right) d\boldsymbol{y} && (26) \\ &= \prod_{t=1}^{T} q\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}\right) && (27) \\ &= q\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{x}_0\right) && (28) \end{aligned}$$

$\hat{q}\left(\boldsymbol{x}_t\right)$ can be derived by using Eq. 28 as follows:

$$\begin{aligned} \hat{q}\left(\boldsymbol{x}_t\right) &= \int_{\boldsymbol{x}_{0:t-1}} \hat{q}\left(\boldsymbol{x}_0, \ldots, \boldsymbol{x}_t\right) d\boldsymbol{x}_{0:t-1} && (29) \\ &= \int_{\boldsymbol{x}_{0:t-1}} \hat{q}\left(\boldsymbol{x}_0\right) \hat{q}\left(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t \mid \boldsymbol{x}_0\right) d\boldsymbol{x}_{0:t-1} && (30) \\ &= \int_{\boldsymbol{x}_{0:t-1}} q\left(\boldsymbol{x}_0\right) q\left(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t \mid \boldsymbol{x}_0\right) d\boldsymbol{x}_{0:t-1} && (31) \\ &= \int_{\boldsymbol{x}_{0:t-1}} q\left(\boldsymbol{x}_0, \ldots, \boldsymbol{x}_t\right) d\boldsymbol{x}_{0:t-1} && (32) \\ &= q\left(\boldsymbol{x}_t\right) && (33) \end{aligned}$$

It is proved by Bayes rule that the unconditional reverse process $\hat{q}\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right) = q\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right)$ when using the identities $\hat{q}\left(\boldsymbol{x}_t\right) = q\left(\boldsymbol{x}_t\right)$ and $\hat{q}\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t\right) = q\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t\right)$.

Note that $\hat{q}$ is able to produce an input function $\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)$. It is shown that this distribution of the input does

Table 5. The quantitative comparison of performance on CelebA.

CelebA	4x SR				Deblur				25% Inpainting			
	PSNR	SSIM	Consistency	FID	PSNR	SSIM	Consistency	FID	PSNR	SSIM	Consistency	FID
DDRM	29.50	0.863	6.82	87.71	**36.51**	**0.98**	35.91	14.30	31.99	0.918	**0.47**	69.46
GDP-x_t	29.19	0.847	14.11	94.98	27.35	0.81	34.87	9.97	36.19	0.963	1.94	22.53
GDP-x_0	**30.26**	**0.868**	**5.33**	**46.64**	28.66	0.83	**32.66**	**4.50**	**37.70**	**0.972**	0.51	**11.62**

Table 6. The quantitative comparison of results on LSUN bedroom.

LSUN Bedroom	4x SR		Deblur		25% Inpainting		Colorization	
	Consistency	FID	Consistency	FID	Consistency	FID	Consistency	FID
DDRM	20.33	40.12	43.78	10.16	**5.33**	22.49	35.16	45.22
GDP-x_t	70.46	58.62	46.90	12.50	9.33	20.63	66.88	57.13
GDP-x_0	**7.66**	**36.94**	**42.28**	**9.51**	6.77	**18.34**	**33.51**	**34.59**

not depend on $\boldsymbol{x}_{t+1}$ (the noisy version of $\boldsymbol{x}_t$), we will discuss this fact later by exploiting:

$$\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t, \boldsymbol{x}_{t+1}\right) = \hat{q}\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t, \boldsymbol{y}\right) \frac{\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)}{\hat{q}\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t\right)} \tag{34}$$

$$= \hat{q}\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t\right) \frac{\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)}{\hat{q}\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t\right)} \tag{35}$$

$$= \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right) \tag{36}$$

In this way, the conditional reverse process can be derived as:

$$\hat{q}\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}, \boldsymbol{y}\right) = \frac{\hat{q}\left(\boldsymbol{x}_t, \boldsymbol{x}_{t+1}, \boldsymbol{y}\right)}{\hat{q}\left(\boldsymbol{x}_{t+1}, \boldsymbol{y}\right)} \tag{37}$$

$$= \frac{\hat{q}\left(\boldsymbol{x}_t, \boldsymbol{x}_{t+1}, \boldsymbol{y}\right)}{\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_{t+1}\right) \hat{q}\left(\boldsymbol{x}_{t+1}\right)} \tag{38}$$

$$= \frac{\hat{q}\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right) \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t, \boldsymbol{x}_{t+1}\right) \hat{q}\left(\boldsymbol{x}_{t+1}\right)}{\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_{t+1}\right) \hat{q}\left(\boldsymbol{x}_{t+1}\right)} \tag{39}$$

$$= \frac{\hat{q}\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right) \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t, \boldsymbol{x}_{t+1}\right)}{\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_{t+1}\right)} \tag{40}$$

$$= \frac{\hat{q}\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right) \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)}{\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_{t+1}\right)} \tag{41}$$

$$= \frac{q\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right) \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)}{\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_{t+1}\right)} \tag{42}$$

where the $\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_{t+1}\right)$ can be treated as a constant because it does not depend on $\boldsymbol{x}_{t+1}$. Therefore, we want to sample from the distribution $Cq\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right) \hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)$ where C denotes the normalization constant. We already have a neural network approximation of $q\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right)$ called $p_\theta\left(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}\right)$, so the rest is $\hat{q}\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)$ that can be obtained by computing a conditioner $p_\phi\left(\boldsymbol{y} \mid \boldsymbol{x}_t\right)$ on noised images $\boldsymbol{x}_t$ derived by sampling from $q\left(\boldsymbol{x}_t\right)$.

H.3. Scaling Conditioner Gradients

The conditioner is incorporated into the sampling process of the diffusion model using Eq. 11. To unveil the effect of scaling conditioner gradients, note that $s \cdot \nabla_{\boldsymbol{x}} \log p(\boldsymbol{y} \mid \boldsymbol{x}) = \nabla_{\boldsymbol{x}} \log \frac{1}{K} p(\boldsymbol{y} \mid \boldsymbol{x})^s$, where K is an arbitrary constant. Thus, the conditioning process is still theoretically based on the re-normalized distribution of the input proportional to $p(\boldsymbol{y} \mid \boldsymbol{x})^s$. If $s > 1$, this distribution becomes sharper than $p(\boldsymbol{y} \mid \boldsymbol{x})$ because larger values are exponentially magnified. Therefore, using a larger gradient scale to focus more on the modes of the conditioner may be beneficial in producing higher fidelity (but less diverse) samples. In this paper, due to the observation that Σ might exert a negative influence on the quality of images. Therefore, with the absence of the Σ, the guidance scale can be a variable scale $\hat{s}$, where $s = \Sigma \hat{s}$. Thanks to this variable scale $\hat{s}$, the quality of images can be promoted

I. Additional Results on Linear inverse problems

We provide additional figures below showing GDP's versatility across different datasets and linear inverse problems (Figures 9, 10, 11, 12), and 14). We present more uncurated samples from the ImageNet experiments in Figures 13, 15, 16, 17, 18, and 19. Moreover, our GDP is also able to recover the corrupted images that undergo multilinear degradations, as shown in Fig. 20

J. Additional Results on Low-light Enhancement

In addition to the linear inverse problems, we further show more samples on the blind and non-linear task of low-light enhancement. As shown in 21, 24, and 26, our GDP

Table 7. The weight of reconstruction loss and quality enhancement loss.

	MSE loss	Exposure Control Loss	Color Constancy Loss	Illumination Smoothness Loss
Colorization	1	0	500	0
Low-light Enhancement	1	1/100	1/200	1
HDR recovery	1	1/100	1/200	1

performs well under the three datasets, including LOL, VE-LOL-L, and LoLi-phone, indicating the effectiveness of GDP under the different distributions of the images. Moreover, we also compare the GDP with other methods on the three datasets. As seen in 23, and 25, GDP-x_0 is able to generate more satisfactory images than other supervised learning, unsupervised learning, self-supervised, and zero-shot learning methods. Note that GDP-x_t tends to yield images lighter than the ones generated by GDP-x_0. Furthermore, GDP can adjust the brightness of generated images by the Exposure Control Loss. As shown in 22, users can change the gray level E in the RGB color space to obtain the target images with specific brightness.

K. Additional Results on HDR Recovery

As shown in 28, our HDR-GDP-x_0 is capable of adjusting the over-exposed and under-exposed areas of the picture in various scenes. It is noted that since the model used by GDP is pre-trained on ImageNet, the tone of the generated picture will be slightly different from ground truth images. Moreover, we also show more samples compared with the state-of-the-art methods, including AHDRNet [25], HDR-GAN [13], DeepHDR [24] and deep-high-dynamic-range [5]. As seen in Fig. 29, our HDR-GDP-x_0 can recover more realistic images with more details.

L. Additional Results on Ablation Study

The visualization comparisons of the ablation study on the trainable degradation and the patch-based tactic are shown in Figs. 30 and 31. It is shown that Model A fails to generate high-quality images due to the interpolation operation, while Model B generates images with more artifacts because of the naive restoration. Model C predicts the outputs in an uncontrollable way thanks to the randomly initiated and fixed parameters.

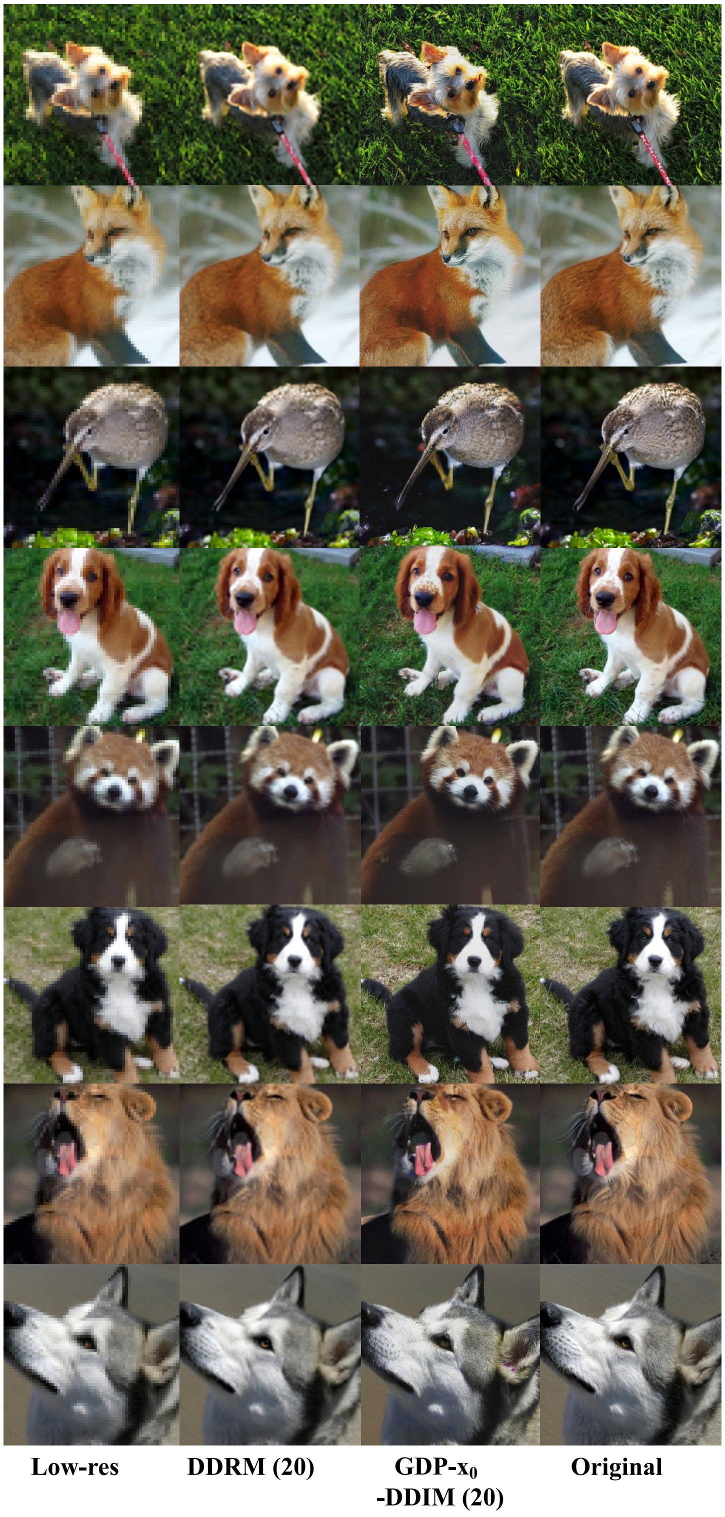


Figure 7. **More samples from the <u>4 × super-resolution</u> task of GDP-x_0-DDIM (20) compare with DDRM (20) on 256 × 256 <u>ImageNet 1K.</u>** The generated images by the DDRM (20) are still blurred, while our proposed GDP-x_0 with 20 steps of DDIM sampling can restore more details.

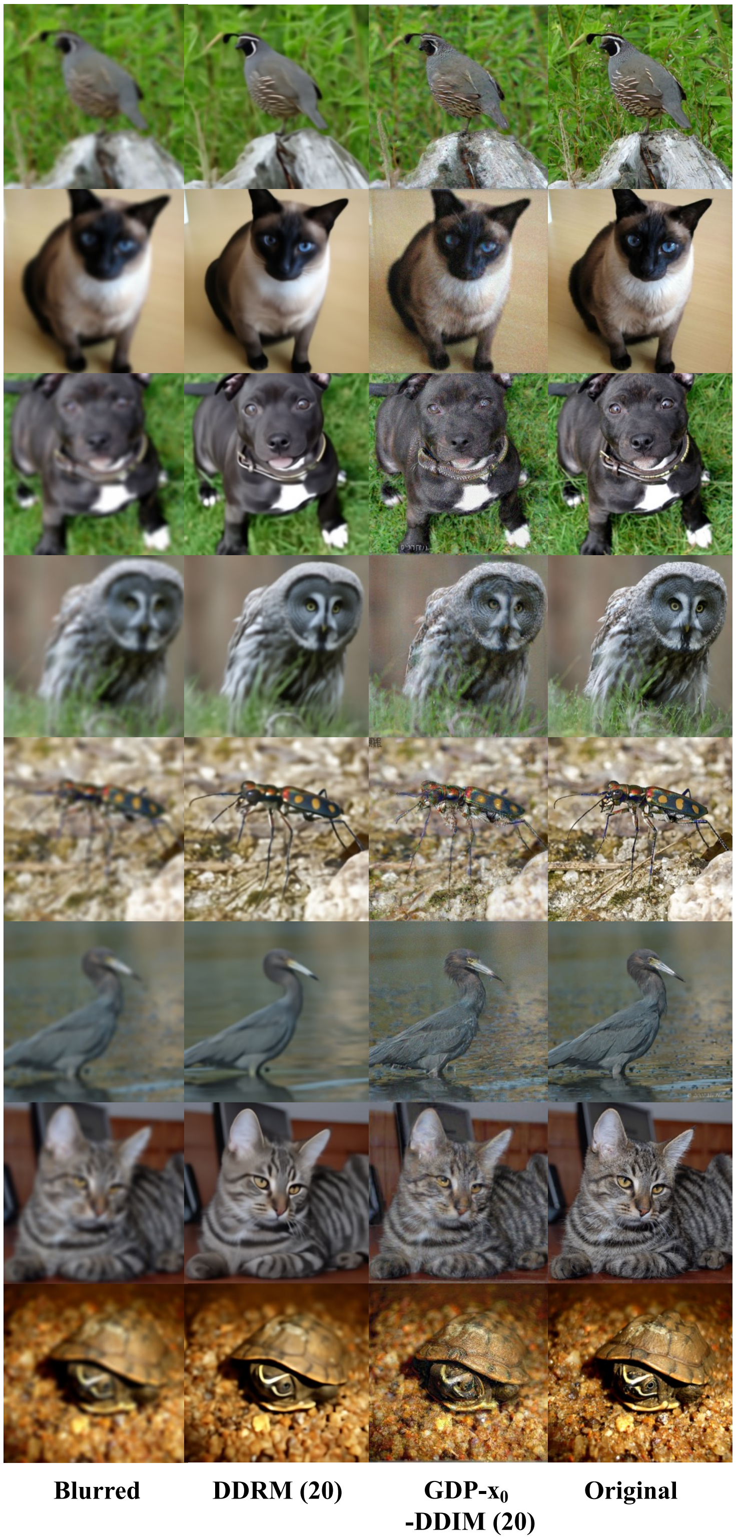


Figure 8. **More samples from the deblurring task of GDP-x_0-DDIM (20) compare with DDRM (20) on 256 × 256 ImageNet 1K.** Our GDP-x_0-DDIM (20) can recover more details than DDRM (20) under the same DDIM steps.

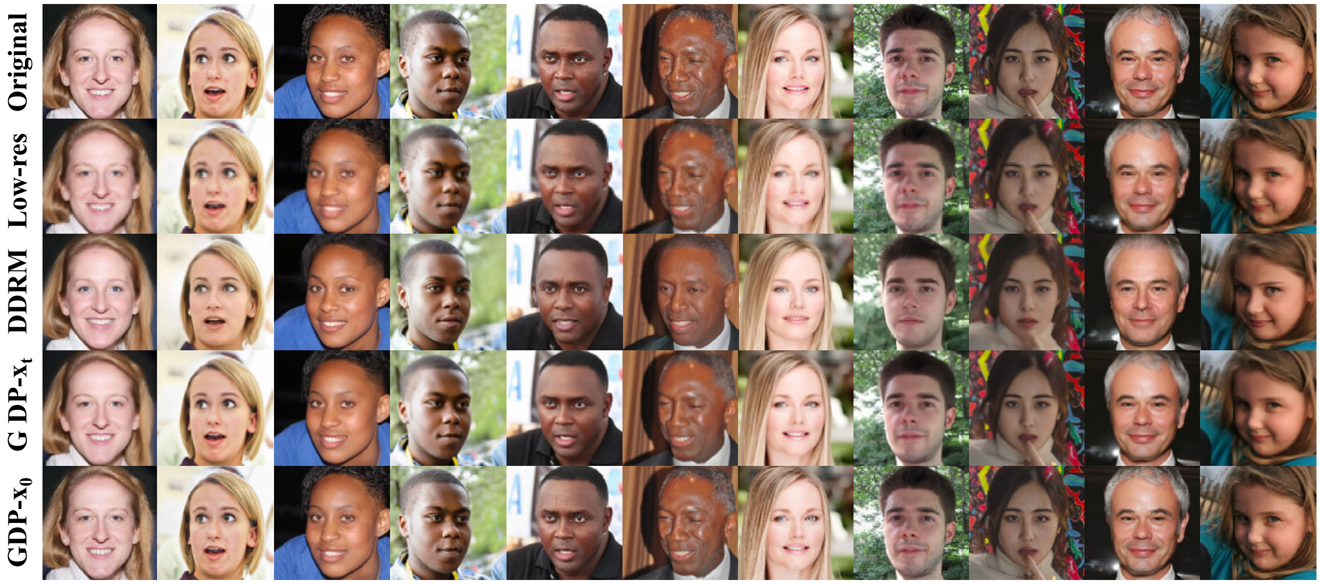


Figure 9. **$4\times$ super-resolution results of DDRM, GDP-$\boldsymbol{x}_t$, and GDP-$\boldsymbol{x}_0$ on CelebA face images.** Compared with GDP-x_t and DDRM, GDP-x_0 can restore more realistic faces, such as the wrinkles on the faces, systematically demonstrating the superiority of the guidance on x_0 protocol.

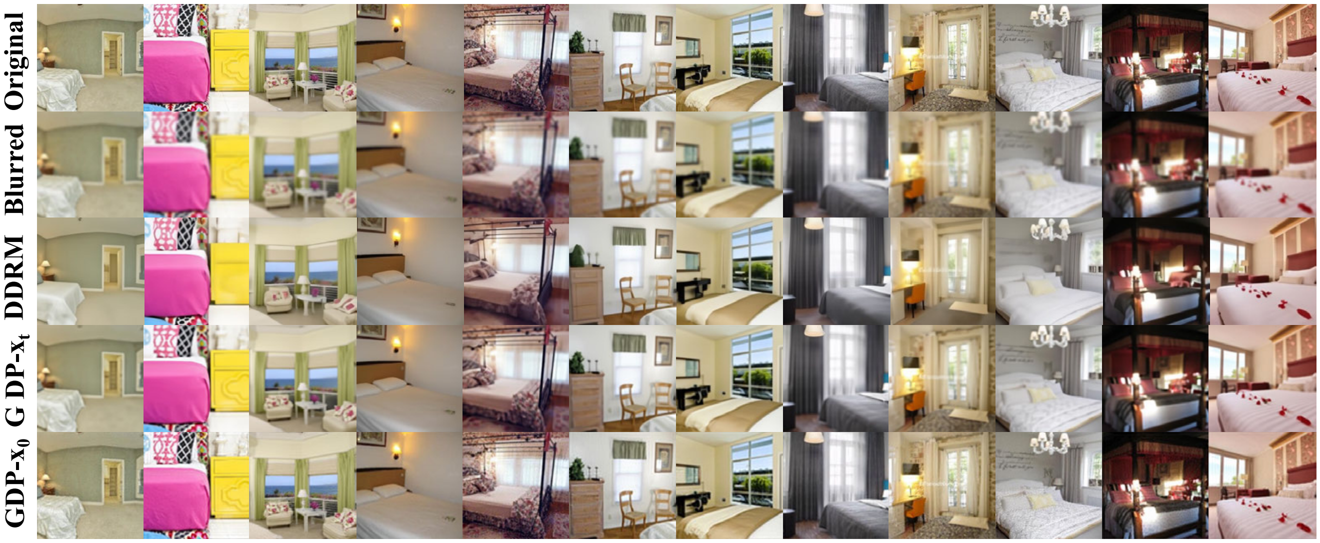


Figure 10. **Deblurring results of DDRM, GDP-$\boldsymbol{x}_t$, and GDP-$\boldsymbol{x}_0$ on LSUN bedroom images.**

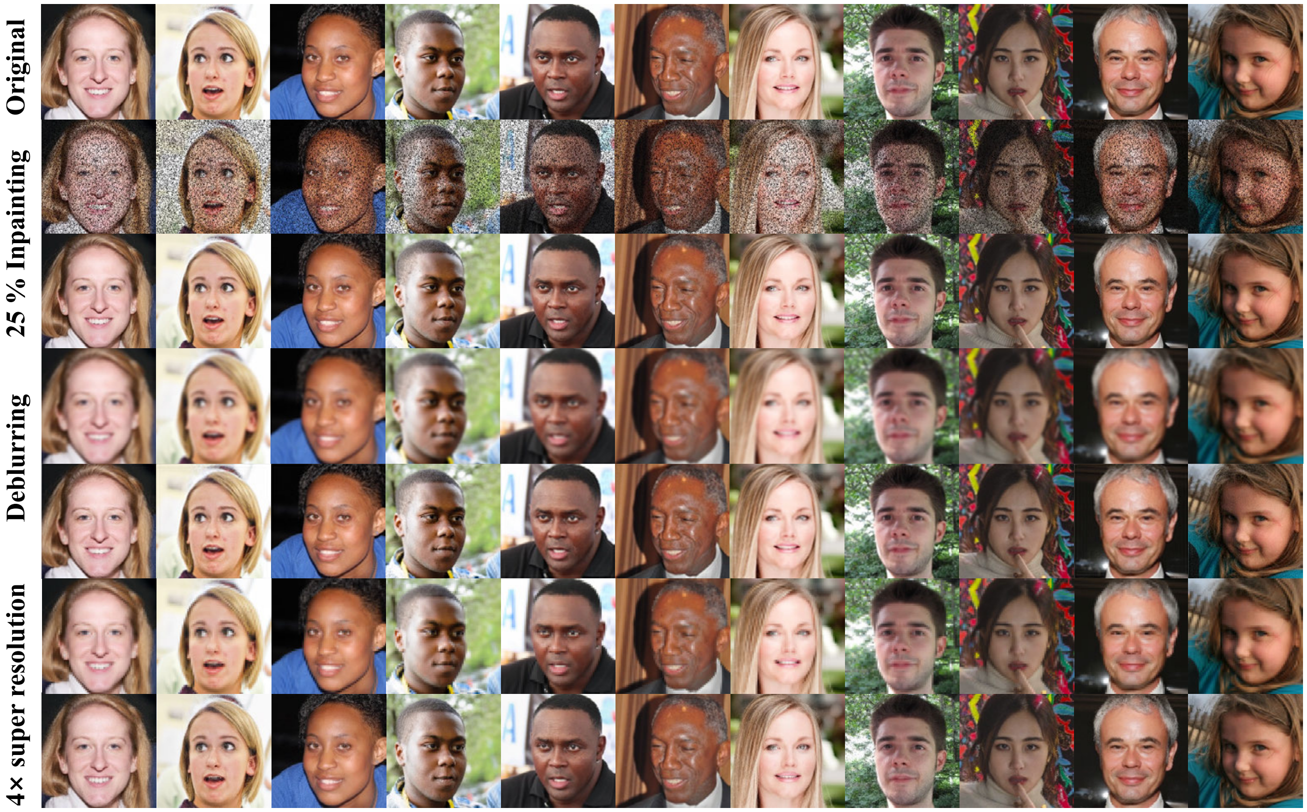


Figure 11. **Pairs of degraded and recovered 256 $\times$ 256 CelebA face images with a GDP-$\boldsymbol{x}_0$.** Three tasks including 25% inpainting, deblurring and 4 $\times$ super-resolution are vividly depicted.

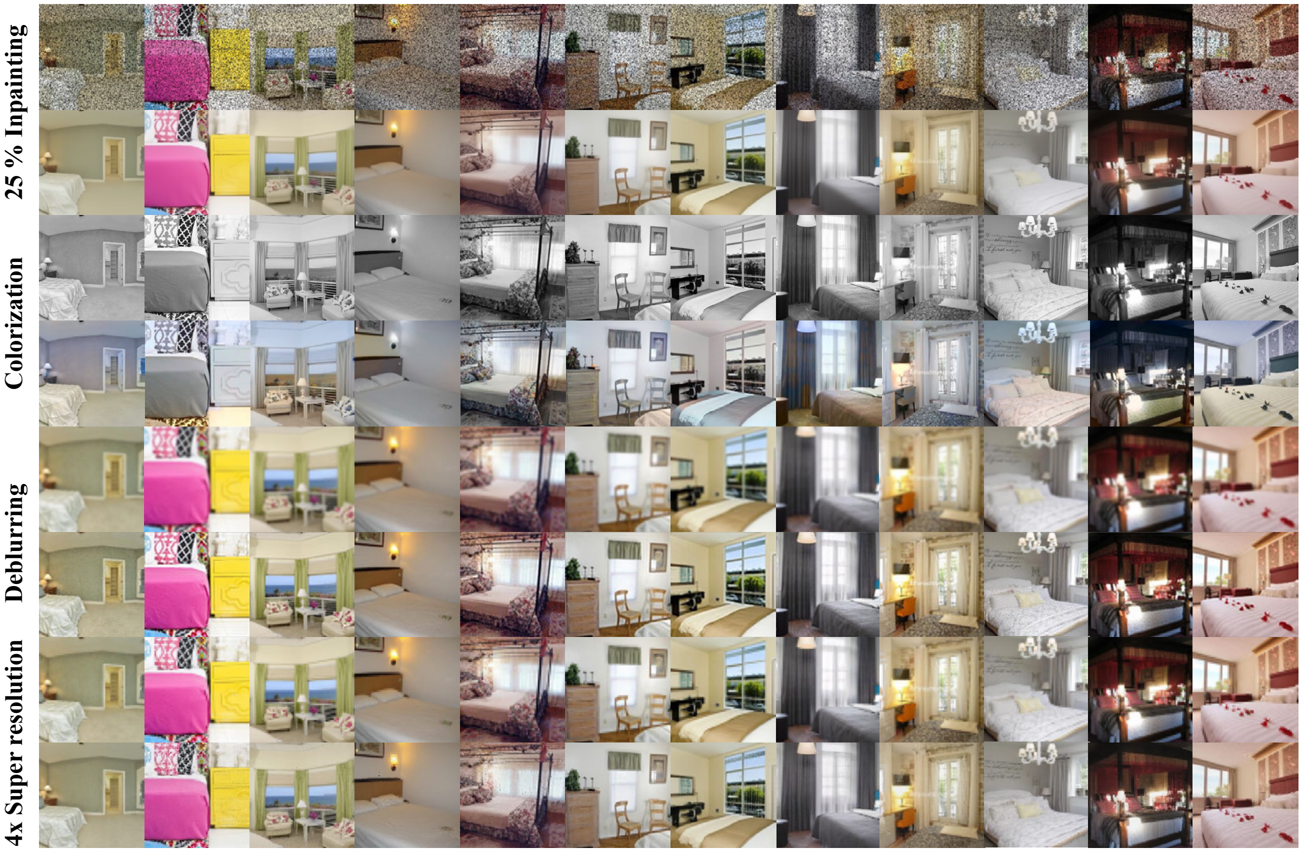


Figure 12. **Pairs of degraded and recovered 256 × 256 <u>LSUN bedroom</u> images with a GDP-$\boldsymbol{x}_0$.** We show more samples under the 25% inpainting, colorization, deblurring and 4 × super-resolution.

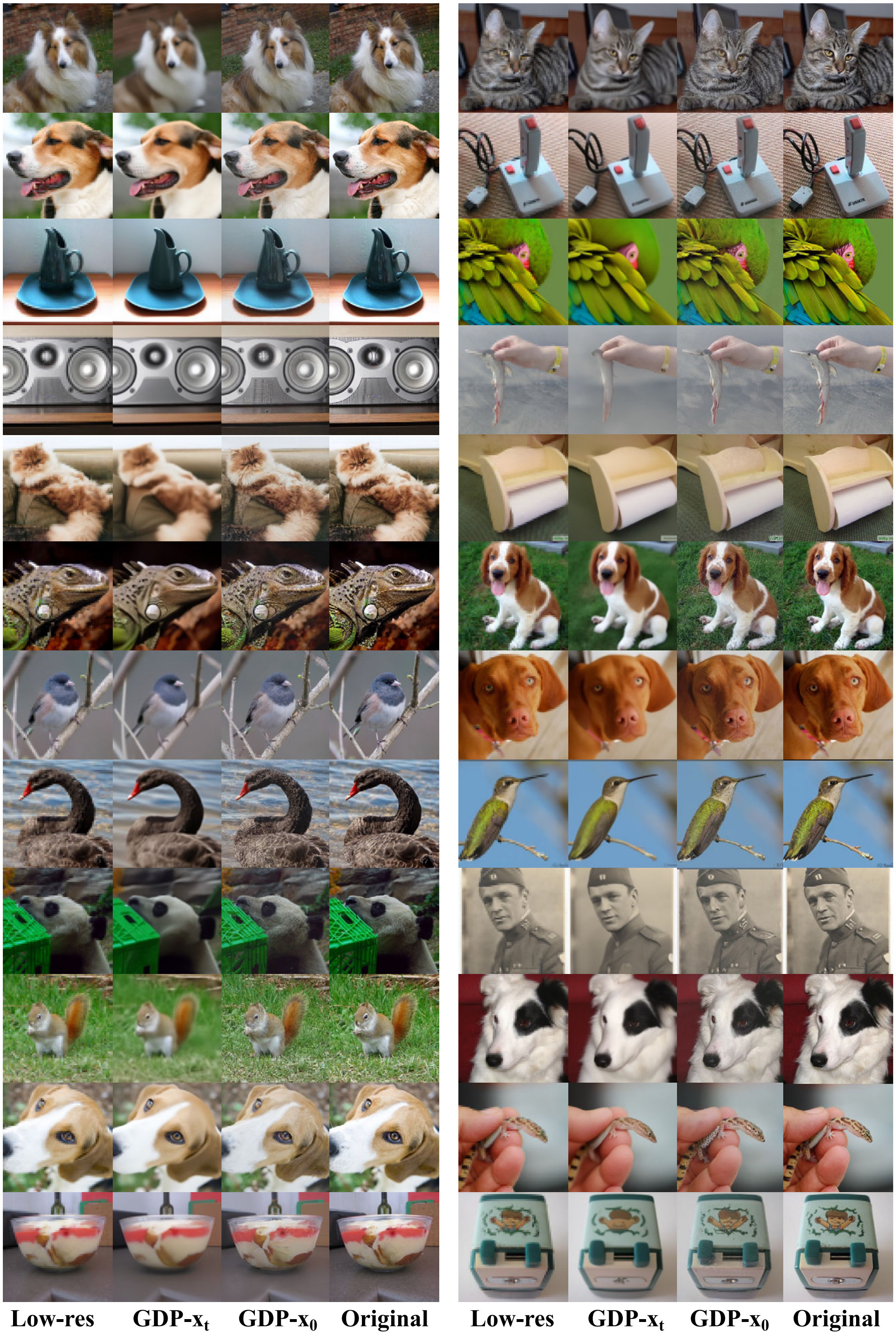


Figure 13. **Uncurated samples from the 4 × super-resolution task on 256 × 256 ImageNet 1K.**

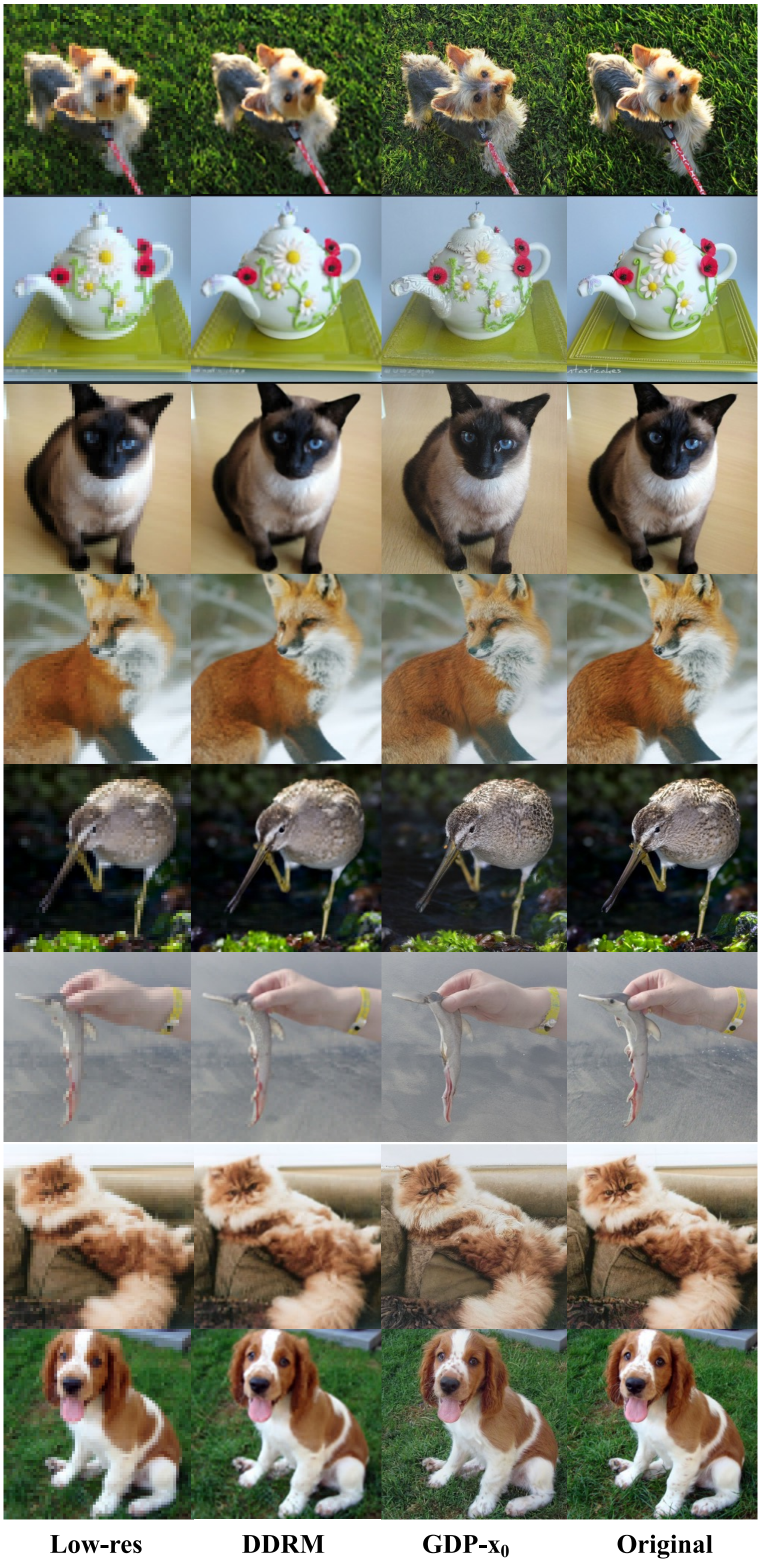


Figure 14. **More samples from the 4 × super-resolution task compare with DDRM on 256 × 256 ImageNet 1K.** As we mentioned above, DDRM adds guidance on the x_t, leading to the less satisfactory results than our GDP-x_0.

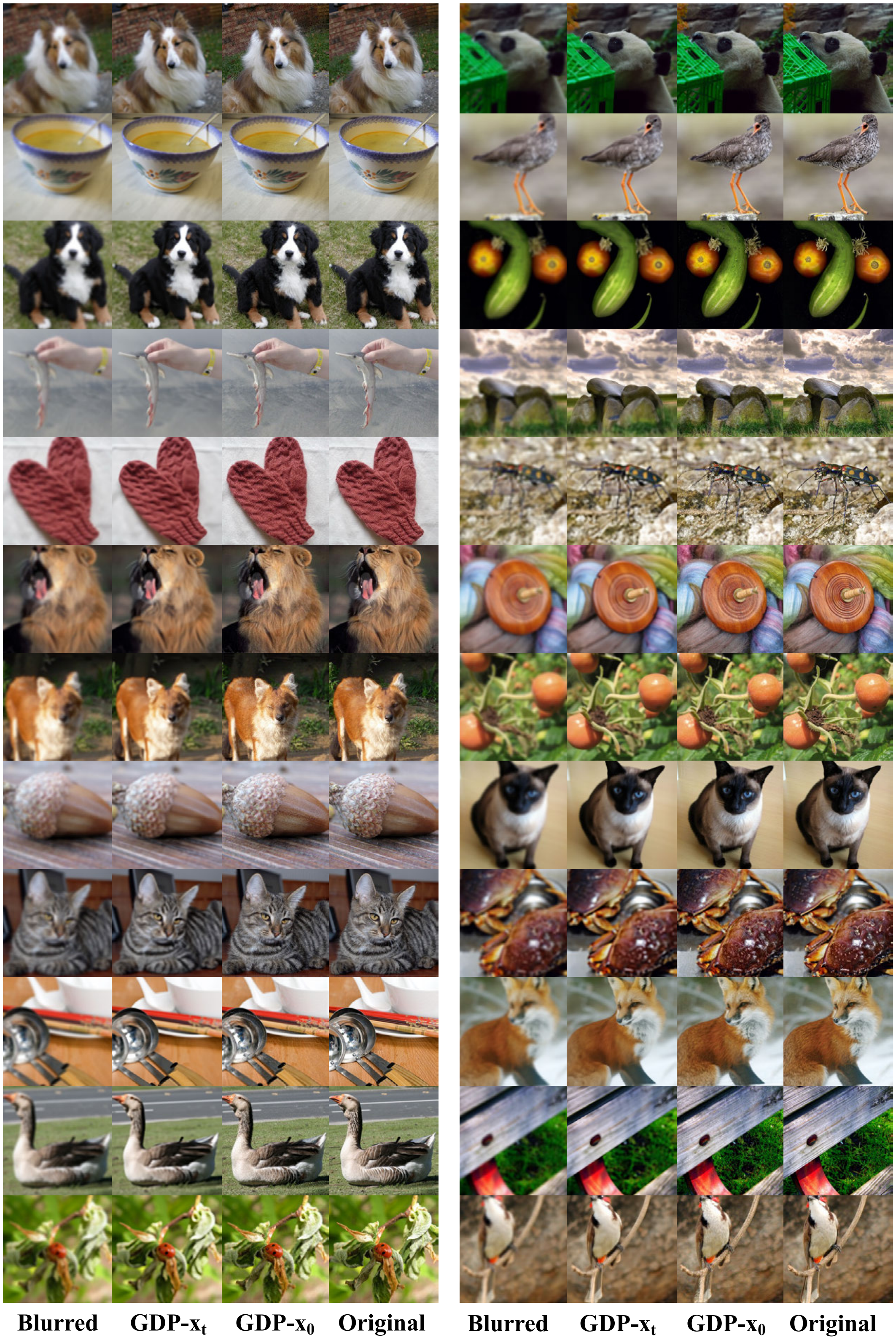


Figure 15. **Uncurated samples from the deblurring task on 256 × 256 ImageNet 1K.**

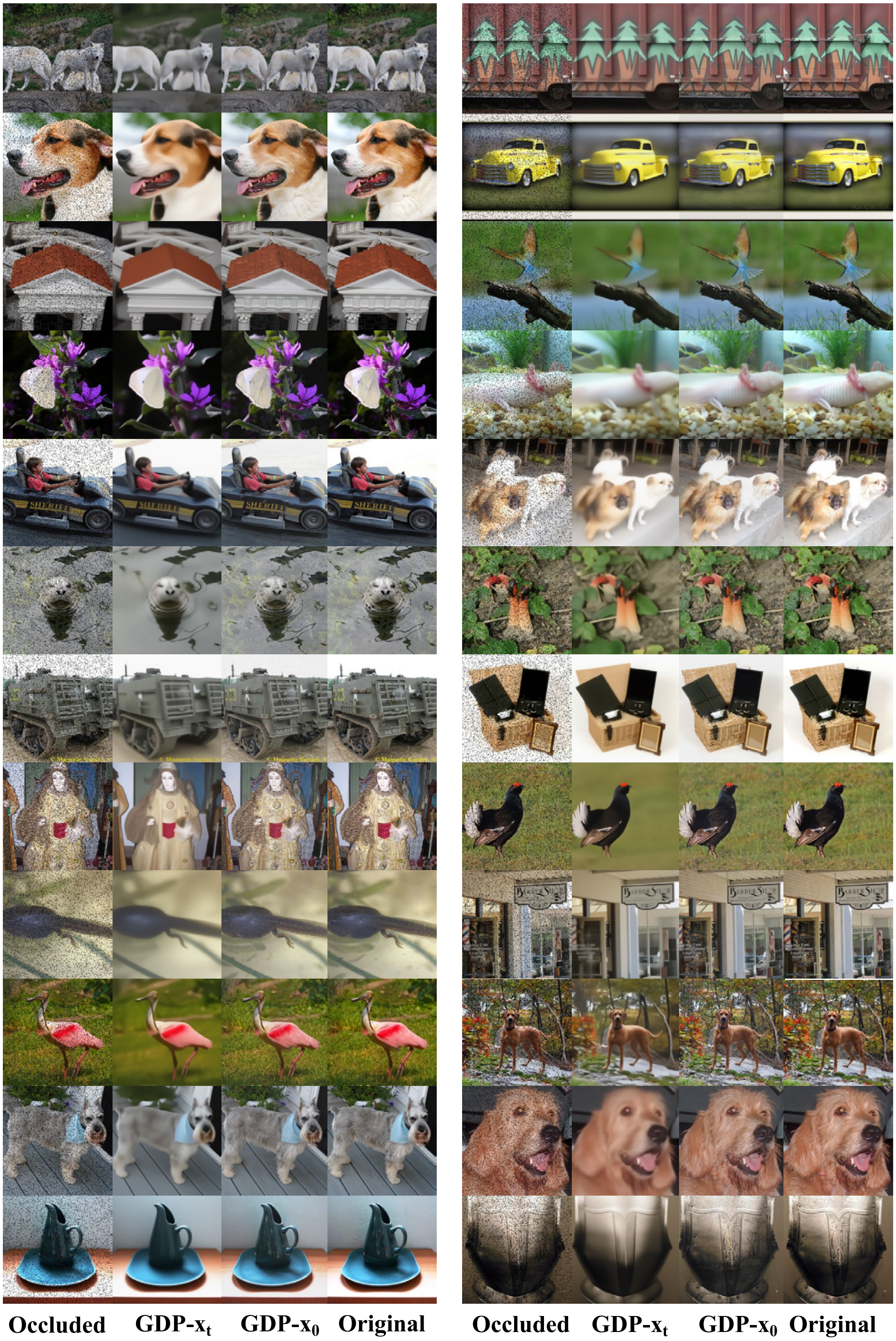


Figure 16. **Uncurated samples from the 10% inpainting task on 256 × 256 ImageNet 1K.**

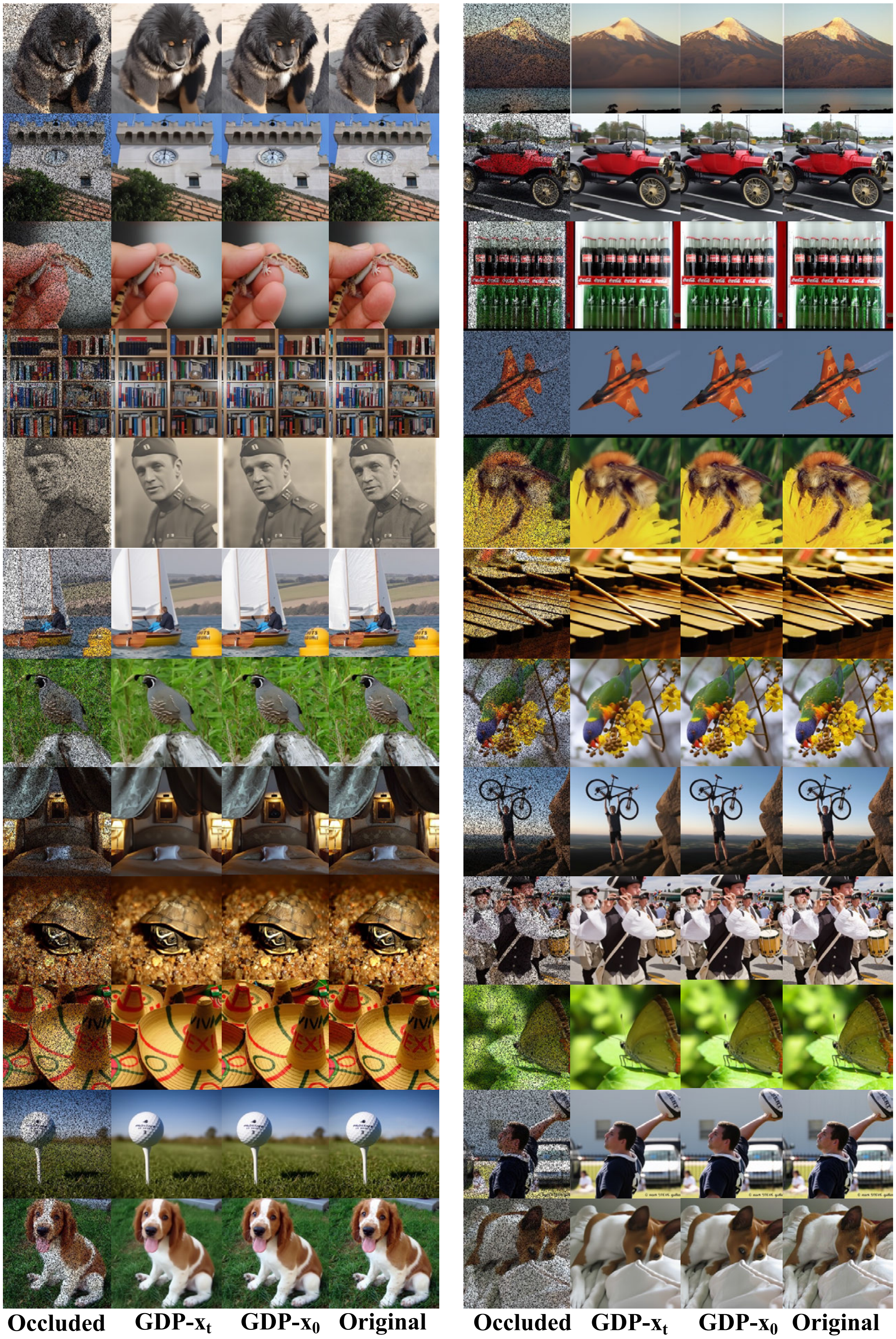


Figure 17. **Uncurated samples from the 25% inpainting task on 256 × 256 ImageNet 1K.**

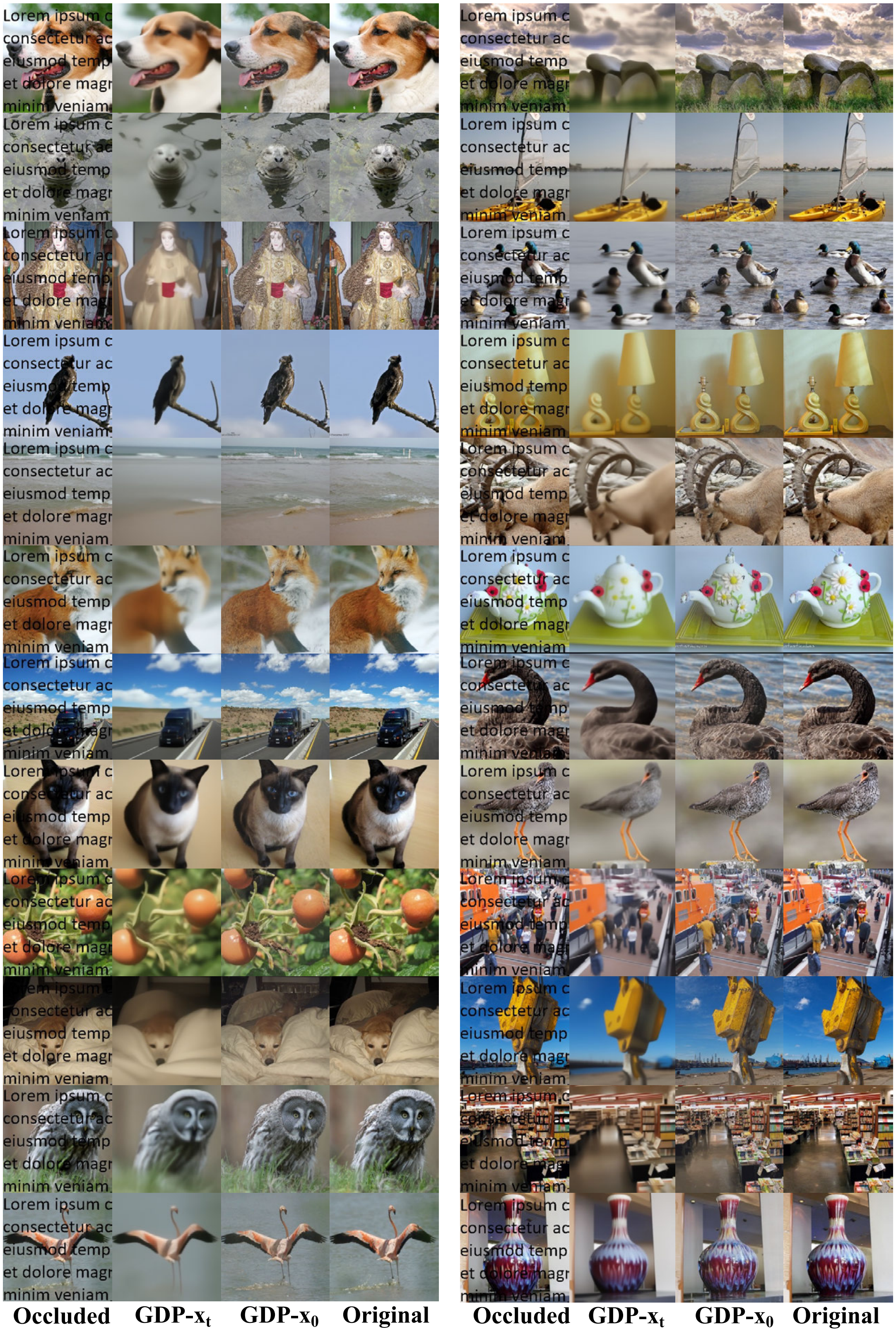


Figure 18. **Uncurated samples from the inpainting task on 256 × 256 ImageNet 1K.**

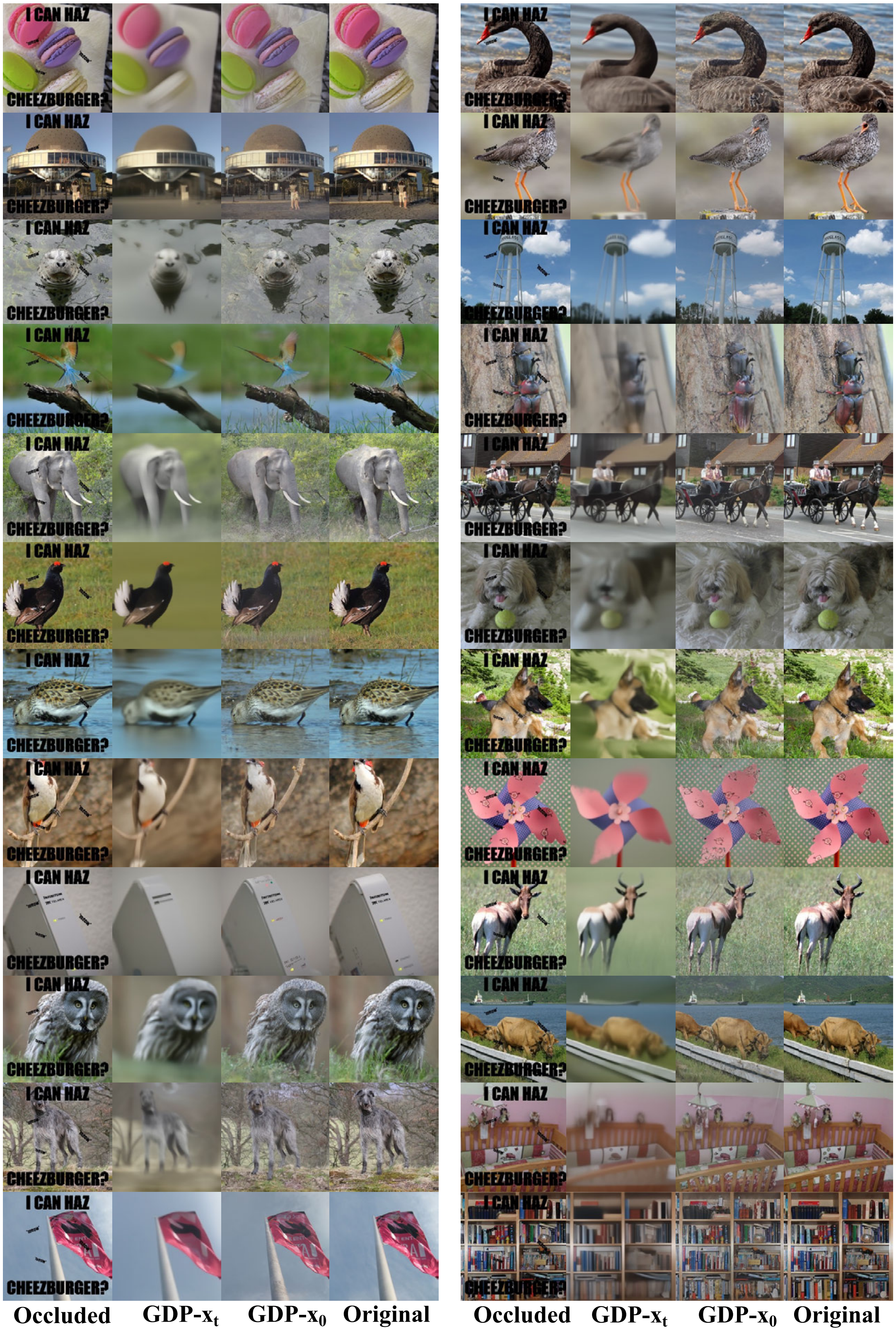


Figure 19. **Uncurated samples from the inpainting task on 256 × 256 ImageNet 1K.**

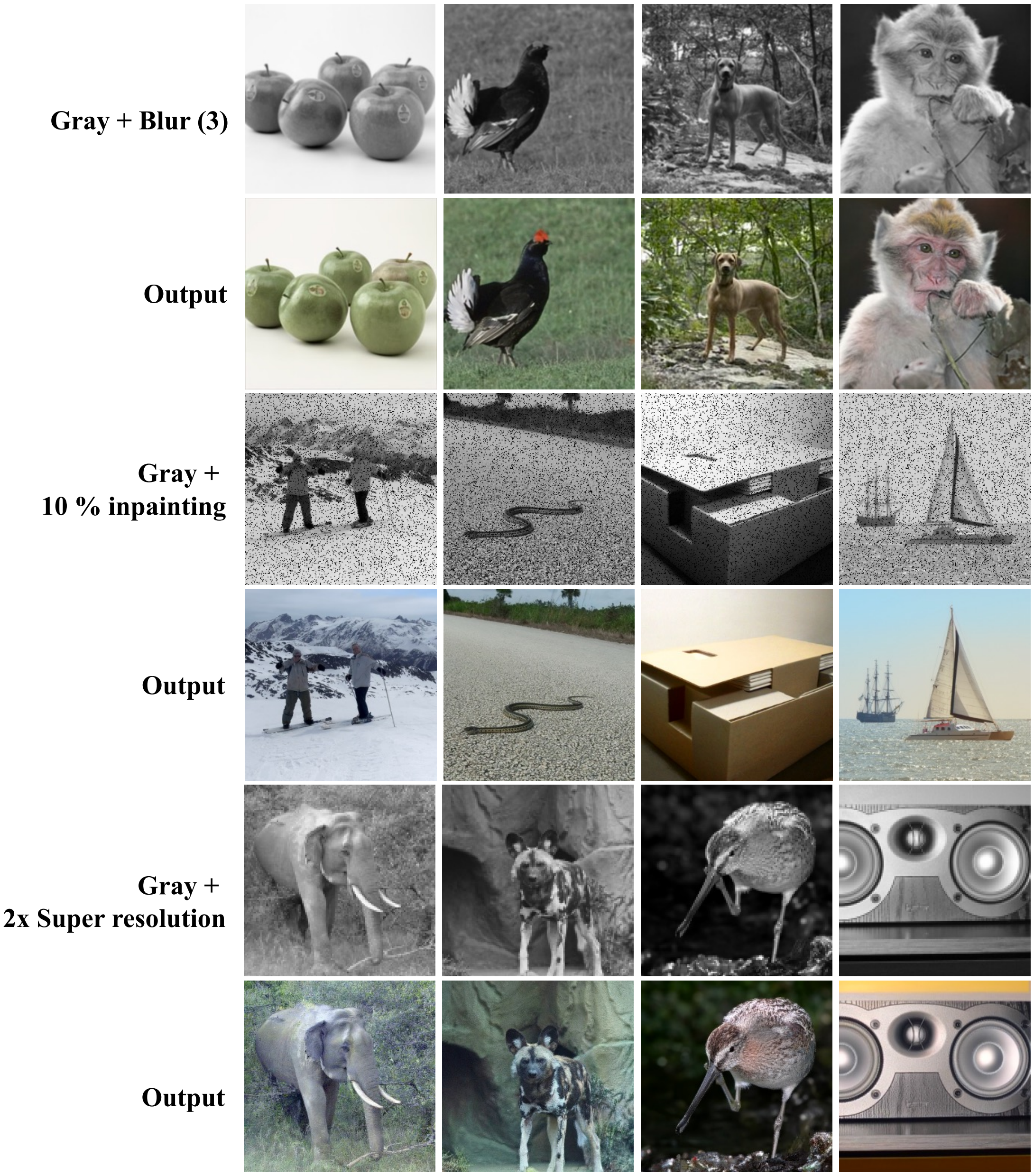


Figure 20. **Samples from the multi-degradation tasks on 256 × 256 ImageNet 1K.** It is shown that GDP can recover the corrupted images undergoing multiple degradations, such as gray + blur, gray + inpainting, and gray + down-sampling. It is noted that multi-linear degradation should be only one degradation model that will damage the contents of the images. In other words, the restoration will be more difficult if two content-damaged degradations occur at the same time, such as down-sampling + mask.

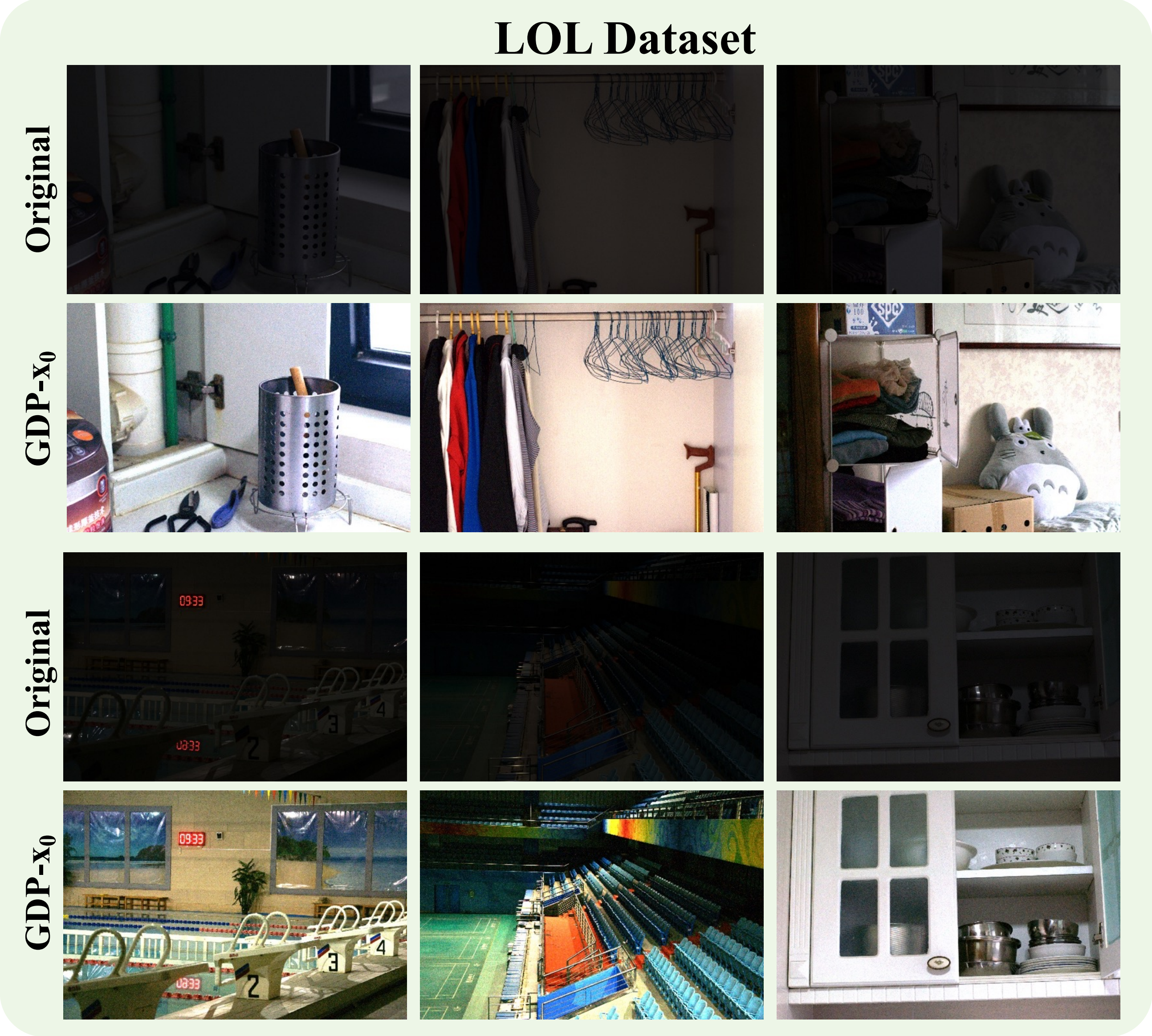
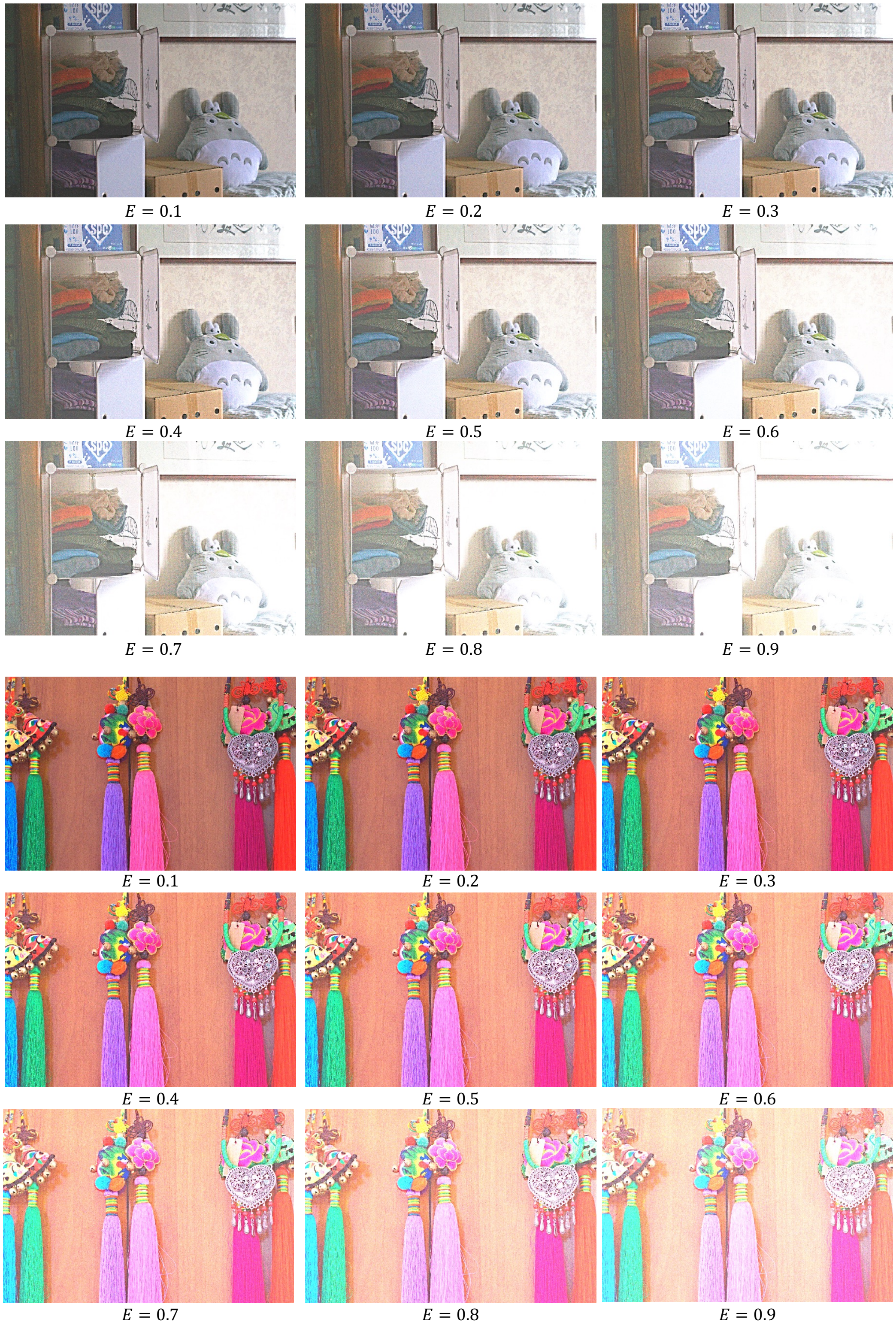


Figure 21. **Results of low-light image enhancement on LOL dataset.**



$E = 0.1$ $E = 0.2$ $E = 0.3$

$E = 0.4$ $E = 0.5$ $E = 0.6$

$E = 0.7$ $E = 0.8$ $E = 0.9$

$E = 0.1$ $E = 0.2$ $E = 0.3$

$E = 0.4$ $E = 0.5$ $E = 0.6$

$E = 0.7$ $E = 0.8$ $E = 0.9$

Figure 22. **Results of light control on LOL dataset.** We can adjust the brightness of the generated images with the help of Exposure Control Loss. Users can adjust the gray level E in the RGB color space to obtain the images according to their needs.

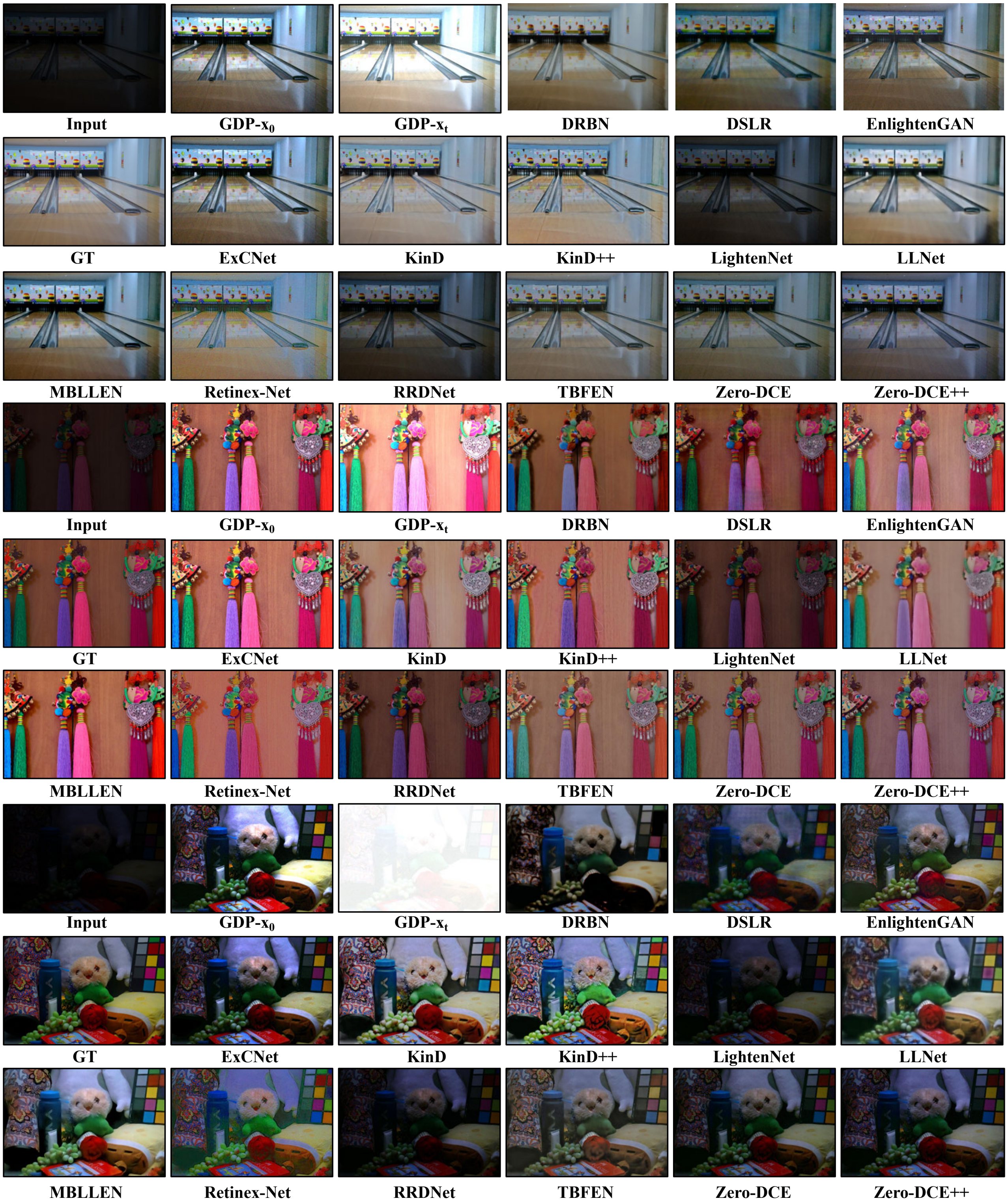


Figure 23. **The comparison of our GDP and other methods on the LOL datasets towards low-light enhancement.**

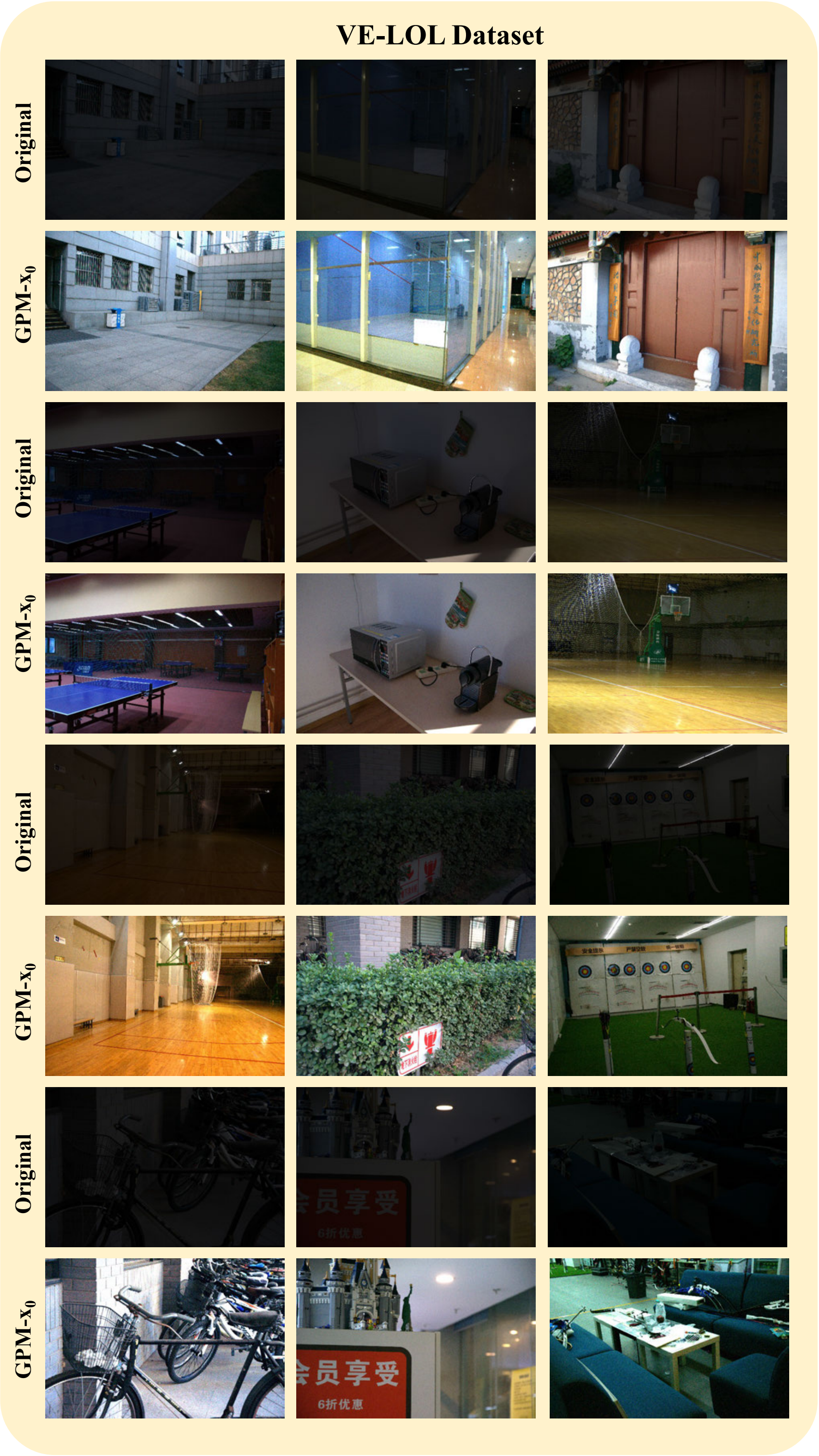


Figure 24. **Results of low-light image enhancement on VE-LOL-L dataset.**

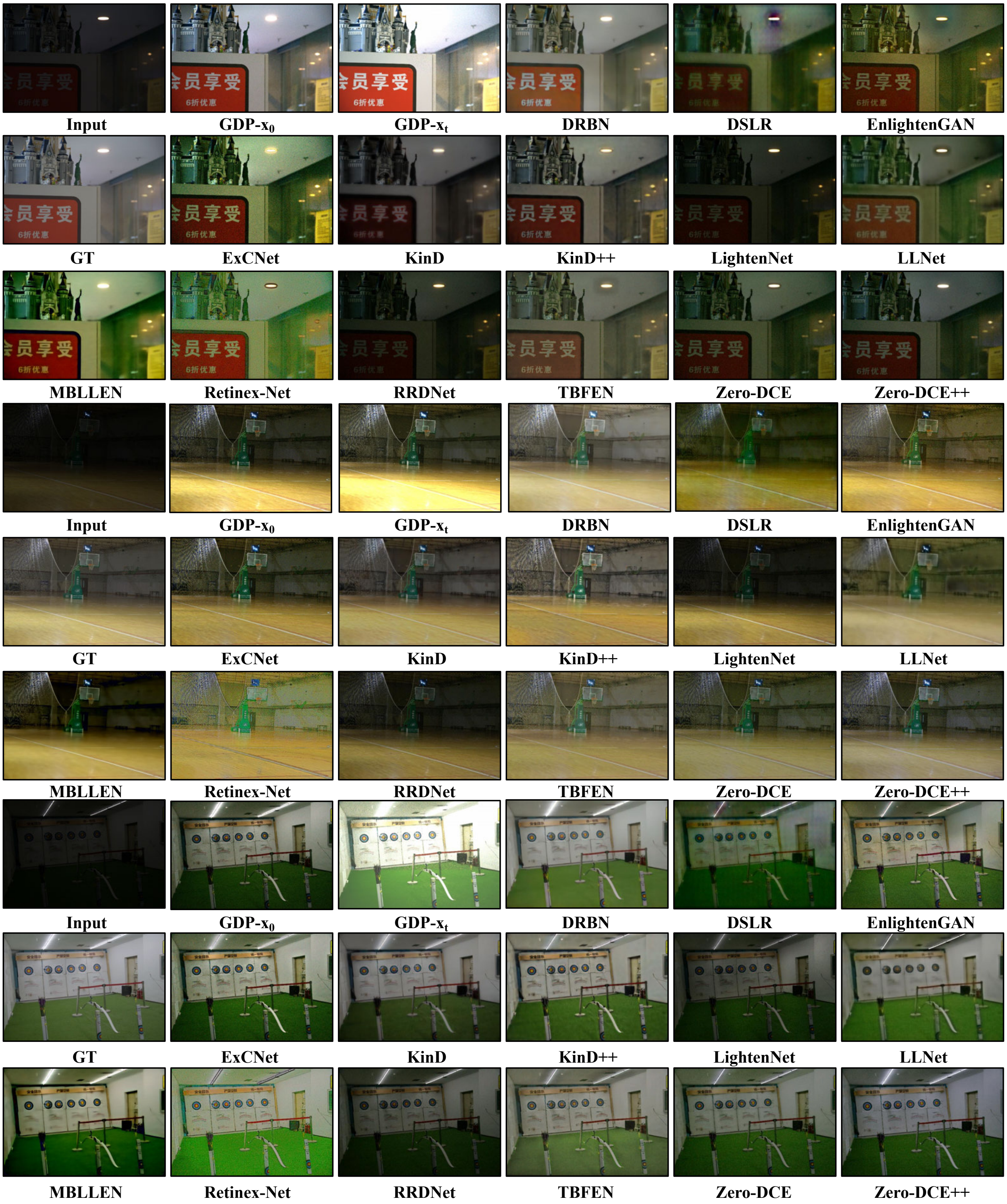


Figure 25. **The comparison of our GDP and other methods on the VE-LOL-L datasets towards low-light enhancement.**

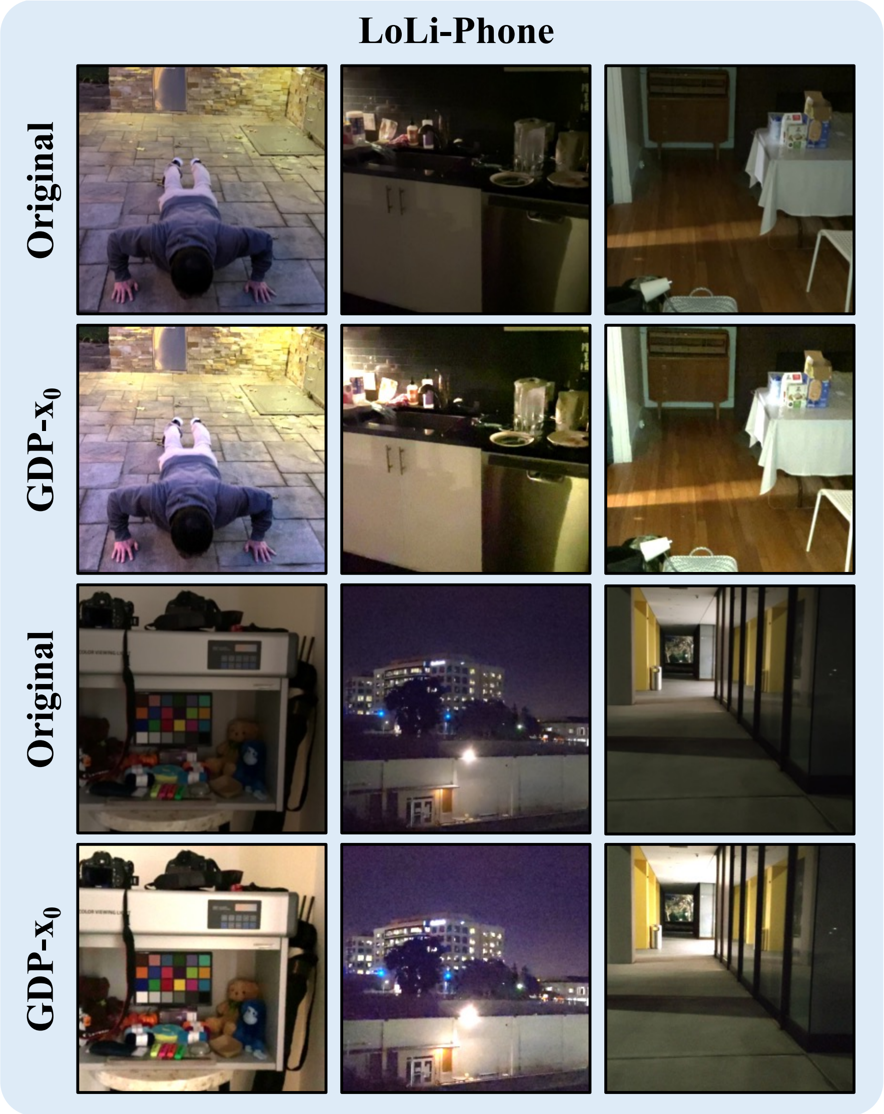


Figure 26. **Results of low-light image enhancement on LoLi-Phone dataset.**

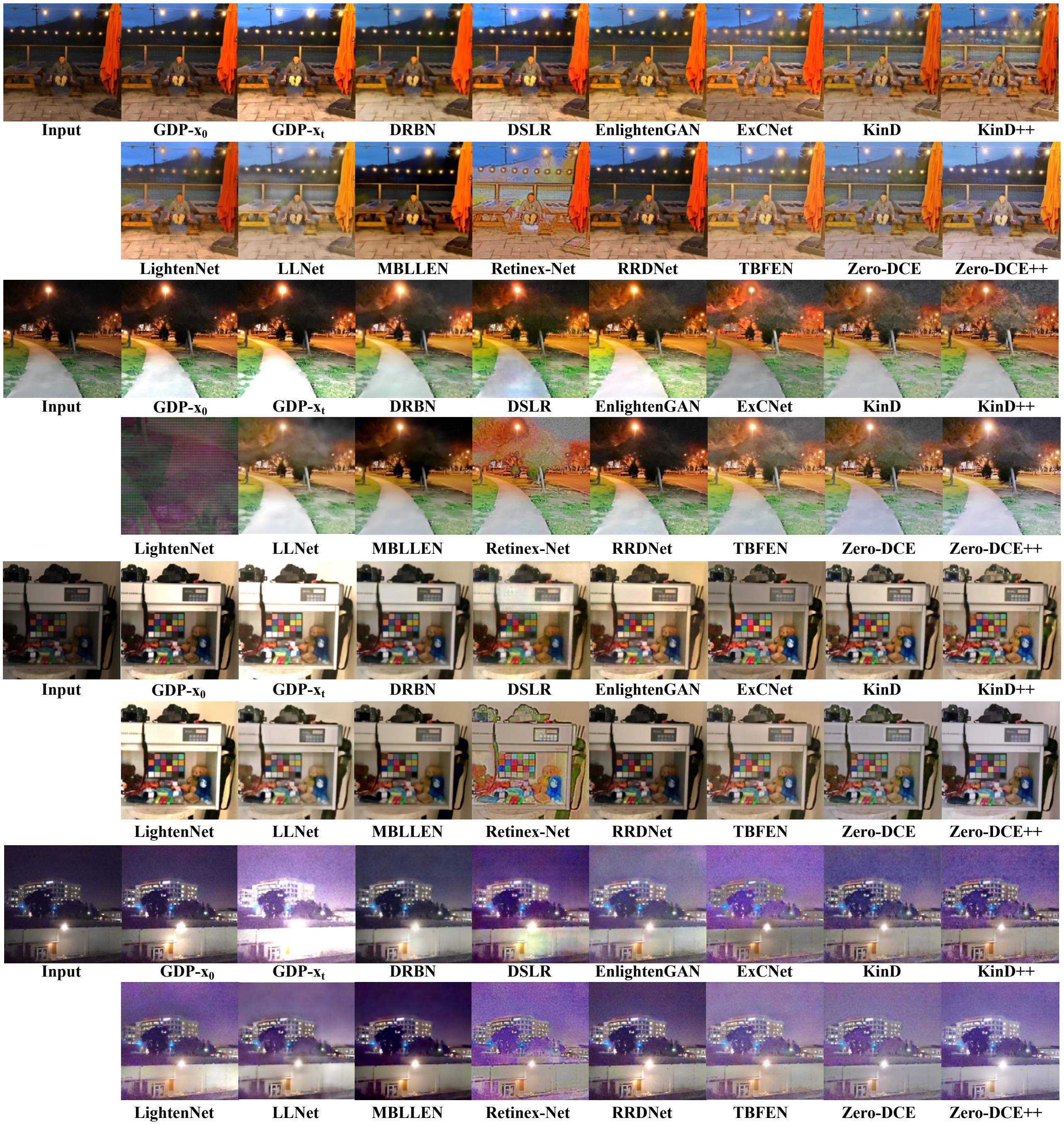


Figure 27. **The comparison of our GDP and other methods on the LoLi-phone datasets towards low-light enhancement.**

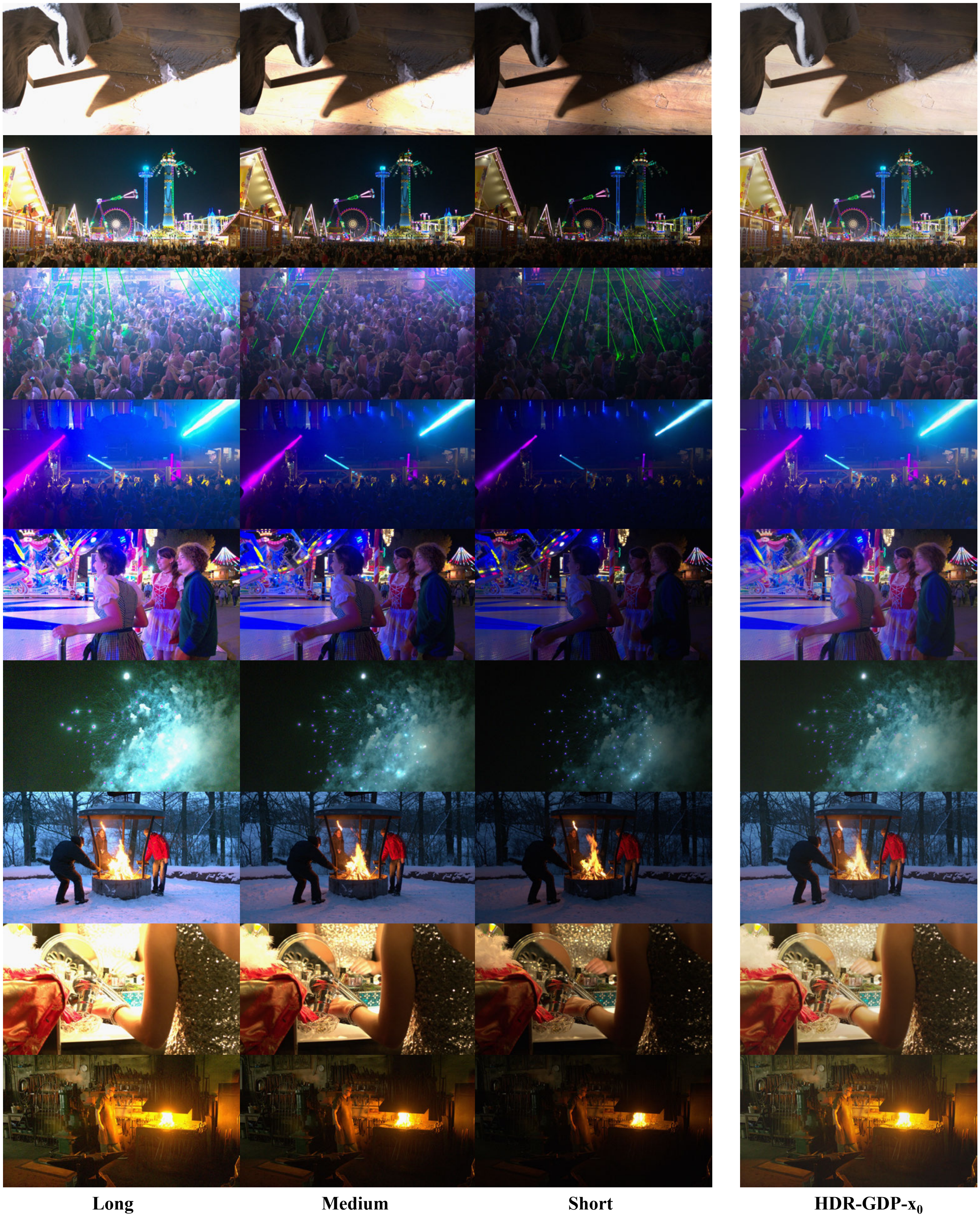
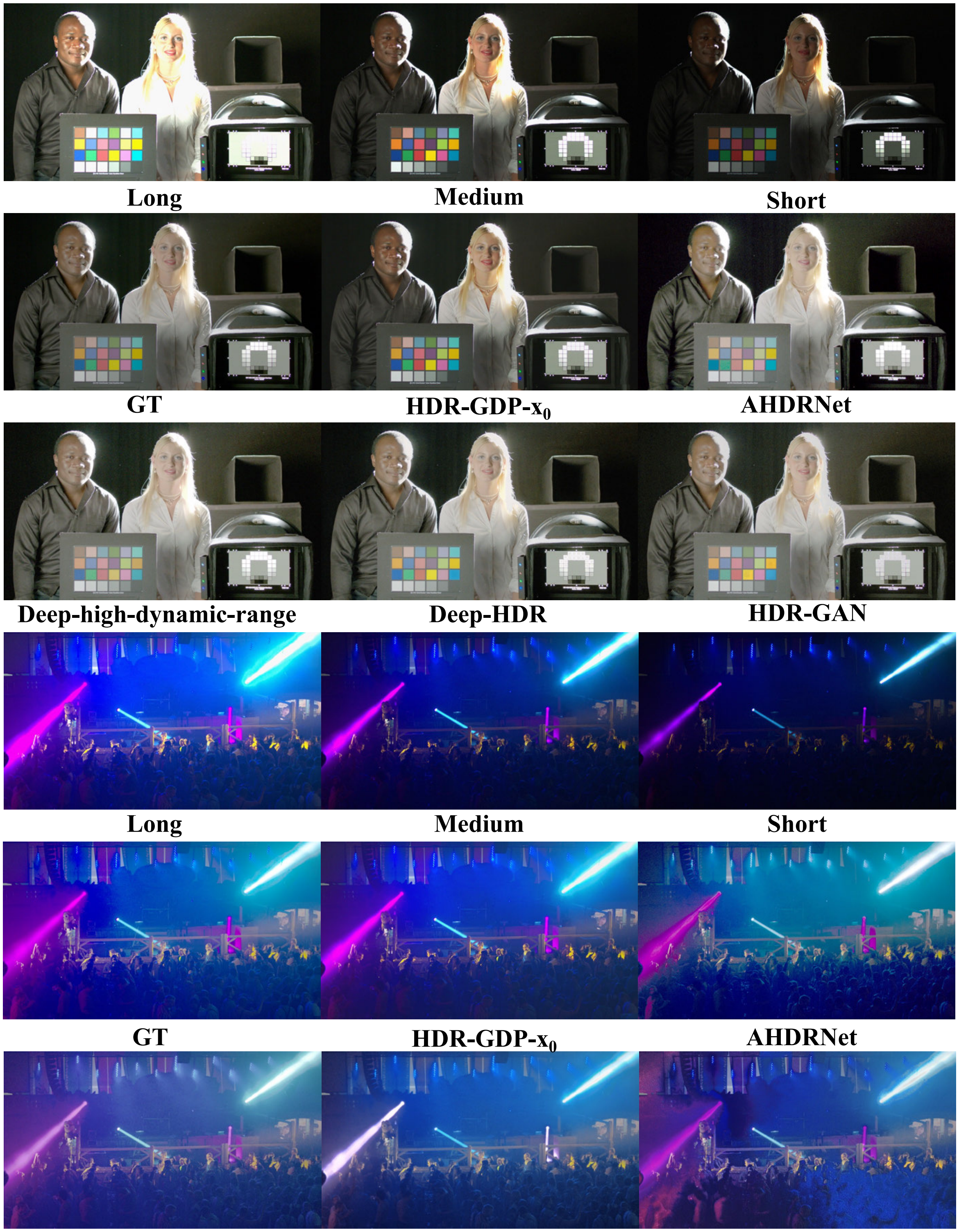


Figure 28. **Results of HDR image recovery on NTIRE2021 dataset.**



Long Medium Short

GT HDR-GDP-x_0 AHDRNet

Deep-high-dynamic-range Deep-HDR HDR-GAN

Long Medium Short

GT HDR-GDP-x_0 AHDRNet

Deep-high-dynamic-range Deep-HDR HDR-GAN

Figure 29. **The comparison of HDR image recovery on NTIRE2021 dataset.**

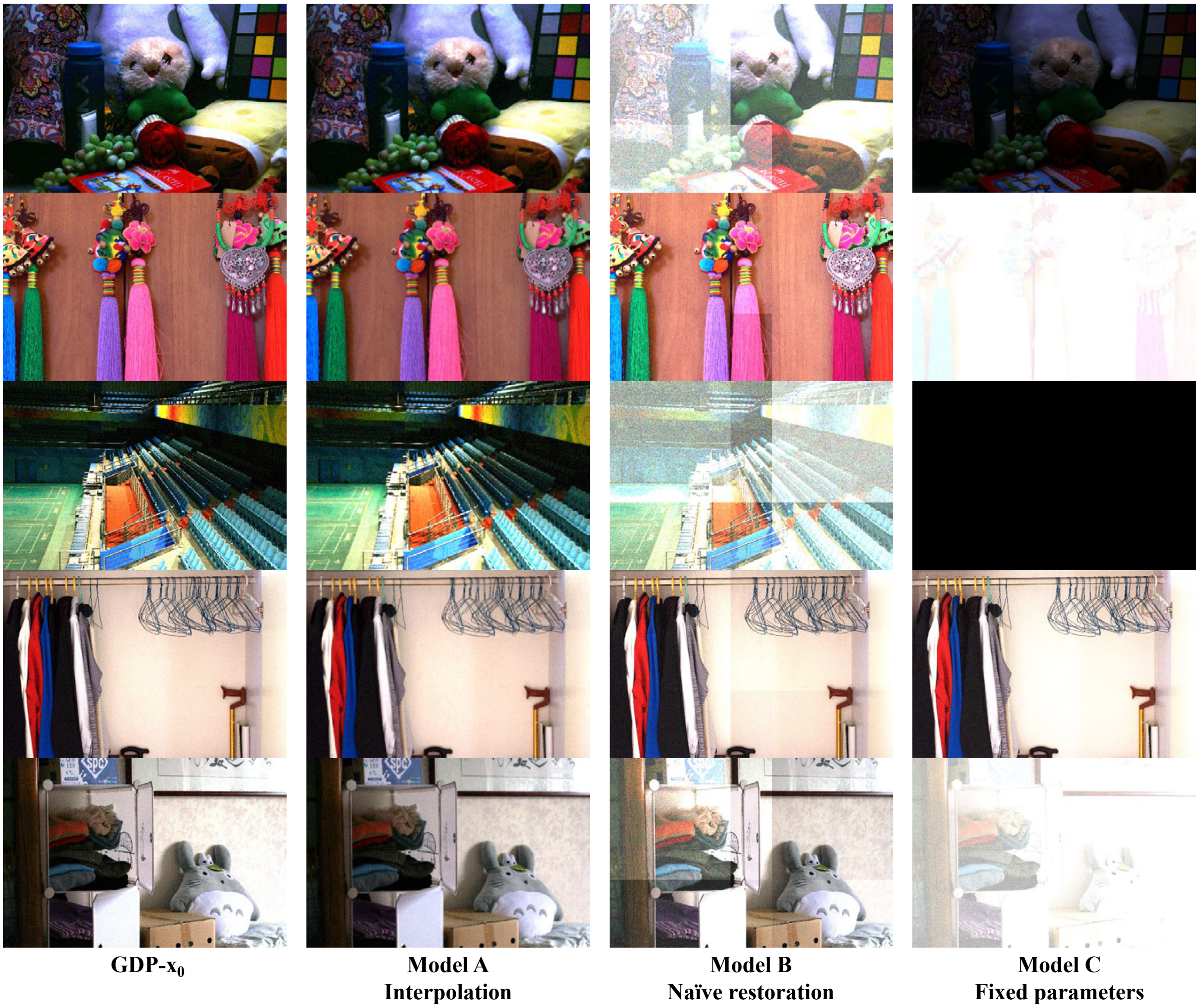
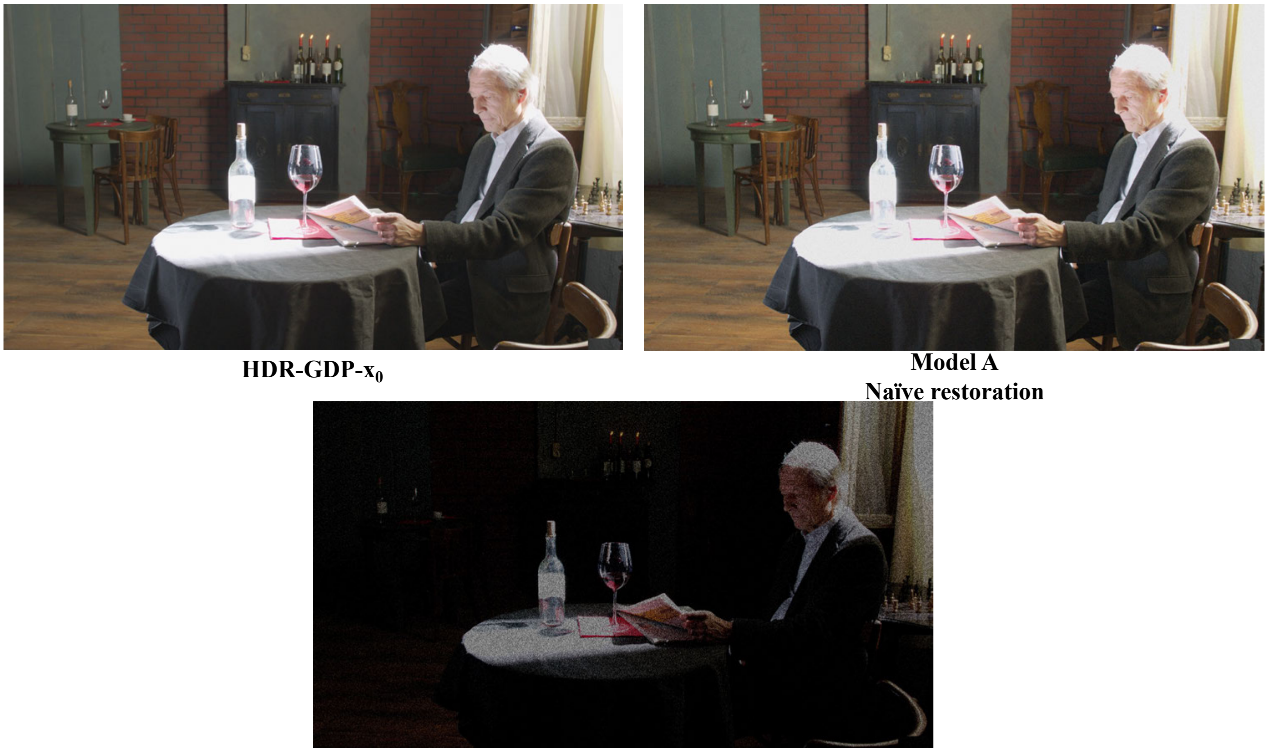


Figure 30. **Qualitative comparison of ablation study on LOL dataset.** Model A recovers the images in $256 \times N$ or $256 \times N$ sizes and is interpolated by the nearest neighbor to the original size. Model B is devised to naively restore the images from patches and patches where the parameters are not related. Model C is designed with fixed parameters for all patches in the images.



HDR-GDP-x_0

Model A
Naïve restoration

Model B
Fixed parameters

Figure 31. **Qualitative comparison of ablation study on NTIRE2021 dataset.**

References

[1] Yu Chen, Ying Tai, Xiaoming Liu, Chunhua Shen, and Jian Yang. Fsrnet: End-to-end learning face super-resolution with facial priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2492–2501, 2018. 7

[2] Ryan Dahl, Mohammad Norouzi, and Jonathon Shlens. Pixel recursive super resolution. In *Proceedings of the IEEE international conference on computer vision*, pages 5439–5448, 2017. 7

[3] Alexey Dosovitskiy and Thomas Brox. Generating images with perceptual similarity metrics based on deep networks. *Advances in neural information processing systems*, 29, 2016. 7

[4] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 4

[5] Nima Khademi Kalantari, Ravi Ramamoorthi, et al. Deep high dynamic range imaging of dynamic scenes. *ACM Trans. Graph.*, 36(4):144–1, 2017. 10

[6] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. *arXiv preprint arXiv:2201.11793*, 2022. 3, 8

[7] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5549–5558, 2020. 3

[8] Chongyi Li, Chunle Guo, Ling-Hao Han, Jun Jiang, Ming-Ming Cheng, Jinwei Gu, and Chen Change Loy. Low-light image and video enhancement using deep learning: A survey. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 44(01):1–1, 2021. 3

[9] Jiaying Liu, Dejia Xu, Wenhan Yang, Minhao Fan, and Haofeng Huang. Benchmarking low-light image enhancement and beyond. *International Journal of Computer Vision*, 129(4):1153–1184, 2021. 3

[10] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015. 3

[11] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal processing letters*, 20(3):209–212, 2012. 5

[12] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 6

[13] Yuzhen Niu, Jianbin Wu, Wenxi Liu, Wenzhong Guo, and Rynson WH Lau. Hdr-gan: Hdr image reconstruction from multi-exposed ldr images with large motions. *IEEE Transactions on Image Processing*, 30:3885–3896, 2021. 10

[14] Ozan Özdenizci and Robert Legenstein. Restoring vision in adverse weather conditions with patch-based denoising diffusion models. *arXiv preprint arXiv:2207.14626*, 2022. 5

[15] Xingang Pan, Xiaohang Zhan, Bo Dai, Dahua Lin, Chen Change Loy, and Ping Luo. Exploiting deep generative prior for versatile image restoration and manipulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 3, 6

[16] Eduardo Pérez-Pellitero, Sibi Catley-Chandar, Ales Leonardis, and Radu Timofte. Ntire 2021 challenge on high dynamic range imaging: Dataset, methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 691–700, 2021. 3, 6

[17] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022. 2

[18] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 4, 7

[19] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 7

[20] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 5

[21] Shuhang Wang, Jin Zheng, Hai-Miao Hu, and Bo Li. Naturalness preserved enhancement algorithm for non-uniform illumination images. *IEEE transactions on image processing*, 22(9):3538–3548, 2013. 5

[22] Allan G Weber. The usc-sipi image database: Version 5. *http://sipi. usc. edu/database/*, 2006. 3

[23] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. *arXiv preprint arXiv:1808.04560*, 2018. 3, 6

[24] Shangzhe Wu, Jiarui Xu, Yu-Wing Tai, and Chi-Keung Tang. Deep high dynamic range imaging with large foreground motions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 117–132, 2018. 10

[25] Qingsen Yan, Dong Gong, Qinfeng Shi, Anton van den Hengel, Chunhua Shen, Ian Reid, and Yanning Zhang. Attention-guided network for ghost-free high dynamic range imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1751–1760, 2019. 10

[26] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 3

[27] Junzhe Zhang, Xinyi Chen, Zhongang Cai, Liang Pan, Haiyu Zhao, Shuai Yi, Chai Kiat Yeo, Bo Dai, and Chen Change Loy. Unsupervised 3d shape completion through gan inversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1768–1777, 2021. 2

[28] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 5