Selective Hourglass Mapping for Universal Image Restoration Based on Diffusion Model
Supplementary Material

Dian Zheng[1] Xiao-Ming Wu[1] Shuzhou Yang[2] Jian Zhang[2] Jian-Fang Hu[1] Wei-Shi Zheng[1,3*]
[1]School of Computer Science and Engineering, Sun Yat-sen University, China
[2]School of Electronic and Computer Engineering, Peking University, China
[3]Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, China
{zhengd35, wuxm65}@mail2.sysu.edu.cn wszheng@ieee.org

A. Standard Diffusion Model

Diffusion model is a novel generative model based on the Markov Chain, which contains a non-parameter noise-adding forward process and a denoising reverse process. We show the details of each process below.

A.1. Forward Process

In the forward process, diffusion model gradually adds noise to an image, the one-step noise adding could be written as follows:

$$x_t = \sqrt{\alpha}_t x_{t-1} + \sqrt{1-\alpha_t}\epsilon_{t-1}, \tag{1}$$

where α is the manually designed noise coefficient variation over time t and ϵ is the added noise. Based on the theory of Markov Chain, it could be approximated to the formula which could diffuse to any-step with only one step:

$$x_t = \sqrt{\overline{\alpha}_t} x_0 + \sqrt{1-\overline{\alpha}_t}\epsilon, \tag{2}$$

where $\overline{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, ϵ is the random Gaussian noise.

A.2. Reverse Process

In the reverse process, diffusion model recovers the target sample from the standard Gaussian noise step by step (*e.g.*, $x_T, x_{T-1}, \ldots, x_1, x_0$), which could be written as:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_\theta^2(x_t, t)), \tag{3}$$

where μ_θ and σ_θ^2 is the mean and variance of the distribution $p_\theta(x_{t-1}|x_t)$, θ means the term is obtain by the model.

There are two widely used sampling strategies in the reverse process of diffusion model: DDPM [2] like full-step sampling and DDIM [13] like skip-step sampling.

Table S1. Summary of the datasets used in the paper.

Dataset	Train set	Test set
Dehazing		
OTS$_{ALPHA}$ [6]	313950	-
SOTS [6]	-	500
Low-light enhancement		
LOL [15]	485	15
DICM [5]	-	64
MEF [9]	-	17
NPE [14]	-	8
Merged Deraining		
Rain800 [18]	700	100
Rain1800 [16]	1800	-
Rain14000 [1]	11200	2800
Rain1200 [17]	-	1200
Rain12 [7]	12	-
Rain100H [16]	-	100
Rain100L [16]	-	100
Practical [16]	-	15
Merged Desnowing		
Snow100K [8]	50000	-
Snow100K-L [8]	-	xxx
Snow100K-S [8]	-	xxx
Snow100K-Real [8]	-	1329
Merged Debluring		
GoPro [10]	2103	1111
HIDE [12]	-	2025
RealBlur-R [11]	-	980
RealBlur-J [11]	-	980
Under-Display Camera		
POLED [19]	-	30
TOLED [19]	-	30

*Corresponding author

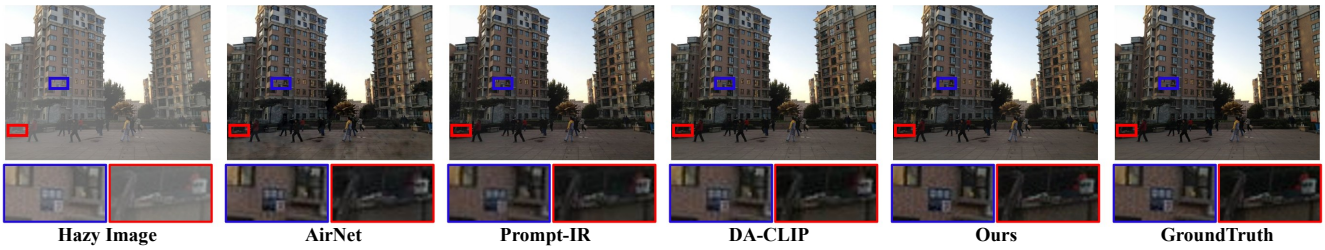

Figure S1. Visualization comparison with state-of-the-art methods on dehazing. Zoom in for best view.

For DDPM sampling strategy, it denoises step by step as mentioned above, the equations are as follows:

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\overline{\alpha}_t}}\epsilon_\theta\left(x_t, t\right)\right), \quad (4)$$

$$x_{t-1} = \sigma_\theta(x_t, t)z + \mu_\theta(x_t, t), \quad (5)$$

where x_t is the recovered image in timestep t, ϵ_θ is the noise predicted by the model, $\beta_t = 1 - \alpha_t$, $\sigma_\theta(x_t, t)$ can be designed manually and z is the random Guassian noise.

For the DDIM sampling strategy, diffusion model performs the skip-step operation. For instance, the whole timestep is 1000 and DDIM sampling strategy could accelerate it into 5 steps (*i.e.*, 1000, 750, 500, 250, 0). The formula is:

$$\begin{aligned} x_{t-1} = &\frac{1}{\sqrt{\alpha_t}}\left(x_t - \sqrt{1-\overline{\alpha}_t}\epsilon_\theta(x_t, t)\right) \\ &+ \sqrt{1-\alpha_{t-1}}\epsilon_\theta\left(x_t, t\right). \end{aligned} \quad (6)$$

Note that there is no extensive random noise term as the deterministic implicit sampling equation [13] sets the noise coefficient as 0.

A.3. Training Objective

The simple version of the training objective is calculated as follows:

$$\mathcal{L}_{simple} = \mathbb{E}_{x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta\left(\sqrt{\overline{\alpha}_t}x_0 + \sqrt{1-\overline{\alpha}_t}\epsilon\right)\|^2\right], \quad (7)$$

where x_0 is the input image, ϵ is the noise established in forward process.

B. Summary about the Datasets

The overall datasets we used in the paper are shown in Table S1. We use the largest and most widely known datasets for each task to validate our DiffUIR in universal image restoration and real-world generalization settings.

C. Complete Derivation of our DiffUIR

As the derivation of the distribution approaching forward process is complete in the paper, we present the complete version of distribution diffusing reverse process and universal training objective here.

C.1. Distribution Diffusing Reverse Process

In the reverse process, we recover the sample from the shared distribution (*e.g.*, $I_T = (1-\overline{\delta}_T)I_{in} + \overline{\beta}_T\epsilon$) to the task-specific distribution. Following the DDPM [2], we use the $q(I_{t-1}|I_t, I_{in}, I_0^\theta, I_{res}^\theta)$ to simulate the distribution of $p_\theta(I_{t-1}|I_t)$ and based on the Bayes' theorem, we could calculate it as follows:

$$\begin{aligned} &p_\theta(I_{t-1}|I_t) \rightarrow q(I_{t-1}|I_t, I_{in}, I_0^\theta, I_{res}^\theta) \\ &= q(I_t|I_{t-1}, I_{in}, I_{res}^\theta)\frac{q(I_{t-1}|I_0^\theta, I_{res}^\theta, I_{in})}{q(I_t|I_0^\theta, I_{res}^\theta, I_{in})}, \end{aligned} \quad (8)$$

where $q(I_t|I_{t-1}, I_{in}, I_{res}^\theta)$ could be calculated by the equation in the paper of Line 231; $q(I_{t-1}|I_0^\theta, I_{res}^\theta, I_{in})$ and $q(I_t|I_0^\theta, I_{res}^\theta, I_{in})$ is calculated by Eq (4) in the paper. As our goal is to represent the distribution of $p_\theta(I_{t-1}|I_t)$ by the mean and the variance. We rearrange it to the probability density function form of the normal distribution (*i.e.*, $f(x) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$) by utilizing the probability density function of the three terms above. As the mean and variance could be calculated by the exponential term of the probability density function, we only consider this term as follows:

$$\begin{aligned} &q(I_t|I_{t-1}, I_{in}, I_{res}^\theta)\frac{q(I_{t-1}|I_0^\theta, I_{res}^\theta, I_{in})}{q(I_t|I_0^\theta, I_{res}^\theta, I_{in})} \\ &\propto exp\left[-\frac{1}{2}\left(\frac{(I_t - I_{t-1} - \alpha_t I_{res}^\theta + \delta_t I_{in})^2}{\beta_t^2}\right.\right. \\ &+\frac{(I_{t-1} - I_0^\theta - \overline{\alpha}_{t-1}I_{res}^\theta + \overline{\delta}_{t-1}I_{in})^2}{\overline{\beta}_{t-1}^2} \\ &\left.\left.-\frac{(I_t - I_0^\theta - \overline{\alpha}_t I_{res}^\theta + \overline{\delta}_t I_{in})^2}{\overline{\beta}_t^2}\right)\right] \\ &= exp\left[-\frac{1}{2}\left(\left(\frac{\overline{\beta}_t^2}{\beta_t^2\overline{\beta}_{t-1}^2}\right)I_{t-1}^2 - 2\left(\frac{I_t + \delta_t I_{in} - \alpha_t I_{res}^\theta}{\beta_t^2}\right.\right.\right. \\ &\left.\left.\left.+\frac{I_0^\theta + \overline{\alpha}_{t-1}I_{res}^\theta - \overline{\delta}_{t-1}I_{in}}{\overline{\beta}_{t-1}^2}\right)I_{t-1} + C(I_t, I_0^\theta, I_{res}^\theta, I_{in})\right)\right]. \end{aligned} \quad (9)$$

Based on the Eq. (9) and the property of the probability density function of the normal distribution, the mean and

Table S2. Ablation study on batch size.

Batch size	Deraining $(5sets)$		Enhancement		Desnowing$(2sets)$		Dehazing		Deblurring	
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
1	27.07	0.849	16.83	0.528	29.37	0.892	30.24	0.951	26.77	0.812
2	27.69	0.865	17.01	0.544	30.13	0.901	31.23	0.953	26.87	0.812
5	30.68	0.897	**25.23**	**0.910**	32.09	0.923	30.68	0.952	29.10	0.863
10	**31.03**	**0.904**	25.12	0.907	**32.65**	**0.927**	**32.94**	**0.956**	**29.17**	**0.864**

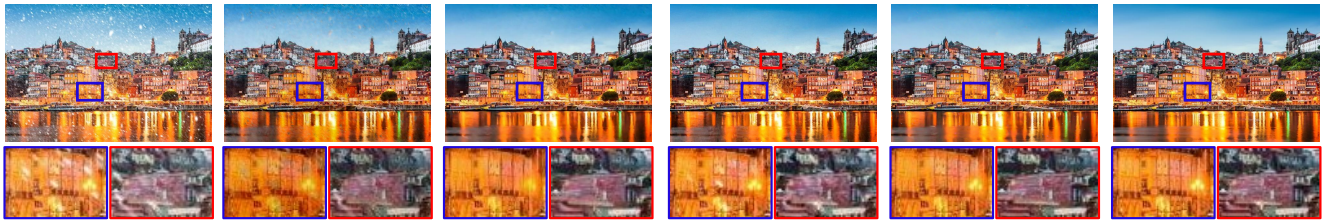

Figure S2. Visualization comparison with state-of-the-art methods on desnowing. Zoom in for best view.

Table S3. Ablation study on sampling steps.

Task	3 steps		5 steps		10 steps	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Deraining	31.03	0.904	31.06	0.904	**31.09**	**0.905**
Low-light	25.12	0.907	25.17	0.907	**25.31**	**0.908**
Dehazing	32.94	0.956	33.00	0.956	**33.06**	**0.956**

variance of the distribution $p_\theta(I_{t-1}|I_t)$ could be calculated by "$-\frac{b}{2a}$" and "$\frac{1}{a}$". The results are as follows:

$$\begin{aligned}\mu_\theta(I_t,t) &= I_t - \alpha_t I^\theta_{res} + \delta_t I_{in} - \frac{\beta_t^2}{\overline{\beta}_t}\epsilon^\theta \\ \sigma^2_\theta(I_t,t) &= \frac{\beta_t^2\overline{\beta}^2_{t-1}}{\overline{\beta}^2_t},\end{aligned} \tag{10}$$

where I^θ_{res} is predicted by the model and ϵ^θ is obtained by the I^θ_{res}. Based on the reparameterization [3, 4] technology, if we use the sampling strategy from the DDPM [2], I_{t-1} could be calculated as follows:

$$I_{t-1} = I_t - \alpha_t I^\theta_{res} + \delta_t I_{in} - \frac{\beta_t^2}{\overline{\beta}_t}\epsilon^\theta + \frac{\beta_t\overline{\beta}_{t-1}}{\overline{\beta}_t}\epsilon_*, \tag{11}$$

where ϵ_* is the random Gaussian noise. In this paper, to accelerate the sampling speed, we use the deterministic implicit sampling strategy of DDIM [13], based on Eq. (3) in the paper, the I_{t-1} is as follows:

$$I_{t-1} = I^\theta_0 + \overline{\alpha}_{t-1} I^\theta_{res} - \overline{\delta}_{t-1} I_{in} + \overline{\beta}_{t-1}\epsilon^\theta, \tag{12}$$

as I^θ_0 could be transformed to $I_t - \overline{\alpha}_t I^\theta_{res} - \overline{\beta}_t\epsilon^\theta + \overline{\delta}_t I_{in}$, the formula is calculated by:

$$\begin{aligned}I_{t-1} &= I_t - (\overline{\alpha}_t - \overline{\alpha}_{t-1})I^\theta_{res} \\ &\quad - (\overline{\beta}_t - \overline{\beta}_{t-1})\epsilon^\theta + (\overline{\delta}_t - \overline{\delta}_{t-1})I_{in} \\ &= I_t - \alpha_t I^\theta_{res} - (\overline{\beta}_t - \overline{\beta}_{t-1})\epsilon^\theta + \delta_t I_{in},\end{aligned} \tag{13}$$

as in our implementation, the value of $\overline{\beta}_t - \overline{\beta}_{t-1}$ is nearly zero and has no impact on the performance, we eschew this term. The final version of our DDIM sampling is:

$$I_{t-1} = I_t - \alpha_t I^\theta_{res} + \delta_t I_{in}. \tag{14}$$

C.2. Universal Training Objective

Here we show how we derive from Eq. (10) to Eq. (11) in the paper. Based on the Eq. (4) in the paper, the ϵ_θ and ϵ could be calculated as follows:

$$\epsilon_\theta = \frac{I_t - I^\theta_0 - \overline{\alpha}_t I^\theta_{res} + \overline{\delta}_t I_{in}}{\overline{\beta}_t} = \frac{(1-\overline{\alpha}_t)I^\theta_{res} + \overline{\delta}_t I_{in}}{\overline{\beta}_t}. \tag{15}$$

$$\epsilon = \frac{I_t - I_0 - \overline{\alpha}_t I_{res} + \overline{\delta}_t I_{in}}{\overline{\beta}_t} = \frac{(1-\overline{\alpha}_t)I_{res} + \overline{\delta}_t I_{in}}{\overline{\beta}_t}. \tag{16}$$

Incorporating Eq. (15) and (16) into Eq. (10) in the paper, one obtains:

$$\begin{aligned}\mathcal{L}(\theta)_{simple} &= \mathbb{E}_{t,I_t,I_{res}}\Big[\| - \alpha_t(I_{res} - I^\theta_{res}(I_t,t)) \\ &\quad + \frac{(1-\overline{\alpha}_t)\beta_t^2}{\overline{\beta}^2_t}(I_{res} - I^\theta_{res}(I_t,t))\|_1\Big] \\ &= \mathbb{E}_{t,I_t,I_{res}}\left[\|C(\alpha,\beta,t)(I_{res} - I^\theta_{res}(I_t,t))\|_1\right],\end{aligned} \tag{17}$$

where $C(\alpha,\beta,t)$ is a constant that has no effect on the training objective, eschewing this term, the final version of our universal training objective (*e.g.*, Eq. (11)) is obtained.

D. More Experimental Results

D.1. Ablation Study

Impact of the batch size. We show the result in Table S2. It could be seen that when the batch size is small, the performance of different tasks varies largely as the model could

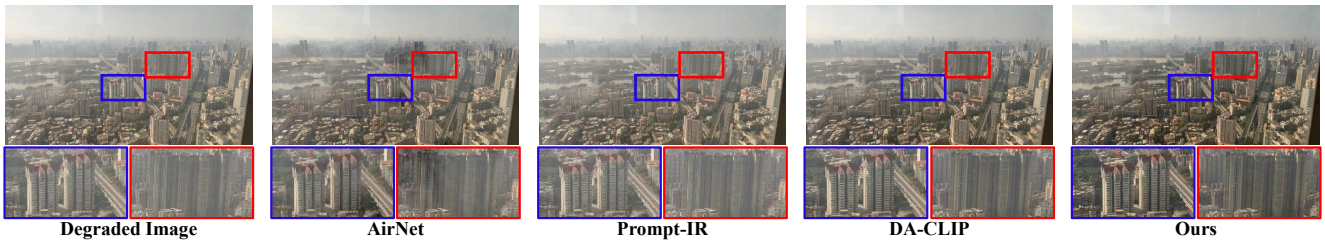

Figure S3. Visualization comparison with state-of-the-art methods on real-world dehazing. Zoom in for best view.

Figure S4. Visualization comparison with state-of-the-art methods on real-world multi-degradation scenarios. Our method achieves outstanding success benefitting from the selective hourglass mapping. Zoom in for best view.

not learn the shared distribution among all of the tasks in one batch. Additionally, the performance of batch size 10 surpasses the 5 as when batch size reaches 10, we adjust the weight of different tasks in one batch based on the size of the datasets. The whole results validate that the number of batch size is significant for our universal image restoration learning.

More sampling steps. We show the result of three degradation tasks in Table S3. The performance is improved with more sampling steps which is similar to the property of other diffusion methods. What's more, the experiment validates that our modification of the diffusion algorithm is rational and rigorous.

D.2. Visual Comparison

We show the visualization results of dehazing, desnowing and real-world generalization in Fig. S1 to Fig. S4 respectively. Our DiffUIR generates more steady and fidelity images than other universal image restoration methods. Especially, when various degradation occurs in one image, benefiting from the selective hourglass mapping, we achieve outstanding recovery results, which validate our motivation in the paper of Line 44.

References

[1] Xueyang Fu, Jiabin Huang, Delu Zeng, Yue Huang, Xinghao Ding, and John Paisley. Removing rain from single images via a deep detail network. In *CVPR*, 2017. 1

[2] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 1, 2, 3

[3] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014. 3

[4] Diederik P Kingma, Max Welling, et al. An introduction to variational autoencoders. *FTML*, 2019. 3

[5] Chulwoo Lee, Chul Lee, and Chang-Su Kim. Contrast enhancement based on layered difference representation. In *ICIP*, 2012. 1

[6] Boyi Li, Wenqi Ren, Dengpan Fu, Dacheng Tao, Dan Feng, Wenjun Zeng, and Zhangyang Wang. Benchmarking single-image dehazing and beyond. *TIP*, 2018. 1

[7] Yu Li, Robby T Tan, Xiaojie Guo, Jiangbo Lu, and Michael S Brown. Rain streak removal using layer priors. In *CVPR*, 2016. 1

[8] Yun-Fu Liu, Da-Wei Jaw, Shih-Chia Huang, and Jenq-Neng Hwang. Desnownet: Context-aware deep network for snow removal. *TIP*, 2018. 1

[9] Kede Ma, Kai Zeng, and Zhou Wang. Perceptual quality assessment for multi-exposure image fusion. *TIP*, 2015. 1

[10] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2017. 1

[11] Jaesung Rim, Haeyun Lee, Jucheol Won, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *ECCV*, 2020. 1

[12] Ziyi Shen, Wenguan Wang, Xiankai Lu, Jianbing Shen, Haibin Ling, Tingfa Xu, and Ling Shao. Human-aware motion deblurring. In *ICCV*, 2019. 1

[13] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2020. 1, 2, 3

[14] Shuhang Wang, Jin Zheng, Hai-Miao Hu, and Bo Li. Naturalness preserved enhancement algorithm for non-uniform illumination images. *TIP*, 2013. 1

[15] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. In *BMVC*, 2018. 1

[16] Wenhan Yang, Robby T Tan, Jiashi Feng, Jiaying Liu, Zongming Guo, and Shuicheng Yan. Deep joint rain detection and removal from a single image. In *CVPR*, 2017. 1

[17] He Zhang and Vishal M Patel. Density-aware single image

de-raining using a multi-stream dense network. In *CVPR*, 2018. 1

[18] He Zhang, Vishwanath Sindagi, and Vishal M Patel. Image de-raining using a conditional generative adversarial network. *TCSVT*, 2019. 1

[19] Yuqian Zhou, David Ren, Neil Emerton, Sehoon Lim, and Timothy Large. Image restoration for under-display camera. In *CVPR*, 2021. 1