Supplementary Materials: Learning Degradation-Independent Representations for Camera ISP Pipelines

A Simulated Image Processing Pipelines

As in [1, 10], we follow the most common ISP modules, which include the following typical stages:

(1) Optics and Sensor: The sensor captures illumination through the optics and produces RAW images with pixel values that are proportional to the intensity of illumination.
(2) Noise Reduction: Noise reduction is the process of removing the noise from the imaging signals, which will also blur the images. The noise is generally considered to be a random variable with zero mean in the simulation.
(3) Demosaicking: The RAW pixels of the color filter array pattern are converted to RGB values for each pixel by performing interpolation.
(4) Digital Gain and White Balance: The RAW pixel values are gain-adjusted and a white-balance correction is applied to the raw RGB values
(5) Color Space Transform: This step transforms the raw RGB values to CIE XYZ color space.
(6) Tone Correction: This step applies gamma curves and adjusts the image contrast by histogram operations to improve the appearance.
(7) Sharpening: This enhancement process is to compensate for the outline of the image, strengthening the edges and enriching the image details.
(8) Compression: Images are compressed by JPEG compression for storage

We adopt an ISP model [10] to mimic the steps (2)∼(8) and generate ISP-degraded images for model training. We set the operational range of noise variance to $[0.15, 0.35]$, and the operational range of compression quality factor to $[5, 45]$.

B Mutual Information Estimation

Proposition 1. *Let* $\mathbf{x}$ *and* $\mathbf{y}$ *represent two random variables, the* $\mathbf{J} = p(\mathbf{x}, \mathbf{y})$ *and* $\mathbf{M} = p(\mathbf{x})p(\mathbf{y})$ *are the joint and the product of marginals of the two variables, respectively. The mutual information between the variables satisfies:*

$$\begin{aligned} I(\mathbf{x};\mathbf{y}) &:= \mathcal{D}_{\mathrm{JSD}}(\mathbf{J};\mathbf{M}) \\ &\geq \mathbb{E}_{z\sim\mathbf{J}}\left[-\sigma(-\mathcal{F}_\omega(z))\right] - \mathbb{E}_{z'\sim\mathbf{M}}\left[\sigma(\mathcal{F}_\omega(z'))\right], \end{aligned} \tag{1}$$

where $\sigma(t) = \log(1 + e^t)$, and the discriminator function $\mathcal{F}_\omega$ [8] is modeled by a neural network with parameters ω.

Table I: Recommended final layer activation functions and their conjugate functions. This table comes from [8].

Name	Output activation g_f	$\mathbf{dom}_{g^\star}$	Conjugate $g^\star(t)$
Kullback-Leibler (KL)	v	$\mathbb{R}$	$\exp(t-1)$
Reverse KL	$-\exp(-v)$	$\mathbb{R}_-$	-1-log$(-t)$
Pearson χ^2	v	$\mathbb{R}$	$\frac{1}{4}t^2+t$
Square Hellinger	$1-\exp(-v)$	$t<1$	$\frac{t}{1-t}$
Jensen-Shannon	$\log(2)-\log(1+\exp(-v))$	$t<\log(2)$	$-\log(2-\exp(t))$

Proof. According to the variational estimation of f-divergences [7], we have

$$\begin{aligned}\mathcal{D}_f(\mathbf{P}||\mathbf{Q}) &= \int q(x) \sup_{t\in\mathrm{dom}_{g^*}} t\frac{p(x)}{q(x)} - g^*(t)\mathrm{d}x \\ &\geq \sup_{\mathcal{V}\in F}\left(\int p(x)\mathcal{V}(x)\mathrm{d}x - \int q(x)g^*(\mathcal{V}(x))\mathrm{d}x\right) \\ &= \sup_{\mathcal{V}\in F}\left(\mathbb{E}_{x\sim\mathbf{P}}[\mathcal{V}(x)] - \mathbb{E}_{x\sim\mathbf{Q}}[g^*(\mathcal{V}(x))]\right)\end{aligned} \tag{2}$$

where the function g^* is a convex conjugate function [2,8] of a convex, lower-semicontinuous function. The function g^* is defined as

$$g^*(t) = \sup_{u\in\mathrm{dom}_f}\{ut - f(u)\} \tag{3}$$

We parameterize $\mathcal{V}$ using a neural network with parameters w and write $\mathcal{V}_\omega$. We assume the the form of the function $\mathcal{V}_\omega = g_f(\mathcal{F}_\omega(x))$. Given two probability distributions $\mathbf{J} = p(x,y)$ and $\mathbf{M} = p(x)p(y)$, their f-divergence satisfies:

$$\mathcal{D}_f(\mathbf{J}||\mathbf{M}) = \sup_{\mathcal{F}_\omega}(\mathbb{E}_{z\sim\mathbf{J}}[g_f(\mathcal{F}_\omega(z))] - \mathbb{E}_{z'\sim\mathbf{M}}[g^*(g_f(\mathcal{F}_\omega(z')))]) \tag{4}$$

where g_f is an activation function specific to the f-divergence used. Table I provides the commonly used g_f and the convex conjugate function g^*. According to this table, for the JSD based divergence, we have $g_f(x) = \log(2) - \log(1+\exp(-x))$ and $g^*(x) = -\log(2-\exp(x))$. By substituting them into Eq. (4), we have

$$\begin{aligned}\mathbb{E}_{z\sim\mathbf{J}}\left[g_f(\mathcal{F}_\omega(z))\right] &= \mathbb{E}\left[\log 2 - \log(1+\exp(-\mathcal{F}_\omega(z)))\right] \\ &= \mathbb{E}_{z\sim\mathbf{J}}\left[\log 2 - \sigma(-\mathcal{F}_\omega(z))\right]\end{aligned} \tag{5}$$

$$\begin{aligned}
&\mathbb{E}_{z'\sim\mathbf{M}}\left[g^{*}(g_f(\mathcal{F}_\omega(z')))\right]\\
&\qquad=\mathbb{E}_{z'\sim\mathbf{M}}\left[-\log(2-\exp^{\log2-\log(1+\exp(-\mathcal{F}_\omega(z')))})\right]\\
&\qquad=\mathbb{E}_{z'\sim\mathbf{M}}\left[-\log(2-2(1+\exp(-\mathcal{F}_\omega(z'))^{-1}))\right]\\
&\qquad=\mathbb{E}_{z'\sim\mathbf{M}}\left[-\log(2\frac{\exp(-\mathcal{F}_\omega(z'))}{1+\exp(-\mathcal{F}_\omega(\mathrm{z}'))})\right]\\
&\qquad=\mathbb{E}_{z'\sim\mathbf{M}}\left[-\log\frac{2\exp(-\mathcal{F}_\omega(z'))\exp(\mathcal{F}_\omega(z'))}{\exp(\mathcal{F}_\omega(z'))+\exp(-\mathcal{F}_\omega(z'))\exp(\mathcal{F}_\omega(z'))}\right]\\
&\qquad=\mathbb{E}_{z'\sim\mathbf{M}}\left[-\log(\frac{2}{\exp(\mathcal{F}_\omega(z'))+1})\right]\\
&\qquad=\mathbb{E}_{z'\sim\mathbf{M}}\left[-(\log2-\log(\exp(\mathcal{F}_\omega(z'))+1))\right]\\
&\qquad=\mathbb{E}_{z'\sim\mathbf{M}}\left[-\log2+\sigma(\mathcal{F}_\omega(z'))\right]
\end{aligned}\tag{6}$$

Combining Eq. (5) and Eq. (6), we can rewrite Eq. (4) as a JSD-divergence based form:

$$\begin{aligned}
\mathcal{D}_{\text{JSD}}(\mathbf{J}||\mathbf{M})=&\sup_{\mathcal{F}_\omega}(\mathbb{E}_{z\sim\mathbf{J}}\left[\log2\right]+\mathbb{E}_{z\sim\mathbf{J}}\left[-\sigma(-\mathcal{F}_\omega(z))\right]\\
&+\mathbb{E}_{z'\sim\mathbf{M}}\left[\log2\right]-\mathbb{E}_{z'\sim\mathbf{M}}\left[\sigma(\mathcal{F}_\omega(z'))\right])\\
\geq&\mathbb{E}_{z\sim\mathbf{J}}\left[-\sigma(-\mathcal{F}_\omega(z))\right]-\mathbb{E}_{z'\sim\mathbf{M}}\left[\sigma(\mathcal{F}_\omega(z'))\right]
\end{aligned}\tag{7}$$

C Self-Supervised Learning for Baseline DiR

The learning objective of the DiRNet is defined as:

$$\mathcal{L}^{1}_{view}=I(\mathbf{r}^{(0)};\mathbf{x}_1|\mathbf{x}_2)-I(\mathbf{r}^{(0)};\mathbf{x}_1)\tag{8}$$

$$\mathcal{L}^{2}_{view}=I(\mathbf{r}^{(0)};\mathbf{x}_2|\mathbf{x}_1)-I(\mathbf{r}^{(0)};\mathbf{x}_2)\tag{9}$$

The average of the two functions Eq.(8) and Eq.(9) is

$$\mathcal{L}_{average}=-\frac{I(\mathbf{r}^{(0)};\mathbf{x}_1)+I(\mathbf{r}^{(0)};\mathbf{x}_2))}{2}+\frac{I(\mathbf{r}^{(0)};\mathbf{x}_1|\mathbf{x}_2)+I(\mathbf{r}^{(0)};\mathbf{x}_2|\mathbf{x}_1)}{2}\tag{10}$$

Following [4], the conditional mutual information $I(\mathbf{r}^{(0)};\mathbf{x}_1|\mathbf{x}_2)$ satisfies the following proposition:

Proposition 2. *Given two degraded observations $\mathbf{x}_1$ and $\mathbf{x}_2$ from the same degradation-free image, and the baseline DiR representation $\mathbf{r}^{(0)}$, the conditional mutual information satisfies*

$$I(\mathbf{r}^{(0)};\mathbf{x}_1|\mathbf{x}_2)\leq\mathbb{E}_{p(\mathbf{x}_1,\mathbf{x}_2)}[\mathcal{D}_{\text{KL}}(p(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)||p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_2))]\tag{11}$$

Proof. As the two-view observations $\mathbf{x}_1$ and $\mathbf{x}_2$ can produce a shared degradation-independent representation $\mathbf{r}^{(0)}$, the $I(\mathbf{r}^{(0)};\mathbf{x}_1|\mathbf{x}_2)$ can be expressed as:

$$\begin{aligned}
&I(\mathbf{r}^{(0)};\mathbf{x}_1|\mathbf{x}_2)=\mathbb{E}_{p(\mathbf{x}_1,\mathbf{x}_2)}\mathbb{E}_{p(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)}\left[\log\frac{p(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)}{p(\mathbf{r}^{(0)}|\mathbf{x}_2)}\right]\\
&=\mathbb{E}_{p(\mathbf{x}_1,\mathbf{x}_2)}\mathbb{E}_{p(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)}\left[\log\frac{p(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_2)}{p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_2)p(\mathbf{r}^{(0)}|\mathbf{x}_2)}\right]\\
&=\mathbb{E}_{p(\mathbf{x}_1,\mathbf{x}_2)}[\mathcal{D}_{\mathrm{KL}}(p(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)||p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_2))]+\mathbb{E}_{p(\mathbf{r}^{(0)},\mathbf{x}_1,\mathbf{x}_2)}\left[\log\frac{p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_2)}{p(\mathbf{r}^{(0)}|\mathbf{x}_2)}\right]\\
&=\mathbb{E}_{p(\mathbf{x}_1,\mathbf{x}_2)}[\mathcal{D}_{\mathrm{KL}}(p(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)||p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_2))]-\mathbb{E}_{p(\mathbf{x}_2)}[\mathcal{D}_{\mathrm{KL}}(p(\mathbf{r}^{(0)}|\mathbf{x}_2)||p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_2))]\\
&\leq\mathbb{E}_{p(\mathbf{x}_1,\mathbf{x}_2)}[\mathcal{D}_{\mathrm{KL}}(p(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)||p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_2))]
\end{aligned}\tag{12}$$

where the above bound holds tight when $p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)$ coincides with $p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_1)$. Similarly,

$$I(\mathbf{r}^{(0)};\mathbf{x}_2|\mathbf{x}_1)\leq\mathbb{E}_{p(\mathbf{x}_1,\mathbf{x}_2)}[\mathcal{D}_{\mathrm{KL}}(p(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)||p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_1))]$$

As the DiRNet used to extract $\mathbf{r}^{(0)}$ is shared by the two-view observations, we have $p(\cdot)=p_\varphi(\cdot)$. Therefore, the upper bound of Eq. (10) is:

$$\mathcal{L}_{average}\leq-\frac{I(\mathbf{r}^{(0)};\mathbf{x}_1)+I(\mathbf{r}^{(0)};\mathbf{x}_2))}{2}+\lambda\mathbb{E}_{p(\mathbf{x}_1,\mathbf{x}_2)}\left[\mathcal{D}_{\mathrm{AKL}}\right]\tag{13}$$

where λ is the trade-off parameter and

$$\mathcal{D}_{\mathrm{AKL}}=\frac{1}{2}\Big(\mathcal{D}_{\mathrm{KL}}(p(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)\,||\,p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_1))+\mathcal{D}_{\mathrm{KL}}(p(\mathbf{r}^{(0)}|\mathbf{x}_1,\mathbf{x}_2)\,||\,p_\varphi(\mathbf{r}^{(0)}|\mathbf{x}_2))\Big)\tag{14}$$

Finally, the loss for baseline DiR learning is

$$\begin{aligned}
\mathcal{L}_0=&-\frac{I(\mathbf{r}^{(0)};\mathbf{x}_1)+I(\mathbf{r}^{(0)};\mathbf{x}_2)}{2}\\
&+\lambda\mathbb{E}_{p(\mathbf{x}_1,\mathbf{x}_2)}\left[\mathcal{D}_{\mathrm{AKL}}\right]\\
&+\beta\mathcal{D}_{\mathrm{KL}}\left[p(\mathbf{r}^{(0)})\,||\,\mathcal{N}(0,\mathbf{I})\right].
\end{aligned}\tag{15}$$

D Network Structure

In this section, we depict the detailed network structures used in our experiments. Table II depicts the detailed network structure for objection detection, image segmentation, and image restoration experiments.

Module	Layer		Parameter
DiRNet (DfRNet)	Conv		$k = 5, s = 1\ , p = 2, c = 64$
	Conv $\times 3$		$k = 4, s = 2, p = 1, c = 64$
	ResBlock $\times 2$		$k = 3, s = 1, p = 1, c = 64$
$\mathcal{A}_1$	3×3 Conv $\times 4$		$k = 3, s = 3, p = 1, c = 64$
$\mathcal{A}_2$	G-Conv $\times 2$	3×3 Conv	$k = 3, s = 3, p = 1, c = 128$
		3×3 Conv	$k = 3, s = 1, p = 1, c = 256$
		AvePooling	$256 \times 1 \times 1$
		FC	$c = 256 \times 4608\ (512 \times 3 \times 3)$
		Reshape	$256 \times 512 \times 3 \times 3$
$\mathcal{A}_3$	3×3 Conv $\times 4$		$k = 3, s = 3, p = 1, c = 64$
Decoder D_*	ResBlock $\times 2$		$k = 3, s = 1, p = 1, c = 64$
	TranConv $\times 3$		$k = 4, s = 2, p = 1, c = 64$
	Conv		$k = 5, s = 1, p = 2, c = 3$

Table II: Detailed parameters of the network structure for the objection detection, image segmentation, and image restoration experiments. The TranConv denotes the transposed convolution. In the parameter column, k, s, p and c denote the kernel size, stride, padding, and the channel of the output feature, respectively. In the table, the 1×1 convolution layers in the $\mathcal{A}_2$ module are omitted where the parameters are set $s = 1, p = 0$.

E Experiments on the Classification Task

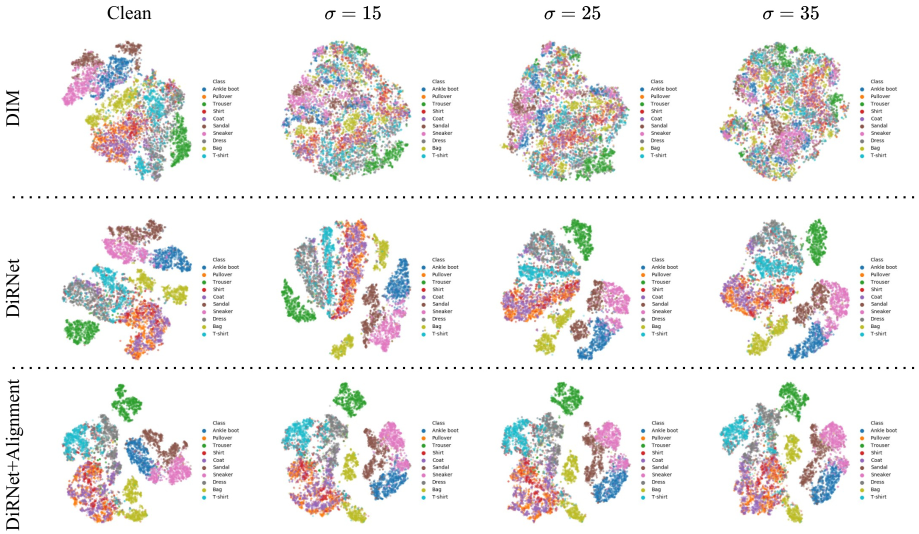

Fig. I: Visualization of the representations by t-SNE on noisy Fashion-MNIST test sets at different noise levels (standard derivations $\sigma = 15, 25, 35$).

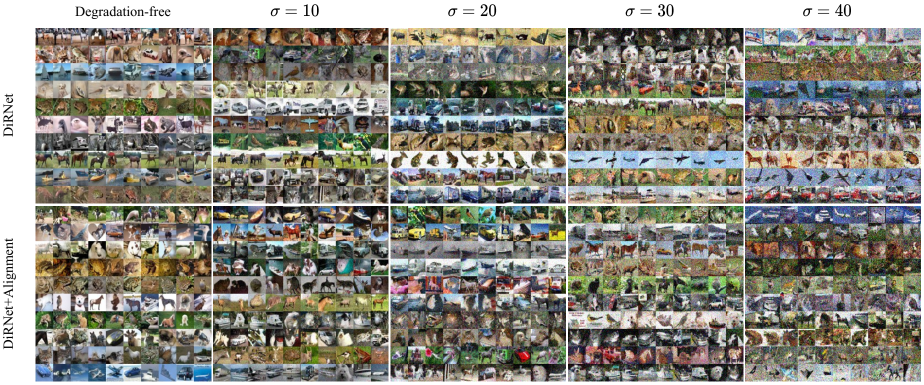

Fig. II: Visualization of the KNN classification results of noisy test images (Fashion-MNIST) using our DiR representations.

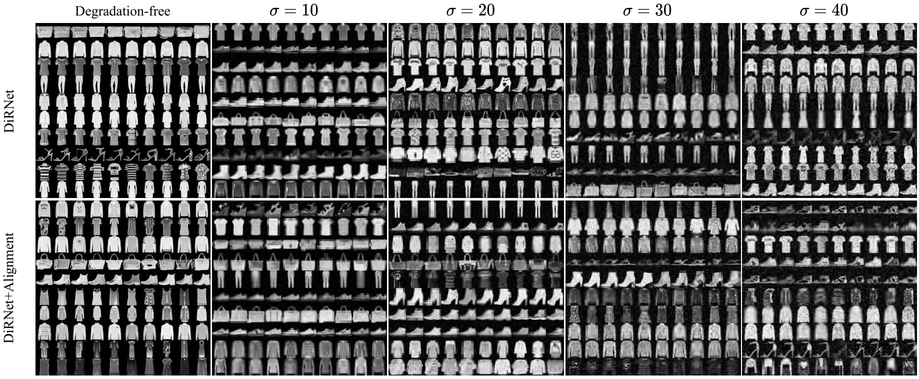

Fig. III: Visualization of the KNN classification results of noisy test images (CIFAR-10) using our DiR representations.

In this section, we experiment with our method on the classification task, where the test images are degraded by random noise. In these experiments, we replace the convolution layers in the DiRNet, and DfRNet with FC layers and generate 64-dimensional representations for the classification experiments. We add Gaussian noise to the degradation-free images from the Fashion-MNIST dataset and CIFAR-10 dataset to generate ISP-degraded images for evaluation. A KNN classifier is used to test the classification task with our DiR representations. Fig. II and Fig. III show the qualitative results of 10 nearest neighbors.

In Fig. I, we visualize the resulting representations of different classes by t-SNE [5]. The Deep InfoMax (DIM) [3] method is adopted to compare, which achieves unsupervised learning of representations by maximizing mutual information between an input and the output of a deep neural network encoder. We can observe that the results of clustering with the proposed DiRNet representations can perform robustly across different noise levels, while the straightforward mutual information maximization method DIM fails in these cases.

F More Qualitative Results on the Image Restoration Task

In this section, we present more results on the image restoration task. Fig IV, Fig V, and Fig VI show the restored results on real-world noisy images from the PolyU [11], Nam [6], and DND [9] datasets, respectively.

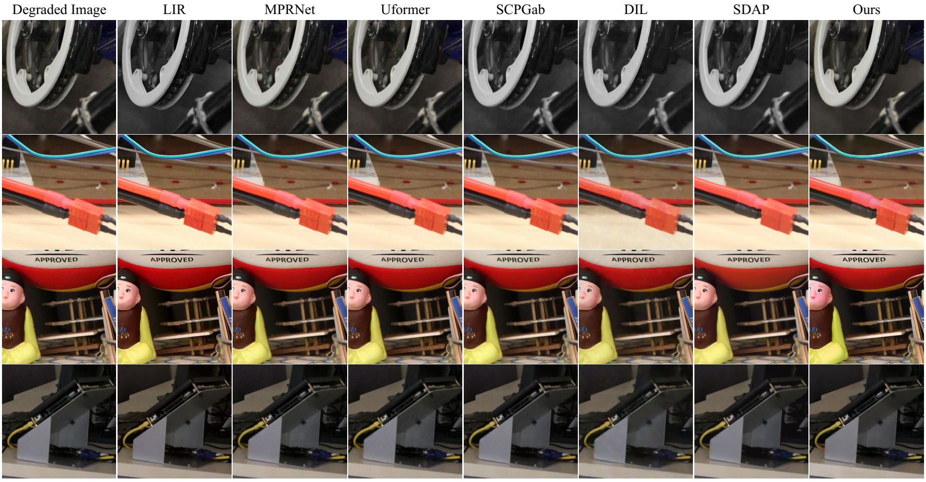

Fig. IV: Qualitative comparison on real-world ISP-degraded images of PolyU dataset [11]. Please zoom in for a better view.

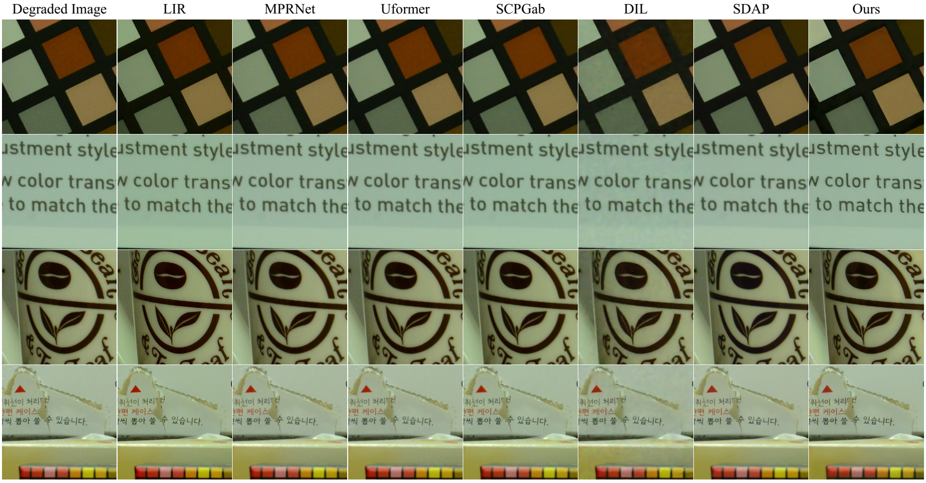

Fig. V: Qualitative comparison on real-world ISP-degraded images of Nam dataset [6]. Please zoom in for a better view.

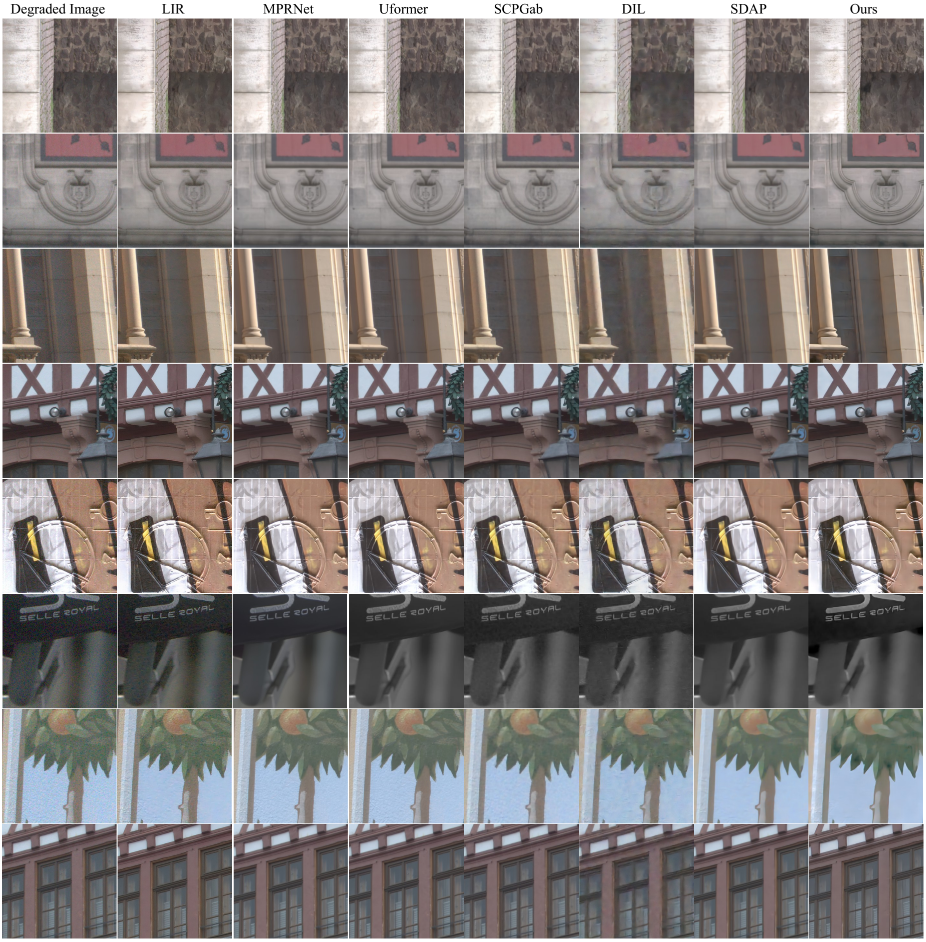

Fig. VI: Qualitative comparison on real-world ISP-degraded images of DND dataset [9]. Please zoom in for a better view.

G Qualitative Results of Ablation Study

In this section, we provide the more qualitative comparison results of different configurations in DiR learning. The results are shown in Fig VII. We can observe that the baseline DiR outperforms the naive auto-encoder learning. The alignment network $\mathcal{A}$ and the pilot DfR $\vec{\mathbf{r}}$ collectively contribute to improving the performance of the baseline DiR.

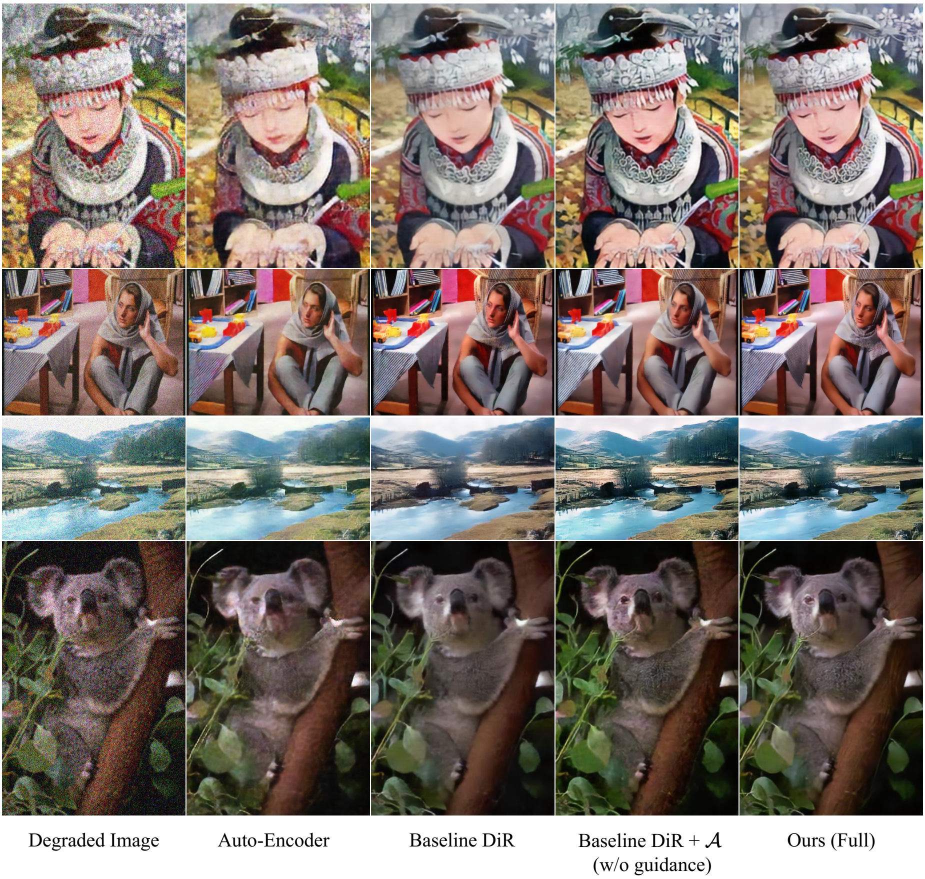

Fig. VII: Visual comparison of different configurations of the DiR learning. Please zoom in for a better view.

References

1. Brown, M.S., Kim, S.: Understanding the in-camera image processing pipeline for computer vision. In: IEEE International Conference on Computer Vision (ICCV)-Tutorial. vol. 3, pp. 1–354 (2019) 1

2. Hiriart-Urruty, J.B., Lemaréchal, C.: Fundamentals of convex analysis. Springer Science & Business Media (2004) 2
3. Hjelm, R.D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., Bengio, Y.: Learning deep representations by mutual information estimation and maximization. In: International Conference on Learning Representations (2018) 7
4. Hwang, H., Kim, G.H., Hong, S., Kim, K.E.: Variational interaction information maximization for cross-domain disentanglement. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 22479–22491. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper/2020/file/fe663a72b27bdc613873fbbb512f6f67-Paper.pdf 3
5. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008) 7
6. Nam, S., Hwang, Y., Matsushita, Y., Kim, S.J.: A holistic approach to cross-channel image noise modeling and its application to image denoising. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1683–1691 (2016) 7, 8
7. Nguyen, X., Wainwright, M.J., Jordan, M.I.: Estimating divergence functionals and the likelihood ratio by convex risk minimization. IEEE Transactions on Information Theory **56**(11), 5847–5861 (2010). https://doi.org/10.1109/TIT.2010.2068870 2
8. Nowozin, S., Cseke, B., Tomioka, R.: f-gan: Training generative neural samplers using variational divergence minimization. In: Proceedings of the 30th International Conference on Neural Information Processing Systems. pp. 271–279 (2016) 1, 2
9. Plötz, T., Roth, S.: Benchmarking denoising algorithms with real photographs (2017) 7, 9
10. Qin, H., Han, L., Xiong, W., Wang, J., Ma, W., Li, B., Hu, W.: Learning to exploit the sequence-specific prior knowledge for image processing pipelines optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22314–22323 (2023) 1
11. Xu, J., Li, H., Liang, Z., Zhang, D., Zhang, L.: Real-world noisy image denoising: A new benchmark (2018) 7, 8