Supplementary Material of UniRestore: Unified Perceptual and Task-Oriented Image Restoration Model Using Diffusion Prior

I-Hsiang Chen[1,*] Wei-Ting Chen[1,2,4,*] Yu-Wei Liu[1] Yuan-Chun Chiang[1]
Sy-Yen Kuo[1,3] Ming-Hsuan Yang[4,5]

[1]National Taiwan University [2]Microsoft [3]Chang Gung University
[4]UC Merced [5]Google Research

1. Dataset Details

In the Unified Image Restoration Task, we address two categories of tasks: Perceptual Image Restoration (PIR) and Task-oriented Image Restoration (TIR). TIR involves three downstream tasks: image classification, semantic segmentation, and object detection. Specifically, we select representative benchmarks in each task for training and validation. For PIR, we use the DIV2K [1], Flickr2K [12], and OST [33] datasets. For image classification, 80,000 images are randomly sampled from 1,000 classes in ImageNet [4]. For semantic segmentation, the Cityscapes [26] dataset is employed, while object detection uses 69,242 images randomly selected from the COCO [14] training set. The sampling dataset is indicated by the suffix "-s".

To simulate diverse and challenging visual scenarios, we augment these datasets with 15 types of synthetic degradations generated by the method [8]. These degradations include fog, snow, frost, exposure, contrast, elastic transform, pixelation, JPEG compression, Gaussian noise, impulse noise, shot noise, motion blur, defocus blur, glass blur, and zoom blur. Additionally, we retain clean images in the training set to enhance the model's ability to restore high-quality images.

As shown in Figure 1, these synthetic degradations effectively mimic a wide range of challenging conditions, where each degradation type having five distinct levels to improve UniRestore's robustness in handling diverse image qualities. Consequently, we not only evaluate the model in a synthetic degradation dataset, but also test it on various existing benchmarks and real-world datasets, demonstrating the generalizability of the proposed model. Detailed attributes of the datasets are provided in Table 1, 2, 3, 4.

* Indicates equal contribution.

Datasets	Domain of Data	Num of images	Degradation types	Dist. types
DIV2K [1]	train	800	degradation	synthetic
Flickr2K [12]	train	2,650	degradation	synthetic
OST [33]	train	10,324	degradation	synthetic
DIV2K [1]	testing	100	degradation	synthetic
RESIDE-SOTS [11]	testing	1,000	fog	synthetic
Rain100L [34]	testing	100	rain	synthetic
UHDSnow [32]	testing	200	snow	synthetic
GoPro [23]	testing	100	blur	synthetic
Urban100 [9]	testing	100	noise	synthetic
BSD68 [22]	testing	68	noise	synthetic
CBSD68 [22]	testing	68	noise	synthetic
KodaK [22]	testing	24	noise	synthetic
McMaster [22]	testing	18	noise	synthetic
Set12 [22]	testing	12	noise	synthetic
UDC [35]	testing	60	unknown	realistic
Practical [34]	testing	15	rain	realistic
Snow100k-real [17]	testing	200	snow	realistic
RESIDE-Uann [11]	testing	4,809	fog	realistic

Table 1. **Sizes and characteristics of PIR datasets.**

Datasets	Domain of Data	Num of Images	Num of Categories	Degradation types	Dist. types
ImageNet-s [4]	train	80,000	1,000	degradation	synthetic
ImageNet-s [4]	testing	20,000	1,000	degradation	synthetic
CUB [31]	testing	5,794	200	degradation	synthetic
Caltech-256 [6]	testing	30,607	256	degradation	synthetic

Table 2. **Sizes and characteristics of image classification datasets.**

Datasets	Domain of Data	Num of images	Degradation types	Dist. types
Cityscapes [26]	train	2,975	degradation	synthetic
Cityscapes [26]	testing	500	degradation	synthetic
FoggyCityscapes [26]	testing	500	fog1, fog2, fog3	synthetic
ACDC [27]	testing	500	foggy, rain, snow, night	realistic

Table 3. **Sizes and characteristics of semantic segmentation datasets.**

Datasets	Domain of Data	Num of images	Num of Categories	Degradation types	Dist. types
COCO-s [14]	train	69,242	80	degradation	synthetic
COCO [14]	testing	2,935	80	degradation	synthetic
RTTS [11]	testing	4,321	5	fog	realistic

Table 4. **Sizes and characteristics of object detection datasets.**

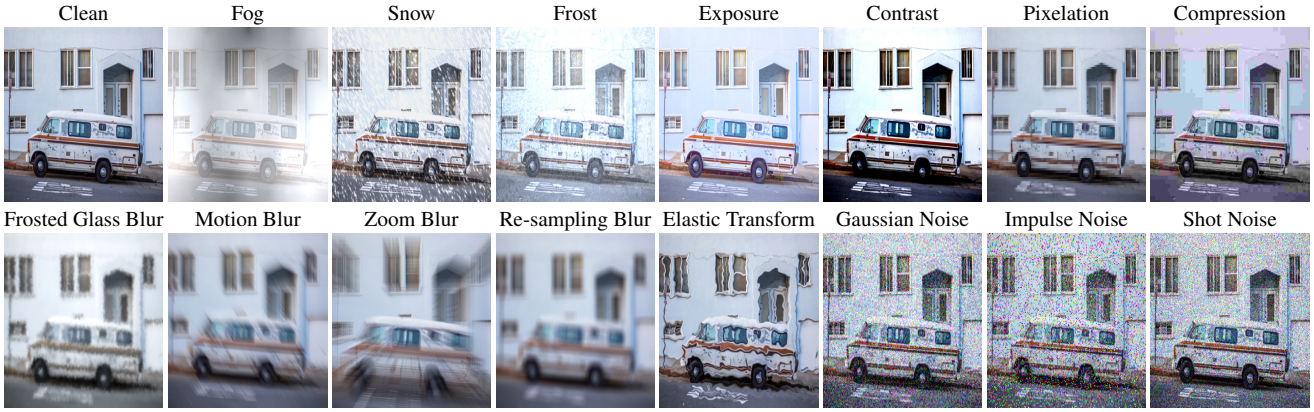

Figure 1. **Illustrative samples of images with synthetic degradations from the DIV2K [1] dataset.**

Module	PIR PSNR ↑	Cls ACC ↑	Seg mIoU ↑
UniRestore w/o CFRM	21.43	63.10	55.48
UniRestore w/o Group Channel Attention	22.93	66.80	61.28
UniRestore	**24.32**	**71.65**	**66.05**

Table 5. **Comparative analysis of different CFRM variants.**

2. More Experiments

2.1. Investigation of CFRM

To verify the effectiveness of the components in Complementary Feature Restoration Module (CFRM), we designed three different variants for analysis: (i) UniRestore w/o CFRM: using the vanilla encoder features without any restoration; (ii) UniRestore w/o Group Channel Attention: employing only the feature enhancement module for feature recovery; and (iii) UniRestore: incorporating all modules. The results are shown in Table 5, demonstrating that UniRestore with all modules outperforms the other two variants, highlighting that both the feature enhancement module and group channel attention play crucial roles in effectively restoring clear features.

2.2. Extendability Evaluation

To further analyze the extendability of UniRestore, we expanded on the experiments in Section. 5.4 of the main paper. Specifically, UniRestore leverages a pre-trained model and optimizes a new learnable prompt using object detection loss. For other baselines, updating jointly across PIR, image classification, semantic segmentation, and object detection to prevent catastrophic forgetting, denoted by the suffix "**", where the original baselines trained only on PIR, image classification, and semantic segmentation, indicated by the suffix "*". We employed RetinaNet [13] as the recognition model, sampling training data from the COCO [14] training set and evaluating on both the intra-domain COCO [14] validation set with the synthetic degradations and the inter-domain RTTS [11] dataset.

Methods	PIR PSNR ↑	Cls ACC ↑	Seg mIoU ↑	Obj-intra mAP ↑	Obj-inter mAP ↑
LQ	-	*51.75*	*40.36*	*16.28*	*45.63*
DIP* [16]	18.62	59.80	51.81	-	-
DIP** [16]	18.17	57.90	50.76	22.31	54.29
URIE* [30]	17.98	65.20	50.56	-	-
URIE** [30]	17.35	64.25	49.23	21.28	49.23
NAFNet* [2]	19.81	57.65	51.91	-	-
NAFNet** [2]	19.23	56.10	50.24	17.64	48.83
PromptIR* [25]	21.94	64.05	54.67	-	-
PromptIR** [25]	21.63	62.95	53.16	21.39	50.61
UniRestore	**24.32**	**71.65**	**66.05**	**29.53**	**58.06**

Table 6. **Extendability performance analysis for object detection.** When adding a new downstream task, existing methods experience a decline in performance on previously trained tasks (i.e., PIR, classification, and semantic segmentation). In contrast, UniRestore only requires updating the prompt for the new task while keeping all other parameters fixed, ensuring that the performance on previously trained tasks remains unaffected.

Training Strategy	COCO	RTTS	Finetuning Parameters
Tuning Prompt Only	29.53	58.06	< 1k
Full Fine-tuning	**30.14**	**58.51**	≈ 21 million

Table 7. **Computational Overhead for Extended Tasks.**

In Table 6, we demonstrate that UniRestore achieves best performance on both intra- and inter-domain object detection datasets (denoted as Obj-intra and Obj-inter). Furthermore, while existing methods suffer from overall performance degradation as tasks increase, UniRestore maintains stable results on existing tasks. We further compare object detection training by adding new prompts versus starting from scratch. Table 7 shows that full fine-tuning slightly improves performance for new tasks but requires significantly more parameters. In contrast, UniRestore only requires the addition of a task-specific prompt with its corresponding objectives and data, while keeping the weights

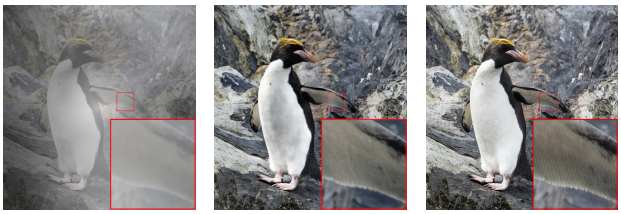

Figure 2. **Visual comparison of using diffusion prior.**

Method	PIR PSNR ↑	Cls ACC ↑	Seg mIoU ↑
UniRestore w/o Diffusion Prior	23.52	69.25	64.93
UniRestore	**24.32**	**71.65**	**66.05**

Table 8. **Effectiveness of diffusion prior.**

and prompts for previously trained tasks fixed. The design ensure that performance on the original tasks remains unaffected, highlighting its efficiency and scalability in extending to other downstream tasks through the Task Feature Adapter (TFA) module.

2.3. Investigation of Diffusion Prior

To further analyze the impact of the diffusion prior on UniRestore, we conducted a comparative study with a variant, "UniRestore w/o Diffusion Prior," which employs only the autoencoder without a denoising processing stage. Specifically, UniRestore utilizes the encoder-generated latent features as inputs to the Controller, which guides the denoising U-Net to produce $F_{\text{latent},0}$ as the decoder's input. To validate the effectiveness of the diffusion prior, we remove the Controller, denoising U-Net, and SC-Tuner from UniRestore, and instead directly use the encoder-generated latent features as inputs to the decoder for training. As shown in Table 8, using the diffusion prior achieves better performance across both PIR and TIR scenarios. Moreover, Figure 2 illustrates that while both methods effectively restore clean images, the integration of denoising processing enables the generation of richer texture details. This experiment demonstrates that UniRestore, with the diffusion prior, further improves the quality of perceptual and semantic feature representations, leading to superior restoration results.

2.4. Investigation of Task Prompt

To assess the effectiveness of task prompts, we perform analyses with different prompts. Figure 3 presents individual restorations of the same image using PIR, Cls, and Seg prompts. The PIR prompt results in a smoother image, whereas the Cls and Seg prompts emphasize more high-frequency details. As shown in Table 9, cross-testing different prompts on downstream tasks shows that the corresponding task prompt yields the best performance.

Figure 3. **Visual comparison of using different task prompts.**

Task Prompt	PIR PSNR ↑	Cls ACC ↑	Seg mIoU ↑
PIR	**24.32**	67.60	62.97
Cls	22.12	**71.65**	63.66
Seg	21.33	68.10	**66.05**

Table 9. **Effectiveness of task prompts.**

$(\beta_{PIR}, \beta_{Cls}, \beta_{Seg})$	Adjust TIR (0.1, 0.1, 10)	Adjust TIR (0.1, 10, 0.1)	Balance (10, 0.1, 0.1)	Adjust PIR (20, 0.1, 0.1)	Adjust PIR (50, 0.1, 0.1)
PIR (PSNR ↑)	22.87	23.19	24.32	24.78	**25.29**
Cls (ACC ↑)	69.85	**72.15**	71.65	70.75	69.10
Seg (mIoU ↑)	**67.19**	63.44	66.05	64.40	62.52

Table 10. **Analysis of the weighting coefficients β_{task}.**

Inputs	LQ	URIE	PromptIR	UniRestore
Caltech-256 (ACC ↑)	58.22	69.49	68.57	**74.28**

Table 11. **Zero-shot classification comparison on the whole Caltech-256 [6] dataset (unseen datasets with synthetic degradations).**

Methods	ResNet-50 [7]	ViT-B [5]	VGG16 [29]	Swin-T [18]	RVT [21]
LQ	51.75	67.65	42.80	69.25	73.10
UniRestore	**71.65**	**77.05**	**64.95**	**78.85**	**79.35**

Table 12. **Generalization performance across unseen classifiers.**

2.5. Justification for Weighting Coefficient β_{task}

β_{task} adjusts the objective function by scaling the task-specific loss. To investigate the effect of different β_{task} scales, we conducted a comprehensive study using various sets of coefficients. As shown in Table 10, increasing β enhance focus on specific tasks but may impair performance on others. Based on experimental results, we set β_{task} to $(10, 0.1, 0.1)$.

2.6. Generalization Performance Evaluation

Zero-shot Performance for Downstream Tasks To ensure diversity and efficiency within our computational constraints, we design the training data of TIR with equal sampling across categories. We then retrain UniRestore on the entire ImageNet [4] dataset and evaluate it on the CUB [31] dataset, where its performance improves from 53.70 to 53.94, confirming that UniRestore maintains strong generalization even with partial data. Additionally, we conduct zero-shot testing of UniRestore on the Caltech-256 [6] dataset with synthetic degradations, demonstrating

Method	fog		snow		frost		exposure		contrast		compression		impulse_noise		pixelate		motion_blur		clastic_transform		Average	
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
URIE [30]	14.15	0.6100	18.32	0.5843	15.26	0.6150	17.99	0.7632	18.86	0.6645	27.05	0.7980	17.45	0.2605	26.46	0.7734	22.82	0.6582	21.38	0.6987	19.97	0.6426
DIP [16]	12.74	0.6315	17.82	0.5648	17.33	0.7221	19.63	0.8468	14.78	0.6111	27.00	0.8039	15.68	0.2520	26.16	0.8039	21.37	0.6573	21.21	0.6915	19.37	0.6585
NAFNet [2]	13.63	0.7429	20.66	0.6973	13.99	0.6728	16.42	0.8089	17.71	0.8154	28.24	0.8241	30.52	0.8088	28.66	0.8439	25.53	0.7809	23.00	0.7015	21.84	0.7697
PromptIR [25]	18.02	0.8423	23.95	0.7624	18.55	0.7432	24.06	0.8684	25.98	0.9142	27.72	0.8267	29.05	0.7878	28.11	0.8461	26.64	0.8296	21.49	0.6414	24.36	0.8062
UniRestore	**21.36**	**0.8707**	**25.58**	**0.8385**	**19.37**	**0.7298**	**24.18**	**0.8734**	**27.82**	**0.9299**	**28.80**	**0.8523**	**31.80**	**0.8523**	**29.19**	**0.8606**	**27.56**	**0.8507**	**23.97**	**0.7152**	**25.96**	**0.8373**

Table 13. **Performance comparison of existing methods on the seen dataset which consists 15 distinct degradation on DIV2K [1] testing set for PIR task.**

Methods	PSNR ↑	SSIM ↑	LPIPS ↓	MUSIQ ↑	MANIQA ↑	CLIP-IQA ↑
URIE [30]	18.81	0.6406	0.3566	42.55	0.2276	0.4987
PromptIR [25]	24.85	0.8447	0.2383	58.67	0.3278	0.6032
DiffBIR [15]	24.16	0.8340	0.2359	53.95	0.3104	0.5767
UniRestore	**26.00**	**0.8668**	**0.1713**	**62.00**	**0.3513**	**0.6444**

Table 14. **Quantitative results of PIR on one DIV2K [1] and five unseen PIR datasets [9, 11, 22, 23, 32, 34] across various metrics.**

Methods	TOLED [35]			POLED [35]		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
DIP [16]	24.32	0.7907	0.3680	13.84	0.4831	0.7520
URIE [30]	24.78	0.7845	0.3699	14.23	0.4911	0.7204
NAFNet [2]	26.97	0.7916	0.3393	14.73	0.5522	0.7526
PromptIR [25]	25.98	0.7977	0.4219	13.07	0.5475	0.7352
UniRestore	**28.12**	**0.8146**	**0.3043**	**17.01**	**0.5563**	**0.7049**

Table 15. **Quantitative results of PIR on unseen under-display camera image restoration datasets (i.e., TOLED [35] and POLED [35]).**

Methods	Practical [34]		RESIDE-Unann [17]		Snow100k-real [11]	
	CLIP-IQA ↑	PAQ2PIQ ↑	CLIP-IQA ↑	PAQ2PIQ ↑	CLIP-IQA ↑	PAQ2PIQ ↑
DIP [16]	0.5995	70.69	0.4625	67.76	0.5201	68.48
URIE [30]	0.4310	68.27	0.3411	66.85	0.3464	66.88
NAFNet [2]	0.6016	70.75	0.4702	68.90	0.5503	68.88
PromptIR [25]	0.6376	70.99	0.5053	69.17	0.5886	69.77
UniRestore	**0.6706**	**71.71**	**0.5294**	**69.74**	**0.6061**	**70.81**

Table 16. **Quantitative results of PIR on unseen real-world datasets.**

improved performance over other baselines across various categories, as shown in Table 11.

Cross-Classifier Generalization UniRestore adopts ResNet-50 [7] (Cls) and DeepLabV3+ [3] (Seg) as recognition models during training. For testing, restored images are fed into ViT [5] (Cls) and RefineNet-lw [24] (Seg) without finetuning. Table 12 further compares additional classifiers as test backbones, demonstrating UniRestore's effectiveness in enhancing downstream models.

Effectiveness on Clean Image. To assess performance on clean inputs, we evaluate a ResNet-50 [7] model for classification, which improves from 72.80 to 74.10. For segmentation, using DeepLabV3+ [3], UniRestore achieves 74.82, closely matching the original DeepLabV3+ [3]'s 75.64. These improvements result from training with a mixture of 15 synthetic degradations and clean inputs, demonstrating the model's robustness.

2.7. Quantitative Evaluation for PIR

In the quantitative evaluation, we extend the evaluation of PIR on seen dataset which involves ten distinct degradation types from the DIV2K [1] testing set, as detailed in Table 13. The experimental results demonstrate that UniRestore achieves the highest overall performance, validating UniRestore's robustness and generalizability across diverse degradation scenarios. Table 14 further shows that UniRestore excels in both full-reference and no-reference IQA.

To further evaluate the generalization for real-world scenarios, we conduct zero-shot testing on several unseen real-world datasets. As shown in Table 15, it demonstrates the UniRestore's robustness in addressing real-world unknown tasks (i.e., under-display camera (UDC) image restoration [35]), without requiring any fine-tuning. The results indicate that UniRestore exhibits a better overall performance, demonstrating its capability to restore both perceptual and semantic features. Furthermore, we conduct a evaluation across three unseen real-world adverse weather datasets, as detailed in Table 16. The results show that UniRestore achieves competitive performance across two image quality assessment (IQA) metrics, highlighting its robustness and applicability in handling diverse real-world scenarios.

2.8. Qualitative Evaluation for PIR

As illustrated in Figure 4 and Figure 5, our qualitative evaluation provides a comprehensive comparison, highlighting the effectiveness of UniRestore across various types of degradation. UniRestore successfully removes attenuation caused by air particles such as fog, rain, and snow while preserving and reconstructing critical object information. In the motion blur and overexposure scenarios, UniRestore reconstructs fine-grain information, resulting in better fidelity and visually natural restoration images. Additionally, UniRestore effectively mitigates artifacts caused by compression, improving texture representation and overall image quality. These results highlight UniRestore's strength in integrating restored encoder features with the diffusion prior, leading to high-quality restoration performance.

Figure 4. **Qualitative Analysis of Perceptual Image Restoration:** A visual comparison across four existing benchmarks, including RESIDE-SOTS [11], Rain100L [34], UHDSnow [32] and BSD68 [22], highlighting the UniRestore's generalizability in unseens datasets.

3. Implementation Details

3.1. Training Details

UniRestore is established based on SD-turbo [28], with pre-trained weights frozen to preserve the diffusion prior. The controller adopts a lightweight U-Net encoder architecture combined with SC-Tuner modules [10] for effective control of the denoising process. The CFRM and TFA modules are integrated between each layer of the pre-trained autoencoder, with $M = 3$ pairs of modules, consisting of seven layers for both the encoder (4 original layers + 3 CFRM) and the decoder (4 original layers + 3 TFA). The number of groups in CFRM is 16, corresponding to 15 degradations plus one clean condition. The training pipeline consists of three stages: in the first stage, CFRM, the controller, and SC-Tuner are trained for feature restoration using paired degraded data, where the coefficients of CFRM loss is (λ_1, λ_2, λ_3) = $(0.1, 0.1, 0.01)$. In the second stage, the TFA is trained with three learnable prompt vectors for PIR, image classification, and semantic segmentation, where dimensions of each prompt is 512 and the coefficients of task loss are (β_{PIR}, β_{Seg}, β_{Cls}) = $(10, 0.1, 0.1)$. Subsequently, a new task prompt vector can be individually fine-tuned to introduce additional tasks, using the corresponding dataset and objective function for training. For the training data, images are resized and cropped to 512×512 with random horizontal flips. The network is optimized using AdamW [19] with a learning rate of 5×10^{-5} and a cosine annealing scheduler [20], and trained on 8 NVIDIA Tesla V100 GPUs (32 GB) with a batch size of 48. In terms of parameter statistics, the SD-turbo comprises 975 million non-trainable parameters. Additionally, we integrate a controller with 74.7 million trainable parameters. For the proposed components, the Complementary Feature Restoration Module (CFRM) and the Task Feature Adapter (TFA) include 26 million and 21 million trainable parameters, respectively.

3.2. Inference Details

During the inference stage, UniRestore enables task-specific image restoration by switching task prompts, allowing the restored outputs to be optimized for specific downstream applications. The model supports inputs of arbitrary resolution, automatically resizing images smaller than 512×512 to meet processing requirements. UniRestore requires only one denoising step to efficiently generate latent features while preserving high-quality results. The post-processing dependents on the downstream task, for image classification, restored images are normalized and resized to 224×224 to comply with recognition model requirements, while for tasks like semantic segmentation or object detec-

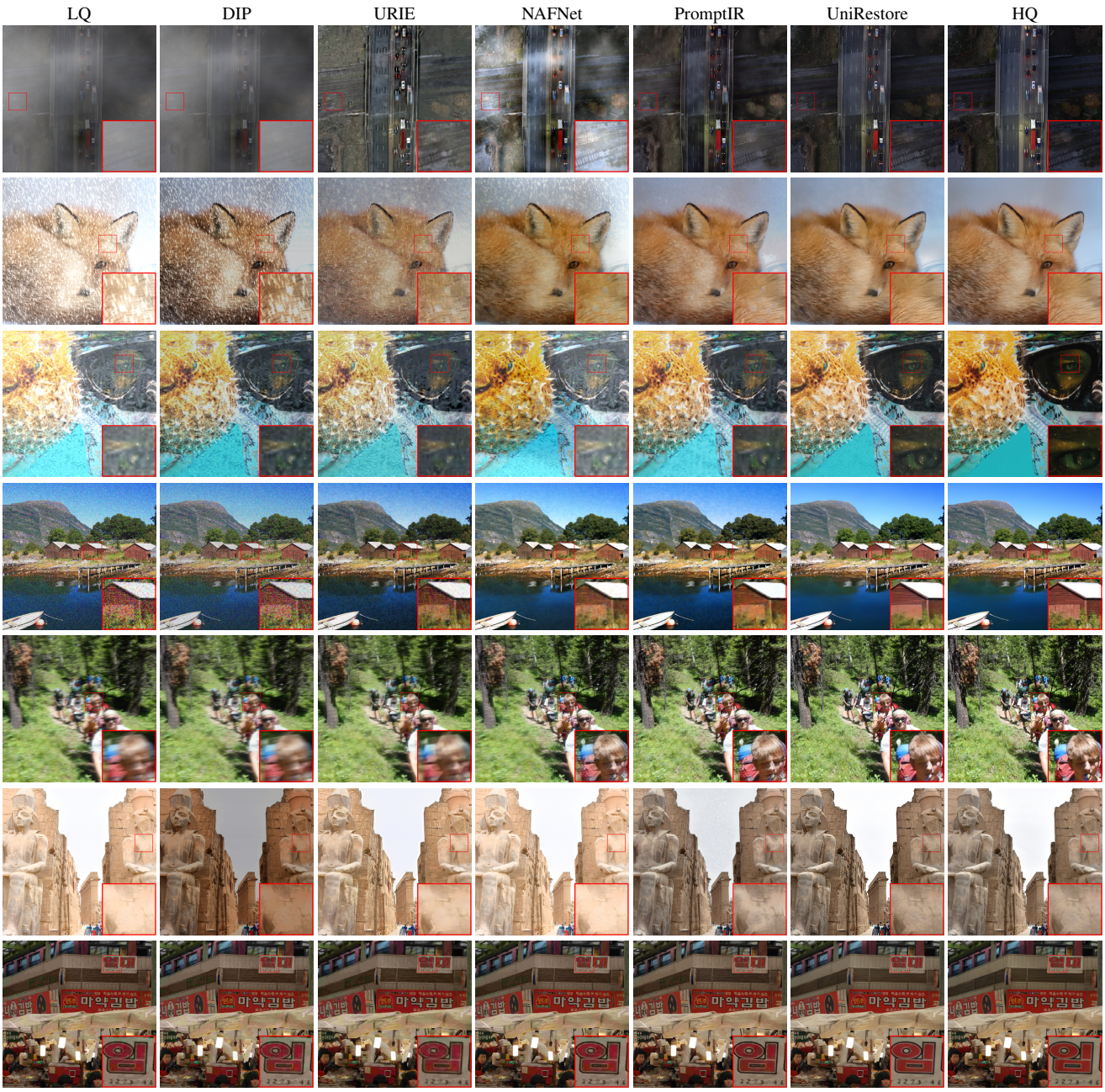

Figure 5. **Qualitative Analysis of Perceptual Image Restoration:** A visual comparison across various synthetic degradations on the DIV2K [1] testing set, including fog, snow, frost, noise, blur, exposure, and JPEG compression, highlights the robustness of UniRestore in restoring images affected by diverse degradation types.

tion, outputs are adjusted to match the original input resolution and format. This design ensures UniRestore's versatility and effectiveness in delivering both perceptual quality and task-specific performance across varied scenarios.

In terms of computational efficiency, we conducted experiments on the complete DIV2K [1] testing set. The inference was performed on an NVIDIA RTX 4090 GPU with input images resized to 512×512. The average inference time was 16 milliseconds for the encoding process, 113 milliseconds per step for denoising process, and 3.8 milliseconds per image for decoding, consuming 6.44G VRAM, 5.93 TFLOPs for each image.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *CVPRW*, 2017. 1, 2, 4, 6

[2] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *ECCV*, 2022. 2, 4

[3] Liang-Chieh Chen. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 4

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 1, 3

[5] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3, 4

[6] Gregory Griffin, Alex Holub, Pietro Perona, et al. Caltech-256 object category dataset. Technical report, Technical Report 7694, California Institute of Technology Pasadena, 2007. 1, 3

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *ICPR*, 2016. 3, 4

[8] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 1

[9] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *CVPR*, 2015. 1, 4

[10] Zeyinzi Jiang, Chaojie Mao, Yulin Pan, Zhen Han, and Jingfeng Zhang. Scedit: Efficient and controllable image diffusion generation via skip connection editing. In *CVPR*, 2024. 5

[11] Boyi Li, Wenqi Ren, Dengpan Fu, Dacheng Tao, Dan Feng, Wenjun Zeng, and Zhangyang Wang. Benchmarking single-image dehazing and beyond. *TIP*, 2018. 1, 2, 4, 5

[12] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *CVPRW*, 2017. 1

[13] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017. 2

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1, 2

[15] Xinqi Lin, Jingwen He, Ziyan Chen, Zhaoyang Lyu, Bo Dai, Fanghua Yu, Wanli Ouyang, Yu Qiao, and Chao Dong. Diffbir: Towards blind image restoration with generative diffusion prior. *arXiv preprint arXiv:2308.15070*, 2023. 4

[16] Wenyu Liu, Gaofeng Ren, Runsheng Yu, Shi Guo, Jianke Zhu, and Lei Zhang. Image-adaptive yolo for object detection in adverse weather conditions. In *AAAI*, 2022. 2, 4

[17] Yun-Fu Liu, Da-Wei Jaw, Shih-Chia Huang, and Jenq-Neng Hwang. Desnownet: Context-aware deep network for snow removal. *TIP*, 2018. 1, 4

[18] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 3

[19] I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[20] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 5

[21] Xiaofeng Mao, Gege Qi, Yuefeng Chen, Xiaodan Li, Ranjie Duan, Shaokai Ye, Yuan He, and Hui Xue. Towards robust vision transformer. In *CVPR*, 2022. 3

[22] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, 2001. 1, 4, 5

[23] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2017. 1, 4

[24] Vladimir Nekrasov, Chunhua Shen, and Ian Reid. Light-weight refinenet for real-time semantic segmentation. *arXiv preprint arXiv:1810.03272*, 2018. 4

[25] Vaishnav Potlapalli, Syed Waqas Zamir, Salman H Khan, and Fahad Shahbaz Khan. Promptir: Prompting for all-in-one image restoration. In *NIPS*, 2024. 2, 4

[26] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *IJCV*, 2018. 1

[27] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *CVPR*, 2021. 1

[28] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *ECCV*, pages 87–103. Springer, 2025. 5

[29] Karen Simonyan. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3

[30] Taeyoung Son, Juwon Kang, Namyup Kim, Sunghyun Cho, and Suha Kwak. Urie: Universal image enhancement for visual recognition in the wild. In *ECCV*, 2020. 2, 4

[31] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. Technical report, California Institute of Technology, 2011. 1, 3

[32] Liyan Wang, Cong Wang, Jinshan Pan, Weixiang Zhou, Xiaoran Sun, Wei Wang, and Zhixun Su. Ultra-high-definition restoration: New benchmarks and a dual interaction prior-driven solution. *arXiv preprint arXiv:2406.13607*, 2024. 1, 4, 5

[33] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *CVPR*, 2018. 1

[34] 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, 4, 5

[35] Yuqian Zhou, David Ren, Neil Emerton, Sehoon Lim, and Timothy Large. Image restoration for under-display camera. In *CVPR*, 2021. 1, 4