Diff-Plugin: Revitalizing Details for Diffusion-based Low-level Tasks

Yuhao Liu[1], Zhanghan Ke[1,†], Fang Liu[1,†], Nanxuan Zhao[2], Rynson W.H. Lau[1,†]
[1]City University of Hong Kong, [2]Adobe Research

Figure 1. Real-world applications of *Diff-Plugin* visualized across distinct single-type and one multi-type low-level vision tasks. *Diff-Plugin* allows users to selectively conduct interested low-level vision tasks via natural languages and can generate high-fidelity results.

Abstract

Diffusion models trained on large-scale datasets have achieved remarkable progress in image synthesis. However, due to the randomness in the diffusion process, they often struggle with handling diverse low-level tasks that require details preservation. To overcome this limitation, we present a new Diff-Plugin framework to enable a single pre-trained diffusion model to generate high-fidelity results across a variety of low-level tasks. Specifically, we first propose a lightweight Task-Plugin module with a dual branch design to provide task-specific priors, guiding the diffusion process in preserving image content. We then propose a Plugin-Selector that can automatically select different Task-Plugins based on the text instruction, allowing users to edit images by indicating multiple low-level tasks with natural language. We conduct extensive experiments on 8 low-level vision tasks. The results demonstrate the superiority of Diff-Plugin over existing methods, particularly in real-world scenarios. Our ablations further validate that Diff-Plugin is stable, schedulable, and supports robust training across different dataset sizes. Project page: https://yuhaoliu7456.github.io/Diff-Plugin

1. Introduction

Over the past two years, diffusion models [9, 21, 22, 61] have achieved unprecedented success in image generation and shown potential to become vision foundation models. Recently, many works [4, 25, 28, 31, 46, 91, 96] have demonstrated that diffusion models trained on large-scale text-to-image datasets can already understand various visual attributes and provide versatile visual representations for downstream tasks, *e.g.*, image classification [31], segmentation [25, 96], translation [46, 91], and editing [4, 28].

However, due to the inherent randomness in the diffusion process, existing diffusion models cannot maintain consistent contents to the input image and thus fail in handling low-level vision tasks. To this end, some methods [46, 63] propose to utilize input images as a prior via the DDIM Inversion [61] strategy when editing images, but they are unstable when the scenes are complex. Other methods [16, 52, 56, 71, 83] attempt to train new diffusion models on task-specific datasets from scratch, limiting them to solve only a single task.

In this work, we observe that an accurate text prompt describing the goal of the task can already instruct a pre-trained diffusion model to address many low-level tasks, but typically leads to obvious content distortion, as illustrated in Fig. 2. Our insight to this problem is that task-specific priors containing both guidance information of the task and spatial information of the input image can adequately guide

[†]Joint corresponding authors. This project is in part supported by a GRF grant (Grant No.: 11205620) from the Research Grants Council of Hong Kong.

pre-trained diffusion models to handle low-level tasks while maintaining high-fidelity content consistency. To harness this potential, we propose *Diff-Plugin*, the first framework enabling a pre-trained diffusion model, such as stable diffusion [54], to accommodate a variety of low-level tasks without compromising its original generative capability.

Diff-Plugin consists of two main components. First, it includes a lightweight Task-Plugin module to help extract task-specific priors. The Task-Plugin is bifurcated into the Task-Prompt Branch (TPB) and the Spatial Complement Branch (SCB). While TPB distills the task guidance prior, orienting the diffusion model towards the specified vision task and minimizing its reliance on complex textual descriptions, SCB leverages task-specific visual guidance from TPB to assist the spatial details capture and complement, enhancing the fidelity of the generated content. Second, to facilitate the use of multiple different Task-Plugins, *Diff-Plugin* includes a Plugin-Selector to allow users to choose their desired Task-Plugins through text inputs (visual illustrations are depicted in Fig. 1). To train the Plugin-Selector, we employ multi-task contrastive learning [49], using task-specific visual guidance as pseudo-labels. This enables the Plugin-Selector to align different visual embeddings with task-specific text inputs, thereby bolstering the robustness and user-friendliness of the Plugin-Selector.

To thoroughly evaluate our method, we conducted extensive experiments on eight diverse low-level vision tasks. Our results affirm that *Diff-Plugin* is not only stable across different tasks but also exhibits remarkable schedulability, facilitating text-driven multi-task applications. Additionally, *Diff-Plugin* showcases its scalability, adapting to various tasks across datasets of varying sizes, from less than 500 to over 50,000 samples, without affecting existing trained plugins. Finally, our results also show that the proposed framework outperforms existing diffusion-based methods both visually and quantitatively, and achieves competitive performances compared to regression-based methods.

Our key contributions are summarized as follows:

- We present *Diff-Plugin*, the first framework to enable a pre-trained diffusion model to perform various low-level tasks while maintaining the original generative abilities.
- We propose a Task-Plugin, a lightweight dual-branch module designed for injecting task-specific priors into the diffusion process, to enhance the fidelity of the results.
- We propose a Plugin-Selector to select the appropriate Task-Plugin based on the text provided by the user. This extends to a new application that can allow users to edit images via text instructions for low-level vision tasks.
- We conduct extensive experiments on eight tasks, demonstrating the competitive performances of *Diff-Plugin* over existing diffusion and regression-based methods.

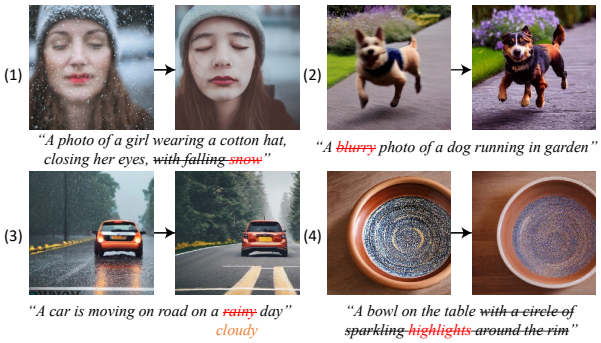


Figure 2. Stable Diffusion (SD) [54] results on four low-level vision tasks: desnowing, deblurring, deraining, and highlight removal. Each sub-figure illustrates a two-step process: First, we generate the left image using SD with a full-text description, where task-critical attributes are highlighted in red. Then, we remove unwanted attributes (indicated with ~~strikethrough~~), optionally add new attributes (denoted with orange word), and employ the *img2img* function in SD, using the left image as a condition to generate the edited image on the right. We observe that while SD can grasp rich attributes of various low-level tasks and create content consistent with descriptions, its inherent randomness often leads to content change in further editing. For instance, in sub-fig (1), besides addressing the primary task-related degradation (*e.g.*, snow), SD also alters unrelated content (*e.g.*, face profile).

2. Related Works

Diffusion models [60, 62] have been applied to image synthesis [9, 21, 22, 61] and achieved remarkable success. With extensive text-image data [59] and large-scale language models [49, 50], diffusion-based text-guided image synthesis [2, 42, 51, 54, 57] has become even more compelling. Leveraging the text-guided synthesis diffusion model, several approaches harness the generative prowess for text-driven editing. Zero-shot approaches [19, 46, 63] rely on a correct initial noise [61] and manipulate the attention map to edit specified content at precise locations. Tuning-based strategies strive to balance between image fidelity and generated diversity through optimized DDIM inversion [65], attention tuning [29], text-image coupling [28, 55, 93] and prompt tuning [10, 14, 39]. Conversely, InstructP2P [4, 89] generates paired data through latent diffusion [54] and prompt-to-prompt [19] for training and editing. However, the randomness in the diffusion process and the absence of task-specific priors render them infeasible for low-level vision tasks that require details preservation.

Conditional generative models use various external inputs to ensure output consistency with the conditions. Training-free methods [8, 76] can generate new contents at specified positions by manipulating attention layers, yet with limited condition types. Fine-tuning-based approaches inject additional guidance to the pre-trained diffusion models by train-

ing a new diffusion branch [40, 90, 94] or the whole model [1]. Despite the global structural consistency, these methods cannot ensure high-fidelity between output and input image details due to the randomness and generative nature.

Diffusion-based low-level methods can be grouped into zero-shot and training-based. The former can borrow generative priors from pre-trained denoising diffusion-based generative models [22] to solve linear [27, 70] and/or non-linear [7, 12] image restoration tasks, but often produce poor results on real-world data. The latter usually train or fine-tune an individual model for different tasks via task-dependent designs, such as super-resolution [58, 74], JPEG compression [56], deblurring [52, 73], face restoration [71, 95], low-light enhancement [24, 83, 92], and shadow removal [16]. Concurrent works, StableSR [66] and DiffBIR [34], use a learnable conditional diffusion branch with degraded or restored images to train diffusion models specifically for blind face restoration. In contrast, our framework enables one pre-trained diffusion model to handle a variety of low-level tasks by equipping it with lightweight task-specific plugins.

Multi-task models can learn complementary information across different tasks, *e.g.*, object detection and segmentation [18], rain detection and removal [80], adverse weather restoration [45, 82, 98] and blind image restoration [33, 47]. However, these methods can only handle the pre-defined tasks after training. Instead, our *Diff-Plugin* is flexible and can integrate new tasks through task-specific plugins, as our Task-Plugins are trained individually. Hence, when adding new low-level tasks to *Diff-Plugin*, we only need to add the pre-trained Task-Plugins to the framework, without the need to retrain the existing ones.

3. Methodologies

In this section, we first review the diffusion model formulations (Sec. 3.1). Then, we introduce our *Diff-Plugin* framework (Sec. 3.2), which developed from our newly proposed Task-Plugin (Sec. 3.3) and Plugin-Selector (Sec. 3.4).

3.1. Preliminaries

The diffusion model consists of a forward process and a reverse process. In the forward process, given a clean input image $\mathbf{x}_0$, the diffusion model progressively adds Gaussian noise to it to get noisy image $\mathbf{x}_t$ at time-step $t \in \{0, 1, ..., T\}$, as $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t$, where $\bar{\alpha}_t$ is the pre-defined scheduling variable and $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is the added noise. In the reverse process, the diffusion model performs iteratively remove noise from a standard Gaussian noise $\mathbf{x}_T$, and finally estimating a clean image $\mathbf{x}_0$. This is typically employed to train a noise prediction network $\boldsymbol{\epsilon}_\theta$, with supervision informed by the noise $\boldsymbol{\epsilon}_t$, as $\mathcal{L} = \mathbb{E}_{\mathbf{x}_0,t,\boldsymbol{\epsilon}\sim\mathcal{N}(0,1)}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\|_2^2\right]$.

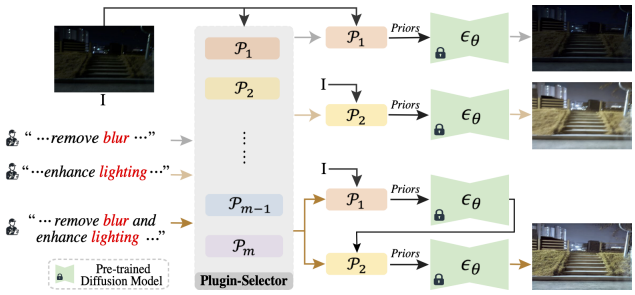


Figure 3. Schematic illustration of the *Diff-Plugin* framework. *Diff-Plugin* identifies appropriate Task-Plugin $\mathcal{P}$ based on the user prompts, extracts task-specific priors, and then injects them into the pre-trained diffusion model to generate the user-desired results.

3.2. Diff-Plugin

Our key observation is the inherent zero-shot capability of pre-trained diffusion models in performing low-level vision tasks, enabling them to generate diverse visual content without explicit task-specific training. However, this capability faces limitations in more nuanced task-specific editing. For example, in the desnowing task, while the model should ideally only remove snow and leave other contents unchanged, as shown in Fig. 2, the inherent randomness of the diffusion process often leads to unintended alterations in the scene beyond just snow removal. This inconsistency arises from the model's lack of task-specific priors, which are crucial for precise detail preservation in low-level vision tasks.

Inspired by modular extensions in NLP [75, 77] and GPT-4 [43], which utilize plug-and-play tools to enhance the capabilities of large language models for downstream tasks without compromising their core competencies, we introduce a novel framework, *Diff-Plugin*, based on a similar idea. This framework integrates several lightweight plugin modules, termed Task-Plugin, into the pre-trained diffusion models for various low-level tasks. Task-Plugins are crafted to provide essential task-specific priors, guiding the models to produce high-fidelity and task-consistent content. In addition, while diffusion models can generate content based on text instructions for targeted scenarios, they lack the ability to schedule Task-Plugins for different low-level tasks. Even existing conditional generation methods [48, 90] can only specify different generation tasks through input conditional images. Thus, to facilitate smooth text-driven task scheduling and enable the switching between different Task-Plugins for complex workflows, *Diff-Plugin* includes a Plugin-Selector to allow users to choose and schedule appropriate Task-Plugins with textual commands.

Fig. 3 depicts the *Diff-Plugin* framework. Given an image, users specify the task through a text prompt, either singular or multiple, and the Plugin-Selector identifies the appropriate Task-Plugin for it. The Task-Plugin then processes the image to extract the task-specific priors, guiding the pre-trained diffusion model to produce user-desired out-

comes. For more intricate tasks beyond the scope of a single plugin, *Diff-Plugin* breaks them down into sub-tasks with a predefined mapping table. Each sub-task is tackled by a designated Task-Plugin, showcasing the framework's capability to handle diverse and complex user requirements.

3.3. Task-Plugin

As illustrated in Fig. 4, our Task-Plugin module is composed of two branches: a Task-Prompt Branch (TPB) and a Spatial Complement Branch (SCB). The TPB is crucial for providing task-specific guidance to the pre-trained diffusion model, akin to using text prompts in text-conditional image synthesis [54]. We employ visual prompts, extracted via the pre-trained CLIP vision encoder [49], to direct the model's focus towards task-relevant patterns (*e.g.*, rain streaks for deraining and snow flakes for desnowing). Specifically, for an input image $\mathbf{I}$, the encoder $Enc_I(\cdot)$ first extracts general visual features, which are then distilled by the TPB to yield discriminative visual guidance priors $\mathbf{F}^p$:

$$\mathbf{F}^p = TPB(Enc_I(\mathbf{I})), \tag{1}$$

where TPB, comprising three MLP layers with Layer Normalization and LeakyReLU activations (except for the final layer), ensures the retention of only the most task-specific attributes. This approach aligns $\mathbf{F}^p$ with the textual features the diffusion model typically uses in its text-driven generation process, thus facilitating better task alignment for Plugin-Selector. Furthermore, using visual prompts simplifies the user's role by eliminating the need for complex text prompt engineering, which is often challenging for specific vision tasks and sensitive to minor textual variations [78].

However, the task-specific visual guidance prior $\mathbf{F}^p$, while crucial for prompting global semantic attributes, is not sufficient for preserving fine-grained details. In this context, DDIM Inversion plays a pivotal role by providing initial noise that contains information about the image content. Without this step, the inference would rely on random noise devoid of image content, resulting in less controllable results in the diffusion process. However, the inversion process is unstable and time-consuming. To alleviate this, we introduce the SCB to extract and enhance spatial details preservation effectively. We utilizes the pre-trained VAE encoder [11] $Enc_V(\cdot)$, to capture full content of input image $\mathbf{I}$, denoted as $\mathbf{F}$. This comprehensive image detail, when combined with the semantic guidance from $\mathbf{F}^p$, is then processed by our SCB to distill the spatial feature $\mathbf{F}^s$:

$$\mathbf{F}^s = SCB(\mathbf{F}, \mathbf{F}^t, \mathbf{F}^p) = Att(Res(\mathbf{F}, \mathbf{F}^t), \mathbf{F}^t, \mathbf{F}^p), \tag{2}$$

where $\mathbf{F}^t$ is time embedding used to denote the varied time step in diffusion process. The *Res* and *Att* blocks represent the standard ResNet and Cross-Attention transformer blocks, from the diffusion model [54]. The output from *Res* is utilized as the Query features and $\mathbf{F}^p$ acts as both Key and Value features in the cross-attention layer.

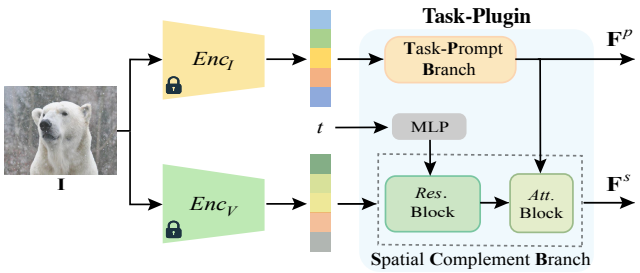


Figure 4. Schematic illustration of task-specific priors extraction via the proposed lightweight Task-Plugin. Task-Plugin processes three inputs: time step t, visual prompt from $Enc_I(\cdot)$, and image content from $Enc_V(\cdot)$. It distills visual guidance $\mathbf{F}^p$ via a task-prompt branch and extracts spatial features $\mathbf{F}^s$ through a spatial complement branch, jointly for task-specific priors.

We then introduce the task-specific visual guidance prior $\mathbf{F}^p$ into the cross-attention layers of the diffusion model, where it serves to direct the model's generation process toward the specific requirements of the low-level vision task. Following this, we directly incorporate the distilled spatial prior $\mathbf{F}^s$ into the final stage of the decoder as a residual. This placement is based on our experimental observations in Table 4, which indicated that the fidelity of spatial details in the stable diffusion [54] tends to decrease from the shallow layers to the deeper ones. By adding $\mathbf{F}^s$ at this specific stage, we effectively counteract this tendency, thereby enhancing the preservation of fine-grained spatial details.

To train the Task-Plugin modules, we adopt the denoising loss as defined in [54], introducing the task-specific priors into the diffusion denoising training process:

$$\mathcal{L} = \mathbb{E}_{\boldsymbol{z}_0, t, \mathbf{F}^p, \mathbf{F}^s, \epsilon \sim \mathcal{N}(0,1)} \left[\| \epsilon - \epsilon_\theta \left(\boldsymbol{z}_t, t, \mathbf{F}^p, \mathbf{F}^s \right) \|_2^2 \right], \tag{3}$$

where $\mathbf{z}_t = \sqrt{\bar{\alpha}_t}\mathbf{z}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t$ represents the noised version of the latent-space image at time t, and $\mathbf{z}_0$, the latent-space representation of the ground truth image $\hat{\mathbf{I}}$, is obtained as $\mathbf{z}_0 = Enc_V(\hat{\mathbf{I}})$. This loss function ensures that the Task-Plugin is effectively trained to incorporate the task-specific priors in guiding the diffusion process.

3.4. Plugin-Selector

We propose the Plugin-Selector, enabling users to select the desired Task-Plugin using text input. For an input image $\mathbf{I}$ and a text prompt $\mathbf{T}$, we define the set of Task-Plugins as $\mathcal{P} = \{\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_m\}$, with each $\mathcal{P}_i$ corresponding to a specific vision task, transforming $\mathbf{I}$ into task-specific priors $(\mathbf{F}_i^p, \mathbf{F}_i^s)$. Then, visual guidance $\mathbf{F}_i^p$ of each Task-Plugin is then cast to a new textual-visual aligned multi-modality latent space via a shared visual projection head $VP(\cdot)$ and denoted as $\mathcal{V} = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_m\}$. Concurrently, $\mathbf{T}$ is encoded into a text embedding by $Enc_T(\cdot)$ [49] and then projected to $\boldsymbol{q}$ using a textual project head $TP(\cdot)$, aligning the

textual and visual embedding. The process is formulated as:

$$\boldsymbol{v}_i = VP(\mathbf{F}_i^p); \quad \boldsymbol{q} = TP(Enc_T(\mathbf{T})). \tag{4}$$

We then compare the textual embedding $\boldsymbol{q}$ with each visual embedding $\boldsymbol{v}_i \in \mathcal{V}$ using cosine similarity function such that $\boldsymbol{s}_i = \text{sim}(\boldsymbol{v}_i, \boldsymbol{q})$, yielding a set of similarity scores $\mathcal{S} = \{\boldsymbol{s}_1, \boldsymbol{s}_2, \cdots, \boldsymbol{s}_m\}$. We select the Task-Plugin $\mathcal{P}_{\text{selected}}$ that meet a specified similarity threshold, θ:

$$\mathcal{P}_{\text{selected}} = \{\mathcal{P}_i \mid \boldsymbol{s}_i \geq \theta, \mathcal{P}_i \in \mathcal{P}\}. \tag{5}$$

We adopt the $\mathbf{F}_i^p$ as the pseudo label and pair it with task-specific text to construct training data. We employ contrastive loss [5, 49] to optimize the vision and text projection heads, enhancing their capability to handle multi-task scenarios. This involves minimizing the distance between the anchor image and positive texts while increasing the distance from negative texts. For each image $\mathbf{I}$, a positive text relevant to its task (*e.g.*, "*I want to remove rain*" for deraining task) and N negative texts from other tasks (*e.g.*, "*enhance the face*" for face restoration) are sampled. The loss function for a positive pair of example (i, j) is as follows:

$$\ell_{i,j} = -\log \frac{\exp\left(\text{sim}\left(\boldsymbol{v}_i, \boldsymbol{q}_j\right)/\tau\right)}{\sum_{k=1}^{N+1} \mathbb{1}_{[k_c \neq i_c]} \exp\left(\text{sim}\left(\boldsymbol{v}_i, \boldsymbol{q}_k\right)/\tau\right)}, \tag{6}$$

where c represents the task type for each sample and $\mathbb{1}_{[k_c \neq i_c]} \in \{0,1\}$ is an indicator function evaluating to 1 iff $k_c \neq i_c$. τ denotes a temperature parameter.

4. Experiments

In this section, we first introduce our experimental setup, including datasets, implementation, and metrics. We then compare *Diff-Plugin* with current diffusion- and regression-based methods in Sec. 4.1, and conduct component analysis of *Diff-Plugin* via ablation studies in Sec. 4.2.

Datasets. To train the Task-Plugins, we utilize specific datasets for each low-level task, desnowing: Snow100K [36], dehazing: Reside [32], deblurring: Gopro [41], deraining: merged train [86], face restoration: FFHQ [26], low-light enhancement: LOL [72], demoireing: LCDMoire [85], and highlight removal: SHIQ [13]. For testing, we evaluate on real-world benchmark datasets, desnowing: realistic test [36], dehazing: RTTS [32], deblurring: RealBlur-J [53], deraining: real test [68], face restoration: LFW [23, 69], low-light enhancement: merged low-light [17, 30, 38, 64, 67, 72], demoireing: LCDMoire [85], and highlight removal: SHIQ [13]. To train the Plugin-Selector, we employ GPT [44] to generate text prompts for each task.

Implementation. During training and testing, we resize the image to 512×512 for a fair comparison. We employ the AdamW optimizer [37] with its default parameters (*e.g.*, betas, weight decay). The training of our Task-Plugins was conducted using a constant learning rate of $1e^{-5}$ and a batch size of 64 on four A100 GPUs, each with 80G of memory. To train the Plugin-Selector, we randomly sample 5,000 images from each task and augment text diversity by randomly combining text inputs from various tasks. We set the batch size to 8 and adopt the same learning rate for Task-Plugins. For negative texts, we set $N = 7$ by default. During inference, we set the specified similarity threshold $\theta = 0$.

Metrics. We follow [54] to employ widely adopted non-reference perceptual metrics, FID [20] and KID [3], to evaluate our *Diff-Plugin* on real data, as GT is not always available. As for the Plugin-Selector, we follow multi-label object classification [6] to report the mean average precision (mAP), the average per-class precision (CP), F1 (CF1), and the average overall precision (OP), recall (OR), and F1 (OF1). For each class (*i.e.*, task type), the labels are predicted as positive if their confidence score is greater than θ. We further propose a stringent zero-tolerance evaluation metric (ZTA) that rigorously assesses sentence-level classification results from a user-first perspective, making binary classification to ensure utmost accuracy:

$$\text{ZTA} = \frac{1}{Q}\sum_{i=1}^{Q}\left(\left(\min_{j \in Y_i} S_{ij} > \theta\right) \wedge \left(\max_{k \in H_i} S_{ik} \leq \theta\right)\right), \tag{7}$$

where Q is the total number of test samples, S_i is the set of predicted similarity scores for sample i, Y_i is the set of indices for positive classes (*i.e.*, user interested tasks), H_i is the set of indices for negative classes (*i.e.*, irrelevant tasks).

4.1. Comparison with State-of-the-Art Methods

We compare the proposed *Diff-Plugin* with the state-of-the-art methods from different low-level vision tasks, including regression-based specialized models: DDMSNet [88], PMNet [81], Restormer [87], NeRCO [79], VQFR [15], UHDM [84], SHIQ [13], multi-task models: AirNet [33], WGWS-Net [98] and PromptIR [47], and diffusion-based models: SD [54], PNP [63], P2P [46], InstructP2P [4], Null-Text [39] and ControlNet [90]. We conduct the experiment on real-world datasets to compare the generalization ability.

Qualitative Results. Fig. 5 demonstrates the superior performances of our *Diff-Plugin* on eight low-level vision tasks with challenging natural images. First, using SD's *img2img* [54] function does not ensure content accuracy. It often leads to major scene changes (column 8). InstructP2P [4], which lacks task-specific priors, also falls short, producing poorer results in tasks like dehazing and low-light enhancement (column 7). The lack of task-specific priors also leads P2P [46] and Null-Text [39] into generating inconsistent contents (columns 5 and 6), despite using initial noise from DDIM Inversion [61]. ControlNet [90] handles some tasks well (column 4) by providing condition information via a diffusion branch, but its strong color distortion reduces its

Figure 5. **Qualitative Comparison.** Our *Diff-Plugin* notably surpasses regression-based method (3) and diffusion-based methods (4)-(8) in performance. Magnified regions of several tasks are provided for clarity. Refer to **Supplemental** for further comparisons.

effectiveness in these tasks. The latest multi-task method, PromptIR [47] (column 3), is limited by model scale and can only handle a few tasks. In contrast, our method uses a lightweight task-specific plugin for each task, offering flexibility and stable performance across all tasks (column 2).

Quantitative Results. We also provide the quantitative comparison in Table 1. Compared with diffusion-based methods, our Diff-Plugin achieves SOTA results overall. While PNP [63] and InstructP2P [4] are capable of producing high-quality images with low FID & KID, they often produce significant content alterations (refer to Fig. 5). Compared with regression-based multi-task methods, our approach delivers competitive performances in most tasks, though it is slightly ineffective in sparse degradation tasks like demoireing and highlight removal. While specialized models may outperform ours in their respective areas, their task-dependent designs limit their applicability to other tasks. Note that the primary goal of this paper is not to achieve top performances in all tasks, but to lay groundwork for future advancements. In addition, *Diff-Plugin*, enables text-driven low-level task processing, a capability absent in regression-based models.

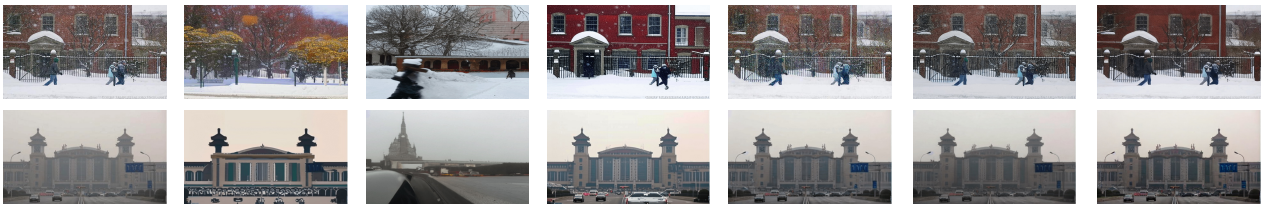
User Study. We conduct a user study with 46 participants to assess various methods through subjective evaluation. Each participant reviewed 5 image sets from the test set, each comprising an input image and 10 predicted images, for a total of 8 tasks. The images were ranked based on content consistency, degradation removal (*e.g.*, rain, snow, highlight), and overall quality. Analyzing 1,840 rankings (46 participants $\times$ 40 sets), we compute the Average Ranking (AR) of each method. Table 2 shows the results. It is obvious to see a preference for our approach among the users.

	Desnowing Realistic [36]		Dehazing Reside [32]		Deblurring RealBlur-J [53]		Deraining real test [68]		Low-light Enhanc. merged low.		Face Restoration LFW [69]		Demoireing LCDMoire [85]		Highlight Removal SHIQ [13]	
	FID ↓	KID ↓	FID ↓	KID ↓	FID ↓	KID ↓	FID ↓	KID ↓	FID ↓	KID ↓	FID ↓	KID ↓	FID ↓	KID ↓	FID ↓	KID ↓
Regression-based *specialized* models																
All	33.92	5.39	36.40	15.66	55.64	15.70	52.78	16.28	48.47	10.96	19.28	6.72	29.59	1.45	33.74	18.79
Regression-based *multi-task* models																
AirNet* [33]	35.02	5.52	39.53	17.86	59.38	20.95	52.04	16.20	59.92	19.74	31.03	13.35	33.05	4.27	10.13	5.89
WGWS-Net* [98]	34.84	5.71	36.25	15.79	56.80	16.83	53.64	16.55	53.67	12.99	29.89	12.08	29.86	2.28	8.28	3.05
PromptIR* [47]	34.66	5.35	40.88	17.80	55.37	16.42	53.78	16.88	53.42	13.16	30.52	12.80	29.01	1.56	9.01	5.07
Diffusion-based models																
SD [54]	35.24	7.88	48.89	24.47	59.21	18.96	51.78	17.69	53.09	15.38	30.90	9.63	58.20	17.34	36.54	12.06
PNP [75]	35.01	6.52	42.82	16.98	63.16	23.58	52.89	21.02	54.19	14.43	34.08	13.45	36.37	6.18	33.09	14.94
P2P [46]	34.48	6.03	42.17	17.33	63.43	25.15	44.49	13.94	52.06	13.26	54.67	24.66	36.37	9.35	26.96	13.11
InstructP2P [4]	42.01	8.54	33.48	12.76	57.38	19.37	54.12	17.87	55.65	15.25	24.66	9.73	34.29	4.73	16.80	6.81
Null-Text [39]	60.49	16.38	39.94	14.88	60.38	20.37	51.49	15.43	52.86	12.79	33.06	12.82	33.72	4.91	14.65	6.52
ControlNet* [90]	34.36	5.70	37.02	15.45	52.30	17.19	52.55	15.22	51.56	15.51	21.59	7.84	41.97	8.80	15.75	8.17
Diff-Plugin (ours)	34.30	5.20	34.68	14.38	51.81	14.63	50.55	13.84	48.98	11.73	20.07	6.91	29.77	1.75	12.58	6.37

Table 1. Quantitative comparisons to SOTAs (both regression-based and diffusion-based methods) on eight low-level vision tasks that need high content-preservation. We summarise all the regression-based specialized models in one line, denoted as "*All*". They are: DDMSNet [88] (desnowing), PMNet [81] (dehazing), Restormer [87] (deblurring and deraining), NeRCO [79] (low-light enhancement), VQFR [15] (face restoration), UHDM [84] (demoireing), SHIQ [13] (highlight removal). KID values are scaled by a factor of 100 for readability. * means that this method is re-trained on eight tasks by us. The best and second-best results are highlighted.

Methods	AirNet [33]	WGWS-Net [98]	PromptIR [47]	SD [54]	PNP [63]	P2P [46]	InstructP2P [4]	Null-Text [39]	ControlNet [90]	Ours
AR ↓	5.26	2.75	3.04	9.66	6.32	7.39	7.14	7.94	4.33	1.17

Table 2. Average Ranking (AR) of different methods in the User Study. The lower the value, the better the human subjective evaluation.



Input ① Inversion+Edit. ② TPB ③ TPB+Inversion ④ SCB ⑤ TPB+SCB (Rec.) Ours

Figure 6. Visual comparison of various Task-Plugin design variants. Row 1 and Row 2 showcase desnowing and dehazing, respectively.

4.2. Ablation Study

Task-Plugin. We first evaluate the efficacy of Task-Plugins by exploring various ablated designs and comparing their performances on desnowing and dehazing. Unless specified otherwise, random noise is used during inference. We have five ablated models. ① Inversion + Editing: DDIM Inversion with a task-specific description (*e.g.*, "*a photo of a snowy day*") inverts the input image into an initial noise, retaining content. This is followed by editing using a target description (*e.g.*, "*a photo of a sunny day*"). ② TPB: The SCB is removed, focusing solely on TPB training. ③ TPB + Inversion: Only TPB is trained, but DDIM Inversion is used for initial noise during inference. ④ SCB: The TPB is removed to train the SCB exclusively. ⑤ TPB + SCB (Reconstruction): Training begins with SCB using self-reconstruction denoising loss, and then proceeds to TPB training with the fixed SCB. Performance results and comparison are presented in Fig. 6 and Table 3.

We have the following observations. ① Inversion + Editing captures the global structure of the input image but loses detailed content. ② TPB provides task-specific visual guidance but lacks spatial content constraints due to its focus on advanced features only. ③ TPB, using inverted initial noise, excels in structured scenes (*e.g.*, large buildings) but tends to deepen colors and create random content for smaller objects. ④ SCB maintains content details, but without task-specific visual guidance, it struggles to effectively remove degradations (*e.g.*, snow or haze). ⑤ TPB, when combined with reconstruction-based SCB, preserves image content through reconstruction while relying solely on TPB to address degradation. However, as SCB reintroduces all image features in each diffusion iteration, including original degradations (*e.g.*, haze in row-2 of Fig. 6), it inadvertently compromises the desired outcomes. Finally, incorporating the task-specific priors from both TPB and SCB in our Task-Plugin enables high-fidelity low-level task processing.

We also confirm the placement of SPB within the pre-trained SD model on desnowing task and show the results in Table 4. Obviously, we can observe that for both the en-

	Methods \ FID ↓	Desnowing	Dehazing
①	Inversion + Editing	48.54	35.05
②	TPB	36.02	37.73
③	TPB + Inversion	34.87	33.05
④	SCB	34.71	36.16
⑤	TPB + SCB (Reconstruction)	34.50	35.94
	TPB + SCB (Ours)	34.30	34.68

Table 3. Ablation studies of variant Task-Plugin designs on two tasks: desnowing, dehazing. Note that although some variants have much lower FID scores, they tend to generate random content (refer to ①-③ of Fig. 6). In contrast, our final model guarantees both content fidelity and robust metric performances.

Metrics	Encoder				Decoder			
	E-1	E-2	E-3	E-4	D-4	D-3	D-2	D-1
FID ↓	34.33	34.46	36.58	37.41	37.71	34.59	34.20	34.30
KID ↓	5.23	5.52	7.18	7.84	7.57	5.55	5.20	5.20
Param.(MB)	14.88	48.77	182.31				48.77	14.88

Table 4. Ablation studies on the placement of SCB within the pre-trained SD's Encoder/Decoder stages on desnowing. 'E/D-i' represents the i-th stage, with higher numbers indicating deeper layers. We modify the feature dimension in SCB to suit various stages of the pre-trained SD model, resulting in varied parameters.

coder and decoder of the pre-trained SD [54], the fidelity diminishes and performance progressively decreases from the shallower to the deeper stages (*e.g.*, stages 1 to 4). Thus, we inject the spatial features into the final stage of the decoder, balancing performance and parameters. Notably, the parameters of Task-Plugin module is only 1.67% of the SD.

Plugin-Selector. As shown in Table 5, we first evaluate the accuracy of Plugin-Selector in both single-task and multi-task scenarios (row-1 and -2), and observe consistently high accuracy. In addition, in a significantly extensive test with 120,000 samples (denoted as Multi-task*), it achieves an mAP accuracy of 0.936, demonstrating its effectiveness. Further, in a robustness test (denoted as Single + Non.) combining task-specific and task-irrelevant texts, it still achieves a notable zero-toleration accuracy of 0.779.

We also conduct an ablation study on the Plugin-Selector to evaluate the significance of each component, with results detailed in Table 6. ① We remove the visual and textual projection heads separately. ② We assess the impact of varying the number of negative samples for contrastive training. The results first reveal that both visual and textual projection heads are crucial. Omitting the visual head results in training collapse and *NaN* output, while removing the textual head lowers the ZTA metric by 15.4%. It also shows that increasing the number of negative samples (*e.g.*, from $N = 1$ to 15) consistently enhances selection accuracy.

Diverse Applications. Fig. 7 demonstrates the versatility of *Diff-Plugin*. Row-1 exemplifies complex, low-level task execution via sub-task integration (*e.g.*, old photo restoration can be roughly divided into restoration and colorization.). Row-2 highlights its ability to invert low-level tasks, enabling the generation of special effects like rain and snow.

The default batch size is 8, implying 7 neg. samples and 1 pos. sample.

Tasks	ZTA ↑	CP ↑	OP ↑	OR ↑	CF1 ↑	OF1 ↑	mAP ↑
Single-task	0.998	-	0.998	-	-	0.998	0.998
Multi-task	0.979	0.988	0.988	0.927	0.956	0.956	0.933
Multi-task*	0.969	0.983	0.983	0.936	0.960	0.959	0.936
Single + Non.	0.779	0.814	0.808	0.941	0.872	0.870	0.775

Table 5. Quantitative evaluation of the proposed Plugin-Selector. Asterisks (*) denotes more sample combinations. A dash (-) indicates metric not applicable. 'Single + Non' refers to random combinations of single-task text inputs with non-existing (*i.e.*, plugin-irrelevant) tasks, to test the Plugin-Selector's robustness.

Single + Non.	Remove		Number of Negative Samples				
	$VP(\cdot)$	$TP(\cdot)$	1	3	5	7	15
ZTA ↑	*NaN*	0.625	0.559	0.648	0.725	0.779	0.817

Table 6. Ablation studies of Plugin-Selector. '*NaN*' indicates non-convergence of training, resulting in unavailable result.

Figure 7. Diverse uses of *Diff-Plugin*: multi-task combination in row-1 and reversed low-level tasks in row-2.

5. Conclusion

In this paper, we presented *Diff-Plugin*, a novel framework tailored for enhancing pre-trained diffusion models in handling various low-level vision tasks that need stringent details preservation. Our Task-Plugin module, with its dual-branch design, effectively incorporates task-specific priors into the diffusion process to allow for high-fidelity details-preserving visual results without retraining the base model for each task. The Plugin-Selector further adds intuitive user interaction through text inputs, enabling text-driven low-level tasks and enhancing the framework's practicality. Extensive experiments across various vision tasks demonstrate the superiority of our framework over existing methods, especially in real-world scenarios.

One limitation of our current *Diff-Plugin* framework is the inability in local editing. For example, in Fig. 1, our method may fail to only remove snow specifically from the river but remain those in the sky. One possible solution for this problem is to integrate LLMs [35, 97] to indicate the region in which the task is performed.

References

[1] Omri Avrahami, Thomas Hayes, Oran Gafni, Sonal Gupta, Yaniv Taigman, Devi Parikh, Dani Lischinski, Ohad Fried, and Xi Yin. Spatext: Spatio-textual representation for controllable image generation. In *CVPR*, pages 18370–18380, 2023. 3

[2] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv*, 2022. 2

[3] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. In *ICLR*, 2018. 5

[4] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, pages 18392–18402, 2023. 1, 2, 5, 6, 7

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607, 2020. 5

[6] Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. Multi-label image recognition with graph convolutional networks. In *CVPR*, pages 5177–5186, 2019. 5

[7] Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *ICLR*, 2023. 3

[8] Guillaume Couairon, Marlène Careil, Matthieu Cord, Stéphane Lathuilière, and Jakob Verbeek. Zero-shot spatial layout conditioning for text-to-image diffusion models. In *ICCV*, pages 2174–2183, 2023. 2

[9] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, pages 8780–8794, 2021. 1, 2

[10] Wenkai Dong, Song Xue, Xiaoyue Duan, and Shumin Han. Prompt tuning inversion for text-driven image editing using diffusion models. In *ICCV*, pages 7430–7440, 2023. 2

[11] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, pages 12873–12883, 2021. 4

[12] Ben Fei, Zhaoyang Lyu, Liang Pan, Junzhe Zhang, Weidong Yang, Tianyue Luo, Bo Zhang, and Bo Dai. Generative diffusion prior for unified image restoration and enhancement. In *CVPR*, pages 9935–9946, 2023. 3

[13] Gang Fu, Qing Zhang, Lei Zhu, Ping Li, and Chunxia Xiao. A multi-task network for joint specular highlight detection and removal. In *CVPR*, pages 7752–7761, 2021. 5, 7

[14] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *ICLR*, 2023. 2

[15] Yuchao Gu, Xintao Wang, Liangbin Xie, Chao Dong, Gen Li, Ying Shan, and Ming-Ming Cheng. Vqfr: Blind face restoration with vector-quantized dictionary and parallel decoder. In *ECCV*, pages 126–143, 2022. 5, 7

[16] Lanqing Guo, Chong Wang, Wenhan Yang, Siyu Huang, Yufei Wang, Hanspeter Pfister, and Bihan Wen. Shadowdiffusion: When degradation prior meets diffusion model for shadow removal. In *CVPR*, pages 14049–14058, 2023. 1, 3

[17] Xiaojie Guo, Yu Li, and Haibin Ling. Lime: Low-light image enhancement via illumination map estimation. *IEEE TIP*, 26(2):982–993, 2016. 5

[18] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 3

[19] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. In *ICLR*, 2022. 2

[20] 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. In *NeurIPS*, 2017. 5

[21] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv*, 2022. 1, 2

[22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, pages 6840–6851, 2020. 1, 2, 3

[23] Gary B Huang, Marwan Mattar, Tamara Berg, and Eric Learned-Miller. Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Technical report, University of Massachusetts, Amherst*, 2007. 5

[24] Hai Jiang, Ao Luo, Haoqiang Fan, Songchen Han, and Shuaicheng Liu. Low-light image enhancement with wavelet-based diffusion models. *TOG*, 42(6):1–14, 2023. 3

[25] Laurynas Karazija, Iro Laina, Andrea Vedaldi, and Christian Rupprecht. Diffusion models for zero-shot open-vocabulary segmentation. *arXiv*, 2023. 1

[26] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410, 2019. 5

[27] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *NeurIPS*, 2022. 3

[28] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *CVPR*, pages 6007–6017, 2023. 1, 2

[29] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *CVPR*, pages 1931–1941, 2023. 2

[30] Chulwoo Lee, Chul Lee, and Chang-Su Kim. Contrast enhancement based on layered difference representation of 2d histograms. *IEEE TIP*, 22(12):5372–5384, 2013. 5

[31] Alexander C. Li, Mihir Prabhudesai, Shivam Duggal, Ellis Brown, and Deepak Pathak. Your diffusion model is secretly a zero-shot classifier. In *ICCV*, pages 2206–2217, 2023. 1

[32] Boyi Li, Wenqi Ren, Dengpan Fu, Dacheng Tao, Dan Feng, Wenjun Zeng, and Zhangyang Wang. Benchmarking single-image dehazing and beyond. *IEEE TIP*, 28(1):492–505, 2018. 5, 7

[33] Boyun Li, Xiao Liu, Peng Hu, Zhongqin Wu, Jiancheng Lv, and Xi Peng. All-in-one image restoration for unknown corruption. In *CVPR*, pages 17452–17462, 2022. 3, 5, 7

[34] Xinqi Lin, Jingwen He, Ziyan Chen, Zhaoyang Lyu, Ben Fei, Bo Dai, Wanli Ouyang, Yu Qiao, and Chao Dong. Diffbir: Towards blind image restoration with generative diffusion prior. *arXiv*, 2023. 3

[35] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 8

[36] Yun-Fu Liu, Da-Wei Jaw, Shih-Chia Huang, and Jenq-Neng Hwang. Desnownet: Context-aware deep network for snow removal. *IEEE TIP*, 27(6):3064–3073, 2018. 5, 7

[37] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv*, 2017. 5

[38] Kede Ma, Kai Zeng, and Zhou Wang. Perceptual quality assessment for multi-exposure image fusion. *IEEE TIP*, 24 (11):3345–3356, 2015. 5

[39] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *CVPR*, pages 6038–6047, 2023. 2, 5, 6, 7

[40] Chong Mou, Xintao Wang, Liangbin Xie, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv*, 2023. 3

[41] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, pages 3883–3891, 2017. 5

[42] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *PMLR*, 2021. 2

[43] OpenAI. Chatgpt plugins: https://openai.com/blog/chatgpt-plugins. 2023. 3

[44] OpenAI. Gpt-4 technical report. *arXiv*, 2023. 5

[45] Ozan Özdenizci and Robert Legenstein. Restoring vision in adverse weather conditions with patch-based denoising diffusion models. *IEEE TPAMI*, 2023. 3

[46] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *SIGGRAPH*, pages 1–11, 2023. 1, 2, 5, 7

[47] Vaishnav Potlapalli, Syed Waqas Zamir, Salman Khan, and Fahad Shahbaz Khan. Promptir: Prompting for all-in-one blind image restoration. In *NeurIPS*, 2023. 3, 5, 6, 7

[48] Can Qin, Shu Zhang, Ning Yu, Yihao Feng, Xinyi Yang, Yingbo Zhou, Huan Wang, Juan Carlos Niebles, Caiming Xiong, Silvio Savarese, et al. Unicontrol: A unified diffusion model for controllable visual generation in the wild. In *NeurIPS*, 2023. 3

[49] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021. 2, 4, 5

[50] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, pages 5485–5551, 2020. 2

[51] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv*, 2022. 2

[52] Mengwei Ren, Mauricio Delbracio, Hossein Talebi, Guido Gerig, and Peyman Milanfar. Multiscale structure guided diffusion for image deblurring. In *ICCV*, pages 10721–10733, 2023. 1, 3

[53] Jaesung Rim, Haeyun Lee, Jucheol Won, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *ECCV*, pages 184–201, 2020. 5, 7

[54] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 2, 4, 5, 6, 7, 8

[55] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, pages 22500–22510, 2023. 2

[56] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *SIGGRAPH*, pages 1–10, 2022. 1, 3

[57] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, pages 36479–36494, 2022. 2

[58] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE TPAMI*, 45(4): 4713–4726, 2022. 3

[59] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. In *NeurIPS*, pages 25278–25294, 2022. 2

[60] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, pages 2256–2265, 2015. 2

[61] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 1, 2, 5

[62] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *NeurIPS*, 2019. 2

[63] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *CVPR*, pages 1921–1930, 2023. 1, 2, 5, 6, 7

[64] Vassilios Vonikakis, Rigas Kouskouridas, and Antonios Gasteratos. On the evaluation of illumination compensation algorithms. *MTA*, 77:9211–9231, 2018. 5

[65] Bram Wallace, Akash Gokul, and Nikhil Naik. Edict: Exact diffusion inversion via coupled transformations. In *CVPR*, pages 22532–22541, 2023. 2

[66] Jianyi Wang, Zongsheng Yue, Shangchen Zhou, Kelvin CK Chan, and Chen Change Loy. Exploiting diffusion prior for real-world image super-resolution. *arXiv*, 2023. 3

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

[68] Tianyu Wang, Xin Yang, Ke Xu, Shaozhe Chen, Qiang Zhang, and Rynson WH Lau. Spatial attentive single-image deraining with a high quality real rain dataset. In *CVPR*, pages 12270–12279, 2019. 5, 7

[69] Xintao Wang, Yu Li, Honglun Zhang, and Ying Shan. Towards real-world blind face restoration with generative facial prior. In *CVPR*, pages 9168–9178, 2021. 5, 7

[70] Yinhuai Wang, Jiwen Yu, and Jian Zhang. Zero-shot image restoration using denoising diffusion null-space model. In *ICLR*, 2022. 3

[71] Zhixin Wang, Ziying Zhang, Xiaoyun Zhang, Huangjie Zheng, Mingyuan Zhou, Ya Zhang, and Yanfeng Wang. Dr2: Diffusion-based robust degradation remover for blind face restoration. In *CVPR*, pages 1704–1713, 2023. 1, 3

[72] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. In *BMVC*, 2018. 5

[73] Jay Whang, Mauricio Delbracio, Hossein Talebi, Chitwan Saharia, Alexandros G Dimakis, and Peyman Milanfar. Deblurring via stochastic refinement. In *CVPR*, pages 16293–16303, 2022. 3

[74] Bin Xia, Yulun Zhang, Shiyin Wang, Yitong Wang, Xinglong Wu, Yapeng Tian, Wenming Yang, and Luc Van Gool. Diffir: Efficient diffusion model for image restoration. In *ICCV*, pages 13095–13105, 2023. 3

[75] Chaojun Xiao, Zhengyan Zhang, Xu Han, Chi-Min Chan, Yankai Lin, Zhiyuan Liu, Xiangyang Li, Zhonghua Li, Zhao Cao, and Maosong Sun. Plug-and-play document modules for pre-trained models. In *ACL*, 2023. 3, 7

[76] Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. Boxdiff: Text-to-image synthesis with training-free box-constrained diffusion. In *ICCV*, pages 7452–7461, 2023. 2

[77] Canwen Xu, Yichong Xu, Shuohang Wang, Yang Liu, Chenguang Zhu, and Julian McAuley. Small models are valuable plug-ins for large language models. *arXiv*, 2023. 3

[78] Xingqian Xu, Jiayi Guo, Zhangyang Wang, Gao Huang, Irfan Essa, and Humphrey Shi. Prompt-free diffusion: Taking" text" out of text-to-image diffusion models. *arXiv*, 2023. 4

[79] Shuzhou Yang, Moxuan Ding, Yanmin Wu, Zihan Li, and Jian Zhang. Implicit neural representation for cooperative low-light image enhancement. In *ICCV*, pages 12918–12927, 2023. 5, 7

[80] 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*, pages 1357–1366, 2017. 3

[81] Tian Ye, Yunchen Zhang, Mingchao Jiang, Liang Chen, Yun Liu, Sixiang Chen, and Erkang Chen. Perceiving and modeling density for image dehazing. In *ECCV*, pages 130–145, 2022. 5, 7

[82] Tian Ye, Sixiang Chen, Jinbin Bai, Jun Shi, Chenghao Xue, Jingxia Jiang, Junjie Yin, Erkang Chen, and Yun Liu. Adverse weather removal with codebook priors. In *ICCV*, pages 12653–12664, 2023. 3

[83] Xunpeng Yi, Han Xu, Hao Zhang, Linfeng Tang, and Jiayi Ma. Diff-retinex: Rethinking low-light image enhancement with a generative diffusion model. In *CVPR*, pages 12302–12311, 2023. 1, 3

[84] Xin Yu, Peng Dai, Wenbo Li, Lan Ma, Jiajun Shen, Jia Li, and Xiaojuan Qi. Towards efficient and scale-robust ultra-high-definition image demoiréing. In *ECCV*, pages 646–662, 2022. 5, 7

[85] Shanxin Yuan, Radu Timofte, Gregory Slabaugh, Aleš Leonardis, Bolun Zheng, Xin Ye, Xiang Tian, Yaowu Chen, Xi Cheng, Zhenyong Fu, et al. Aim 2019 challenge on image demoireing: Methods and results. In *ICCVW*, pages 3534–3545, 2019. 5, 7

[86] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *CVPR*, pages 14821–14831, 2021. 5

[87] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *CVPR*, pages 5728–5739, 2022. 5, 7

[88] Kaihao Zhang, Rongqing Li, Yanjiang Yu, Wenhan Luo, and Changsheng Li. Deep dense multi-scale network for snow removal using semantic and depth priors. *IEEE TIP*, 30: 7419–7431, 2021. 5, 7

[89] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. In *NeurIPS*, 2023. 2

[90] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, pages 3836–3847, 2023. 3, 5, 6, 7

[91] Yuxin Zhang, Nisha Huang, Fan Tang, Haibin Huang, Chongyang Ma, Weiming Dong, and Changsheng Xu. Inversion-based style transfer with diffusion models. In *CVPR*, pages 10146–10156, 2023. 1

[92] Yi Zhang, Xiaoyu Shi, Dasong Li, Xiaogang Wang, Jian Wang, and Hongsheng Li. A unified conditional framework for diffusion-based image restoration. In *NeurIPS*, 2023. 3

[93] Zhixing Zhang, Ligong Han, Arnab Ghosh, Dimitris N Metaxas, and Jian Ren. Sine: Single image editing with text-to-image diffusion models. In *CVPR*, pages 6027–6037, 2023. 2

[94] Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. In *NeurIPS*, 2023. 3

[95] Yang Zhao, Tingbo Hou, Yu-Chuan Su, Xuhui Jia, Yandong Li, and Matthias Grundmann. Towards authentic face restoration with iterative diffusion models and beyond. In *ICCV*, pages 7312–7322, 2023. 3

[96] Yuzhong Zhao, Qixiang Ye, Weijia Wu, Chunhua Shen, and Fang Wan. Generative prompt model for weakly supervised object localization. In *ICCV*, pages 6351–6361, 2023. 1

[97] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. In *ICLR*, 2024. 8

[98] Yurui Zhu, Tianyu Wang, Xueyang Fu, Xuanyu Yang, Xin Guo, Jifeng Dai, Yu Qiao, and Xiaowei Hu. Learning weather-general and weather-specific features for image restoration under multiple adverse weather conditions. In *CVPR*, pages 21747–21758, 2023. 3, 5, 7