DiffRetouch: Using Diffusion to Retouch on the Shoulder of Experts

Zheng-Peng Duan[1, 2], Jiawei Zhang[2], Zheng Lin[3], Xin Jin[1], XunDong Wang[5], Dongqing Zou[2, 4], Chun-Le Guo[1, 6], Chongyi Li[1, 6*]

[1]VCIP, CS, Nankai University
[2]SenseTime Research
[3]BNRist, Department of Computer Science and Technology, Tsinghua University
[4]PBVR
[5]Wuhan University of Technology
[6]NKIARI, Shenzhen Futian
{adamduan0211, zhjw1988, frazer.linzheng, srameojin}@gmail.com wangxundong@whut.edu.cn
dqzou.lhi@gmail.com {guochunle, lichongyi}@nankai.edu.cn

Abstract

Image retouching aims to enhance the visual quality of photos. Considering the different aesthetic preferences of users, the target of retouching is subjective. However, current retouching methods mostly adopt deterministic models, which not only neglects the style diversity in the expert-retouched results and tends to learn an average style during training, but also lacks sample diversity during inference. In this paper, we propose a diffusion-based method, named DiffRetouch, for image retouching. Thanks to the distribution modeling ability of diffusion, our method can capture the complex fine-retouched distribution covering various visual-pleasing styles in the training data. Moreover, four image attributes are made adjustable to provide a user-friendly editing mechanism. By adjusting these attributes in specified ranges, users are allowed to customize preferred styles within the learned fine-retouched distribution. Additionally, the affine bilateral grid and contrastive learning scheme are introduced to handle the problem of texture distortion and control insensitivity, respectively. Extensive experiments demonstrate the performance of our method on visually appealing and sample diversity.

Project — https://adam-duan.github.io/projects/retouch/

Introduction

Smartphones have made taking photographs a daily activity. However, captured photographs may be unsatisfactory due to varying factors like the illumination condition (Wang et al. 2019; Zhu et al. 2020; Liu et al. 2021; Li, Guo, and Loy 2021). Thus, post-processing is inevitably desired. A series of professional image-processing software (Hu et al. 2018; Kosugi and Yamasaki 2020; Ouyang et al. 2023; Tseng et al. 2022) provide the users with useful tools to improve image quality. However, these manual adjustments require specialized skills. To help non-experts obtain visual-pleasing photos automatically, numerous deep learning-based methods (Moran, McDonagh, and Slabaugh 2021; Gharbi et al. 2017; Wang et al. 2019; Chen et al. 2018a; Kim et al. 2021; Sun et al. 2021; Chen et al. 2018b; He et al. 2020; Shin et al. 2024) for image retouching have been proposed.

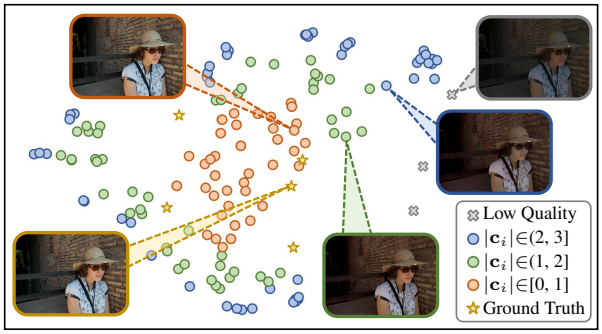

Figure 1: DiffRetouch supports editing the retouching style by adjusting the condition $\mathbf{c}$, where each coefficient $\mathbf{c}_i$ corresponds to one image attribute. We generate numerous results with $|\mathbf{c}_i|$ randomly sampled in [0,1], (1,2], and (2,3]. The features of these results extracted by style encoder (Song, Qian, and Du 2021) are shown using t-SNE (Van der Maaten and Hinton 2008). Since our DiffRetouch is trained with $|\mathbf{c}_i|$ limited to [0,1], the results sampled in this range are within the fine-retouched distribution surrounded by ground truths, otherwise, the results will deviate from it and be closer to low-quality images. This means that users can adjust within [0,1] to obtain preferred styles and meanwhile the final outputs tend to be objectively visual-pleasing.

Considering the different aesthetic preferences of users, retouching is a subjective process. Even the same expert may adjust the images with different styles to satisfy various demands (Song, Qian, and Du 2021). However, most methods (He et al. 2020; Gharbi et al. 2017; Zeng et al. 2020; Moran et al. 2020; Moran, McDonagh, and Slabaugh 2021; Li et al. 2020; Li, Guo, and Loy 2021; Li et al. 2022) ignore the subjectivity of this task and adopt deterministic models. Their drawbacks come from three aspects. **1) Although trained with the subset retouched by one specific expert, they neglect the intrinsic diversity within it, and actually learn the average style.** This situation is more serious when trained with images retouched by multiple experts. **2) During inference, they can only produce one retouching style, which may not always meet users' aesthetic prefer-**

*Corresponding author.

ences. To generate additional retouching results, they need to train multiple models, which limits practical applications. **3) Although several methods (Song, Qian, and Du 2021; Kim, Koh, and Kim 2020; Kim and Lee 2023) support additional styles, extra images are required to indicate the desired style.** This adds extra burden to users and adjusting the style through images is ambiguous.

In practice, a retouching method needs to cover the fine-retouched distribution, which includes various visual-pleasing styles to satisfy different aesthetic preferences. Recently, diffusion (Ho, Jain, and Abbeel 2020; Nichol and Dhariwal 2021) has shown its strong modeling ability for complex distributions, and it has shown potential in various low-level vision tasks (Saharia et al. 2022; Lugmayr et al. 2022; Xie et al. 2023; Özdenizci and Legenstein 2023; Whang et al. 2022; Ren et al. 2023; Wang et al. 2023b; Yin et al. 2023; Hou et al. 2023; Lv et al. 2024), especially in image super-resolution (Yang et al. 2023; Wu et al. 2023; Chen et al. 2023; Sun et al. 2023; Yu et al. 2024). In this work, we introduce the diffusion model into image retouching, where the benefits come from two aspects. 1) The diffusion-based model can capture the complex distributions covering various styles appearing in the training data even when trained with results retouched by multiple experts. 2) During inference, the model can generate various styles within the fine-retouched distribution with no additional images.

In this study, we propose a Stable Diffusion-based (Rombach et al. 2022) retouching method, which is trained by directly conditioning on low-quality input images via concatenation. To provide the user with a friendly and understandable editing mechanism, four image attributes (Wang, Chan, and Loy 2023) are made adjustable by coefficients. These coefficients constitute a vector and are mapped to the intermediate layers of U-Net via cross-attention mechanism.

However, due to the information loss existing in the encoding and decoding process (Jiang et al. 2023), the results will exhibit noticeable distortion in image textures, which we refer to as **texture distortion**. Inspired by HDRNet (Gharbi et al. 2017), we introduce the affine bilateral grid to address this problem. Besides the latent prediction for the progressive denoising process, the underlying U-Net backbone within each denoising step outputs the affine bilateral grid as well. As for the last denoising step, we directly apply the affine transformations, which are inferred from the obtained bilateral grid, to the input image. The other problem is that the influence caused by adjusting these attributes tends to be weak to satisfy practical needs, which we denote as **control insensitivity**. To encourage models more aware of the adjustment brought by each coefficient, we design a contrastive learning scheme, which involves explicit supervision w.r.t. these attributes. It is worth mentioning that regardless of how the coefficients are adjusted within the specified range, the final result tends to be sampled from the learned fine-tuned distribution, which is visualized in Figure 1. Therefore, users can adjust within the specified range to achieve their preferred styles, while ensuring that the final outputs remain objectively visually appealing

Our contributions can be summarized as follows:

- We propose a diffusion-based retouching method, to cover the fine-retouched distribution, along with four adjustable attributes to edit the final results.
- The bilateral grid is introduced into the diffusion model to overcome the texture distortion caused by the information loss in the encoding and decoding process.
- To address the control insensitivity, we design a contrastive learning scheme to encourage models more aware of the adjustment brought by each coefficient.

Methodology

Overview

Given an image $\mathbf{R}$ suffering from photographic defects, such as improper exposures and limited contrast, image retouching aims to generate a visually pleasing rendition. In this work, we propose a Stable Diffusion-based method for retouching, named DiffRetouch. To help users customize the styles that better fit their aesthetic preferences, four image attributes are made adjustable by coefficients and constitute the condition input $\mathbf{c}$. Note that our method is a framework where these attributes are extendable. The direct application of Stable Diffusion, along with the condition inputs ($\mathbf{R}$ and $\mathbf{c}$), is viewed as a baseline model. However, the information loss existing in the encoding and decoding process causes texture distortion. To overcome the texture distortion, we introduce the affine bilateral grid into our baseline model.

Stable Diffusion adopts the DDPM training strategy in the latent space, where the reconstruction supervision is imposed on the latent for predicting the added noise. For the training of affine bilateral grid, the reconstruction supervision is also utilized in the pixel space. To alleviate the control insensitivity, we design a contrastive learning scheme.

Architecture

Baseline The baseline model of our DiffRetouch is built upon Stable Diffusion (Rombach et al. 2022), which is a variant of DDPM (Ho, Jain, and Abbeel 2020; Chung et al. 2022). DDPM involves sequentially corrupting training data with noise, and then learning to reverse this corruption. Specifically, denoising model $\boldsymbol{\epsilon}_\theta(\mathbf{X}_t, t)$ is trained to predict the added noise on the sampled image $\mathbf{X}$, where $\mathbf{X}_t$ is a noisy version of $\mathbf{X}$ at timestamp t. The underlying backbone of the denoising model $\boldsymbol{\epsilon}_\theta$ is the time-conditional U-Net. To improve both the training and sampling efficiency of DDPM, Stable Diffusion applies the forward and reverse processes in the latent space rather than in pixel space. Equipped with powerful pre-trained autoencoders consisting of encoder $\mathcal{E}$ and decoder $\mathcal{D}$, Stable Diffusion can efficiently obtain the latent space representation $\mathbf{Z}$ of $\mathbf{X}$ by $\mathbf{Z} = \mathcal{E}(\mathbf{X})$ during training, and transform the latent space samples to the pixel space through $\mathcal{D}$. Moreover, by replacing the original denoising model with the conditional one $\boldsymbol{\epsilon}_\theta(\mathbf{Z}_t, t, \mathbf{m})$, where $\mathbf{Z}_t$ is the noisy latent at timestamp t and $\mathbf{m}$ is the condition input such as text, Stable Diffusion can be turned into more flexible conditional image generators.

Our baseline also performs the DDPM process in the latent space and utilizes the conditional denoising models. The sampling process starts from the Gaussian noise map $\mathbf{Z}_T$, where T is the total number of time steps for denoising, and

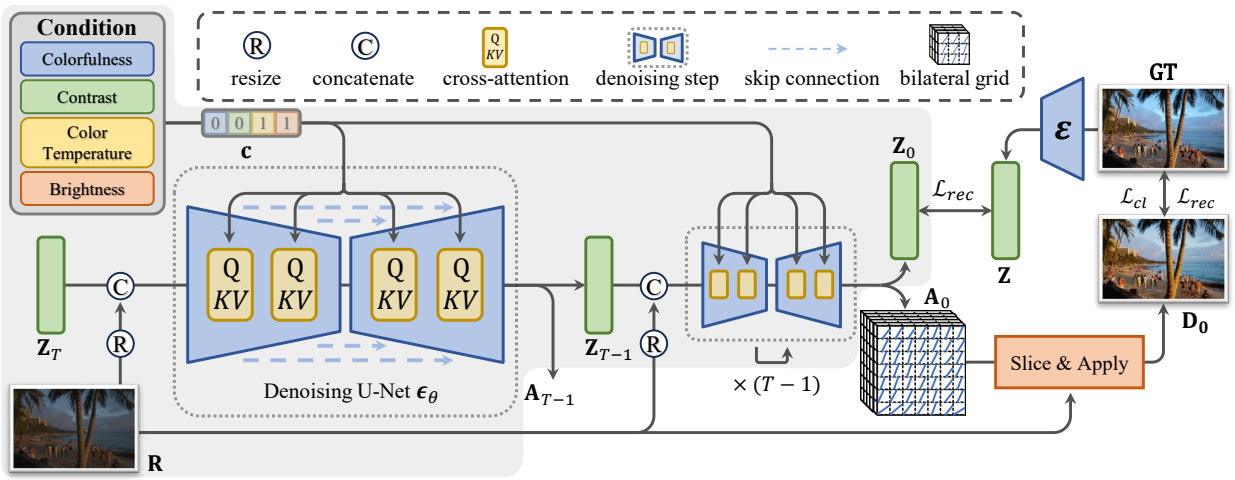

Figure 2: Pipeline of our DiffRetouch. The sampling process and supervision during training are included. The baseline model part is marked in gray. The affine bilateral grid and $\mathcal{L}_{cl}$ are additionally introduced in DiffRetouch to tackle texture distortion and control insensitivity. During training, the denoising model takes the noisy latent $\mathbf{Z}_t$, resized version of $\mathbf{R}$ and condition $\mathbf{c}$ w.r.t. image attributes as input for each step, then generates $\mathbf{Z}_{t-1}$ and affine bilateral grid $\mathbf{A}_{t-1}$ simultaneously. After looking up in $\mathbf{A}$ based on the position and intensity of each pixel in $\mathbf{R}$, which is similar to (Gharbi et al. 2017), the output $\mathbf{D}$ is obtained by matrix multiply between the sliced affine matrics and pixel colors of $\mathbf{R}$. $\mathcal{L}_{rec}$ (Eq. (5)) is imposed in both the latent ($\mathbf{Z}$) and pixel ($\mathbf{D}$) space, along with the $\mathcal{L}_{cl}$ (Eq. (6)). During inference, at each step of the sampling, $\mathbf{Z}_{t-1}$ is used as the input of the next denoising step for the progressive denoising process. Only for the last step, $\mathbf{A}_0$ is used to obtain the final output $\mathbf{D}_0$.

$\mathbf{Z}_T$ has a smaller resolution than input image $\mathbf{R}$. In order to provide the information of the image to be retouched, $\mathbf{R}$ is resized to the same resolution as $\mathbf{Z}_t$, and the resized image $\mathbf{R}'$ is fed into the denoising model via concatenation with $\mathbf{Z}_t$. Another input condition is a vector consisting of four adjustable coefficients regarding four attributes (colorfulness, contrast, color temperature, and brightness), which can be denoted as $\mathbf{c} = [\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \mathbf{c}_4]$. In this design, by adjusting $\mathbf{c}_i$, the final retouching style of the output can be edited on the corresponding attributes. Once the coefficient $\mathbf{c}_i$ is adjusted within $[-1, 1]$, the final result is likely to be sampled in the learned fine-retouched distribution. The adjustable vector is mapped to the intermediate layers of the backbone U-Net via cross-attention layers.

For each denoising step during sampling, the denoising model takes the noisy latent $\mathbf{Z}_t$, the resized low-quality image $\mathbf{R}'$, and adjustable vector $\mathbf{c}$ as input, and output the noise prediction, which can be formulated as

$$\boldsymbol{\epsilon}_{pred,t} = \boldsymbol{\epsilon}_\theta\left([\mathbf{Z}_t, \mathbf{R}'], t, \mathbf{c}\right). \tag{1}$$

With the noise prediction $\boldsymbol{\epsilon}_{pred,t}$, we can obtain $\mathbf{Z}_{t-1}$, which is the input for the next denoising step. The denoising step repeats until obtaining $\mathbf{Z}_0$. By passing $\mathbf{Z}_0$ through $\mathcal{D}$, we can obtain the result output by the baseline model.

Affine Bilateral Grid Despite the great power and good training of the pre-trained autoencoders, the result output by the baseline model has severe texture distortion, which is shown in the top row of Figure 3. The distortion is mainly caused by the inherent stochasticity of the diffusion model (Wang et al. 2023a), as well as the compression-reconstruction process (Jiang et al. 2023).

Inspired by HDRNet (Gharbi et al. 2017), we introduce the affine bilateral grid $\mathbf{A}$ into our DiffRetouch to overcome the texture distortion. $\mathbf{A}$ can be viewed as a 3D array, where each grid is indexed by position and intensity. In each grid, $\mathbf{A}$ stores a 3×4 affine matrix. When applying $\mathbf{A}$ on the full-resolution image, we can lookup in $\mathbf{A}$ with the position and intensity of each pixel, and retrieve the affine matrix through trilinear interpolation. Next, the final color for each pixel is obtained by multiplying the affine matrix with the original color. Due to the nature of the bilateral grid, nearby pixels with similar intensities tend to retrieve similar affine matrics. After matrix multiplication, these nearby pixels still keep similar colors in the output, avoiding distortion.

In our implementation, besides the noise estimation, the underlying U-Net outputs the affine bilateral grid as well, which can be formulated as

$$[\boldsymbol{\epsilon}_{pred}, \tilde{\mathbf{A}}] = \boldsymbol{\epsilon}_\theta\left([\mathbf{Z}_t, \mathbf{R}'], t, \mathbf{c}\right). \tag{2}$$

After unrolling the channels of the output $\tilde{\mathbf{A}}$, we can obtain the affine bilateral grid $\mathbf{A} \in \mathbb{R}^{H_{grid} \times W_{grid} \times D \times 12}$, where D is the dimension of intensity and 12 represents the 3×4 affine matrix. Note that H_{grid} and W_{grid} are the height and width of $\mathbf{A}$, which are smaller than the resolution of $\mathbf{R} \in \mathbb{R}^{H \times W \times 3}$. The operation of slicing and applying $\mathbf{A}$ on the full-resolution image is similar to HDRNet (Gharbi et al. 2017). The guidance map $\mathbf{G} \in \mathbb{R}^{H \times W \times 1}$ is obtained by feeding $\mathbf{R}$ into a pixel-level network. Given the position and intensity indicated by $\mathbf{G}$, the slicing performs a per-pixel lookup in $\mathbf{A}$ and retrieve the sliced $\mathbf{A}' \in \mathbb{R}^{H \times W \times 1 \times 12}$ through trilinear interpolation. The final result is obtained by multiplying the affine matrices with the original colors for the pixels in $\mathbf{R}$. As shown in Figure 3, the final result of our DiffRetouch maintains the original details.

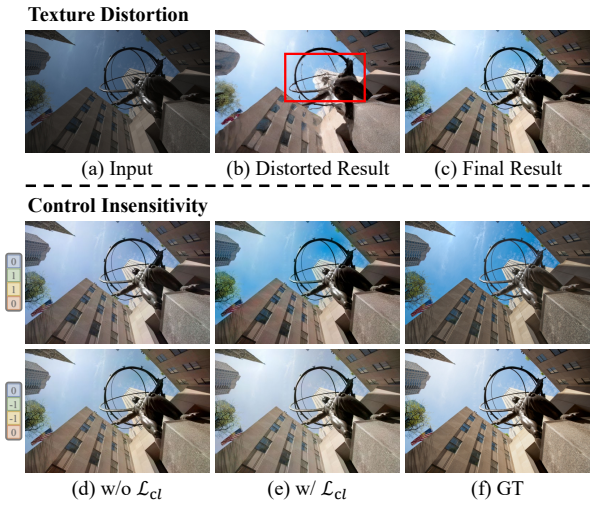

Figure 3: Examples of **Texture Distortion** and **Control Insensitivity**. Top row: (a) Input image; (b) and (c) are the results generated w/o and w/ the affine bilateral grid. Bottom two rows: (d) and (e) are the results generated by the model w/o and w/ $\mathcal{L}_{cl}$; (f) are GT retouched by two experts. Condition $\mathbf{c}$ is shown on the left, where the adjusted attributes are contrast and color temperature. With $\mathcal{L}_{cl}$ (Eq. (6)), the region of the sky is closer to the expert-retouched results.

Training Strategy

Reconstruction Supervision Stable Diffusion performs the standard DDPM training strategy in the latent space. More specifically, with the time step t randomly sampled from a uniform distribution, the noisy latent $\mathbf{Z}_t$ at the timestamp t can be obtained by

$$\mathbf{Z}_t = \sqrt{\bar{\alpha}_t}\mathbf{Z}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \tag{3}$$

where $\sqrt{1-\bar{\alpha}_t}$ is the noise schedule of the diffusion model. Along with resized image $\mathbf{R}'$ and the condition $\mathbf{c}$ as input, the denoising model is trained to predict the noise $\boldsymbol{\epsilon}$ according to Eq. (1). Since the clean latent $\mathbf{Z}_0$ can be estimated with the predicted noise $\boldsymbol{\epsilon}_{pred,t}$, the supervision between $\boldsymbol{\epsilon}$ and $\boldsymbol{\epsilon}_{pred,t}$ can be viewed as the reconstruction supervision in the latent space.

In our DiffRetouch, the role of the denoising model is to generate the noise estimation and the affine bilateral grid simultaneously, which is indicated by Eq. (2). For the training of the affine bilateral grid, we also impose the reconstruction supervision on the pixel-space output $\mathbf{D}_t$. We collectively refer to these two supervisions in both the latent ($\mathbf{Z}$) and pixel ($\mathbf{D}$) space as reconstruction supervision.

Contrastive Learning Before introducing our contrastive learning, a brief description of the construction of image-condition pairs is needed. Current datasets for retouching consist of results retouched by different experts. However, they lack a description of the retouching style for each result. To provide the users with an understandable editing mechanism, we describe the retouching style from the aspects of four image attributes. To construct such image-condition pairs, commonly used measurements (more details are in the supp.) are utilized to calculate the score for each attribute,

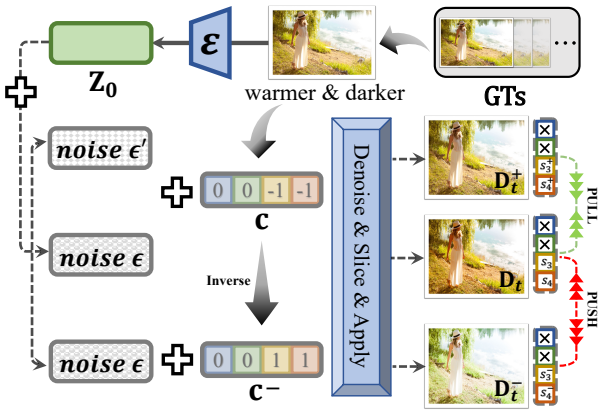

Figure 4: Framework of contrastive learning scheme. The regular branch takes the latent $\mathbf{Z}_0$, the noise map $\boldsymbol{\epsilon}$, and the condition $\mathbf{c}$ as input to generate the result $\mathbf{D}_t$. Another two branches produce the positive sample $\mathbf{D}_t^+$ with a different noise map $\boldsymbol{\epsilon}'$ and the same condition $\mathbf{c}$, and negative samples $\mathbf{D}_t^-$ with the same $\boldsymbol{\epsilon}$ and the opposite condition $\mathbf{c}^-$. For coefficients $\mathbf{c}_i \neq 0$, $\mathcal{L}_{cl}$ (Eq. (6)) steers the corresponding $\mathbf{s}_i$ closer to $\mathbf{s}_i^+$ and away from $\mathbf{s}_i^-$. In this example, the adjusted attributes are color temperature and brightness.

which can be denoted as $\mathbf{s} = [\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, \mathbf{s}_4]$. For each low-quality input $\mathbf{R}$, the dataset provides several ground truth (GT) retouched by different experts. Among these GTs for the same $\mathbf{R}$, we first calculate their $\mathbf{s}$ and initialize their $\mathbf{c}$ as 0. For each attribute (i), the corresponding coefficient $\mathbf{c}_i$ is set to 1(-1) for the GT with the highest(lowest) $\mathbf{s}_i$. That is to say, for the most colorful, highest contrast, coolest, and brightest GT among the GTs for the same $\mathbf{R}$, the corresponding coefficient $\mathbf{c}_i$ will be set to 1, and vice versa.

Although training with such image-condition pairs under the reconstruction supervision enables the model to sample various styles according to $\mathbf{c}$, the influence caused by adjusting these coefficients tends to be weak. As shown in Figure 3(d), adjusting the coefficients related to contrast and color temperature has little effect on the final result. In order to encourage our DiffRetouch more aware of the adjustment brought by each attribute, we design the contrastive learning scheme, which involves explicit supervision w.r.t. these attributes. An example is shown in Figure 4.

More specifically, apart from the regular branch which takes the latent $\mathbf{Z}_0$, the noise map $\boldsymbol{\epsilon}$, and the condition $\mathbf{c}$ as input and operates as Eq. (2), another two branches are included to produce the positive and negative samples. The positive branch shares the same $\mathbf{c}$, but adopt another noisy map $\boldsymbol{\epsilon}'$ as input. The negative branch still utilizes $\boldsymbol{\epsilon}$ but with the opposite condition $\mathbf{c}^- = -\mathbf{c}$. After the operation of slicing and applying, three results in the pixel space can be obtained, then we calculate their scores w.r.t. the four attributes, which can be denoted as $\mathbf{s}, \mathbf{s}^+$, and $\mathbf{s}^-$ respectively. For the attribute whose coefficient is not 0, we steer the corresponding score $\mathbf{s}_i$ closer to $\mathbf{s}_i^+$ and away from $\mathbf{s}_i^-$. The effect of the contrastive learning scheme is shown in Figure 3. Equipped with contrastive learning, the influence brought by the condition input is more sensitive and the re-

Method	PSNR↑/SSIM↑						FID↓	NIMA↑
	A	B	C	D	E	Average		
PIENet†	**21.54/0.882**	**26.02**/0.948	25.29/0.919	22.95/0.905	24.22/0.925	24.00/0.916	14.507	4.338
TSFlow†	20.65/0.869	25.34/**0.952**	25.57/0.935	22.48/0.913	23.65/0.935	23.54/0.921	9.678	4.976
StarEnhancer†	20.75/0.880	25.84/**0.952**	**25.73/0.937**	**23.50/0.922**	**24.60/0.947**	**24.09/0.928**	**9.493**	**4.977**
DiffRetouch†	**24.48/0.936**	**26.12/0.958**	**26.21/0.944**	**24.51/0.940**	**24.67/0.953**	**25.20/0.946**	**8.957**	**5.022**

Table 1: Quantitative comparison on MIT-Adobe FiveK dataset with subsets retouched by five experts (A/B/C/D/E). Symbol † represents the model trained with the mixture of five subsets. The best result is in **red** whereas the second is in **blue**. The evaluations are done on the 340p setting.

sults are closer to that retouched by experts.

So far, the overall training objective can be written as

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda \mathcal{L}_{cl}. \tag{4}$$

The reconstruction supervisions are imposed on both the latent and pixel space, which can be formulated as

$$\mathcal{L}_{rec} = \|\boldsymbol{\epsilon}_{pred,t} - \boldsymbol{\epsilon}\|^2 + \beta \|\mathbf{D}_t - \mathbf{X}_0\|^2, \tag{5}$$

where λ and β are the scalars. Inspired by InfoNCE (Oord, Li, and Vinyals 2018), the loss function of the contrastive learning is defined as

$$\mathcal{L}_{cl} = \sum_{\{i|\mathbf{c}_i \neq 0\}} -\log \frac{e^{-|\mathbf{s}_i - \mathbf{s}_i^+|/\tau}}{e^{-|\mathbf{s}_i - \mathbf{s}_i^+|/\tau} + e^{-|\mathbf{s}_i - \mathbf{s}_i^-|/\tau}}, \tag{6}$$

where τ is the temperature parameter. $\mathbf{s}_i$, $\mathbf{s}_i^+$, and $\mathbf{s}_i^-$ are the scores of the pixel space output $\mathbf{D}_t$, $\mathbf{D}_t^+$, and $\mathbf{D}_t^-$ w.r.t. the corresponding attributes.

Experiments

Settings

Datasets. Our experiments are conducted on the MIT-Adobe FiveK dataset (Bychkovsky et al. 2011) and the PPR10K dataset (Liang et al. 2021). The MIT-Adobe FiveK dataset contains 5,000 RAW images. For each RAW image, five reference images retouched by different experts (A/B/C/D/E) are provided. We follow the pre-processing pipeline in (Song, Qian, and Du 2021; Wang et al. 2019), and split the dataset into 4,500 pairs for training and 500 pairs for validation, which is also known as MIT-Adobe-5K-UPE. Both 340p (short side of the images) and full resolution are used for validation. The PPR10K dataset (Liang et al. 2021) contains 11,161 portrait photos with 1,681 groups, and each photo has three retouched versions processed by three experts (A/B/C). Following (Liang et al. 2021), we divide the PPR10K dataset into a training set with 1,356 groups and 8,875 photos, and a testing set with 325 groups and 2,286 photos. We employ the 360p setting in (Liang et al. 2021). For both datasets, we construct the image-condition pairs for each image following the practice introduced above.

Evaluation Metrics. To align with previous methods (Song, Qian, and Du 2021; Ouyang et al. 2023), we use the PSNR, SSIM, and LPIPS (Zhang et al. 2018) as the evaluation metrics. To evaluate the similarity between the distribution of the results and that of expert-retouched GTs, perceptual metric FID (Heusel et al. 2017) is employed. We also adopt the no-reference metric NIMA (Talebi and Milanfar 2018) to evaluate the results from an aesthetic perspective.

Comparison with Other Methods

We compare our DiffRetouch with other methods proposed for retouching. Based on the ability to generate diverse retouched results, we classify these methods into two categories. The first type of method adopts deterministic models, and can only produce a single retouching style for each input, which includes HDRNet (Gharbi et al. 2017), DeepUPE (Wang et al. 2019), CURL (Moran, McDonagh, and Slabaugh 2021), DeepLPF (Moran et al. 2020), 3DLUT (Zeng et al. 2020), CSRNet (He et al. 2020), and RSFNet (Ouyang et al. 2023). Following the default setting, these methods train multiple models for subsets retouched by different experts, and evaluate separately on each subset. The other type of methods, like PIENet (Kim, Koh, and Kim 2020), StarEnhancer (Song, Qian, and Du 2021), and TSFlow (Wang et al. 2022), supports generating multiple retouching styles. These methods adopt one model trained with results retouched by different experts. During inference, to generate the corresponding predictions for each expert, style embedding (Song, Qian, and Du 2021; Wang et al. 2022), which represents the overall retouching style of the expert, is extracted from additional images and sent into their models. Thanks to the superiority of our editing mechanism, we can generate the desired retouching style by adjusting condition $\mathbf{c}$, without the need for additional images. To evaluate the multi-style retouching performance, we provide the model with the pre-calculated condition of GT styles, thus making it produce corresponding output for evaluation. For each input image, we follow the practice in 'Sec. Contrastive Learning' to construct $\mathbf{c}$ for different experts.

For MIT-Adobe FiveK, the qualitative and quantitative comparisons are shown in Figure 5, Table 1, and Table 2. Table 2 shows the comparison against the state-of-the-art methods evaluated on the Expert-C subset. For methods that support various styles, the performance of generating retouching styles of five experts are shown in Table 1. We also compare our DiffRetouch with HDRNet (Gharbi et al. 2017), CSRNet (He et al. 2020), RSFNet (Ouyang et al. 2023), and 3DLUT (Zeng et al. 2020) on PPR10K. Note that for subsets of three experts(A/B/C), these methods are evaluated with three models trained on three subsets respectively, while our DiffRetouch is a single model trained with a mixture of subsets. As can be seen in Table 3, despite using only one model against three models of other methods, our method achieves better or comparable performance on referenced and non-referenced evaluation metrics. The visual results on PPR10K datasets are shown in Figure 6.

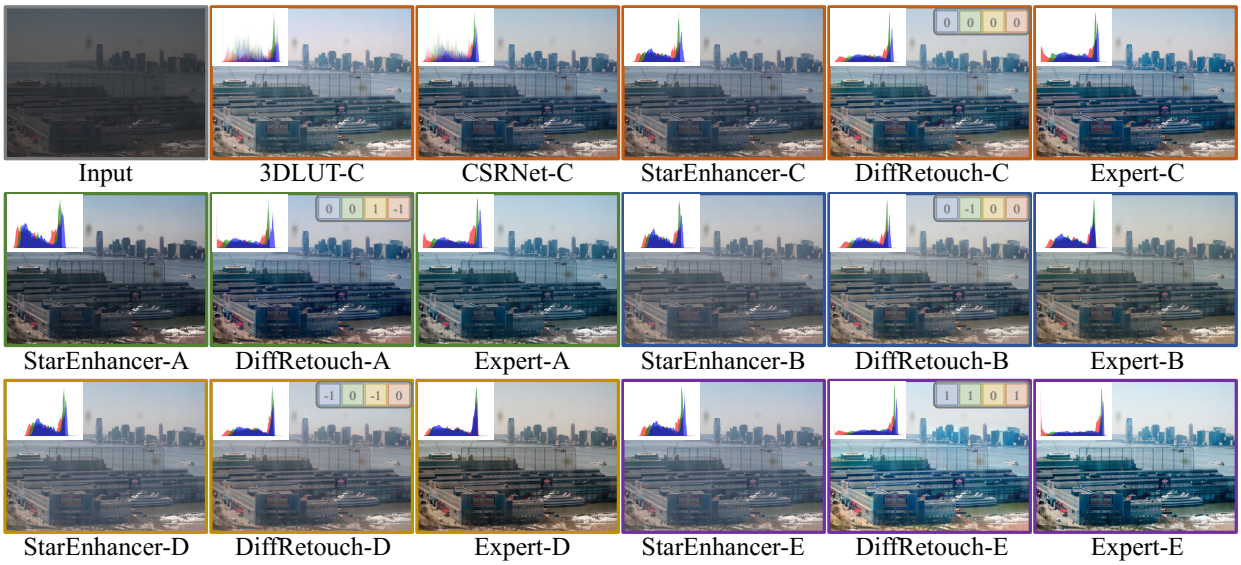

Figure 5: Qualitative comparison on MIT-Adobe FiveK dataset with subsets retouched by five experts (A/B/C/D/E). Since 3D-LUT and CSRNet are unable to produce multiple retouching styles, only the results corresponding to Expert-C are displayed. Condition **c** is shown at the top of each DiffRetouch generated result. Results generated by our DiffRetouch are more similar to the corresponding expert-retouched result, especially for the color histogram.

Method	340p			Full Resolution		
	PSNR↑	SSIM↑	LPIPS↓	PSNR ↑	SSIM ↑	LPIPS↓
HDRNet	23.71	0.899	0.080	23.20*	0.917*	0.120*
DeepUPE	23.48	0.907	0.085	23.24*	0.893*	0.158*
CURL	24.40	0.935	0.061	24.20*	0.880*	0.108*
DeepLPF	24.43	**0.937**	0.059	24.48*	0.887*	0.103*
3DLUT	25.07	**0.937**	0.055	24.92*	0.934*	0.093*
CSRNet	25.55	0.936	0.057	25.06*	0.935*	0.090*
RSFNet	25.34	0.938	**0.051**	25.09	0.915	**0.081**
PIENet†	25.29	0.920	0.099	-	-	-
TSFlow†	25.57	0.935	0.055	**25.36***	**0.944***	**0.079***
StarEnhancer†	**25.73**	**0.937**	0.055	25.29*	0.943*	0.086*
DiffRetouch†	**26.21**	**0.944**	**0.054**	**25.41**	**0.952**	0.088

Table 2: Quantitative comparison on MIT-Adobe FiveK with Expert-C subset. † represents trained with the mixture of five subsets. * represents these results are replicated from StarEnhancer. The best is in **red** whereas the second is in **blue**. Both 340p (the shorter edge of the image) and full resolution are used for validation. The training code of PIENet is not yet accessible, and only the 340p results are released.

Method	E	PSNR↑	SSIM↑	LPIPS↓	FID↓	NIMA↑
HDRNet§	A	23.01	0.953	0.057		
	B	23.17	0.952	0.058	2.782	5.490
	C	23.34	0.951	0.058		
CSRNet§	A	23.86	0.952	0.055		
	B	23.70	0.952	0.057	3.077	5.465
	C	23.87	0.953	0.055		
3DLUT§	A	**25.98**	**0.967**	**0.040**		
	B	**25.06**	**0.959**	**0.046**	**2.485**	**5.492**
	C	**25.45**	**0.961**	**0.045**		
DiffRetouch†	A	**26.23**	**0.970**	**0.040**		
	B	**25.65**	**0.969**	**0.042**	**2.664**	**5.517**
	C	**25.67**	**0.966**	**0.043**		

Table 3: Quantitative comparisons on PPR10K dataset with subsets retouched by three experts (A/B/C). The best result is in **red** whereas the second is in **blue**. E is short for Expert. § means training three models with three subsets respectively. † means one model trained with the mixture of three subsets. Despite using only one model, our DiffRetouch achieves better or comparable performance against other methods that require three models.

Ablation Study

As shown in Table 4, we carry out the ablation study to demonstrate the necessity of each design.

Introduction of affine bilateral grid. Our intention of introducing the affine bilateral grid is to solve the texture distortion existing in the baseline model. Motivated by StableSR (Wang et al. 2023a), the controllable feature wrapping (CFW) module is adopted, which extracts the intermediate features from the encoder to modulate the decoder features. As shown in Figure 7(c), the severe texture distortion can be somewhat alleviated. However, it cannot handle complex structures like human faces. With the introduction of the affine bilateral grid, this problem can be well solved and original details are reserved, which is shown in Figure 7(d). The improvements in Table 4 are significant, which demonstrates the effectiveness of the affine bilateral grid.

Contrastive learning scheme. As shown in Figure 7(d), the influence brought by adjusting the attributes is too weak to reach the retouching style of experts in Figure 7(e). With the implementation of a contrastive learning scheme, our DiffRetouch is encouraged to be more aware of the adjustment brought by each attribute and outputs results more like experts especially for the color temperature, which is shown

Figure 6: Qualitative comparison on PPR10K dataset with subsets retouched by three experts (A/B/C). The input condition $\mathbf{c}$ is shown at the top of each DiffRetouch generated result. Results generated by our DiffRetouch are more similar to the corresponding expert-retouched result.

Figure 7: Ablation study of our model. 'A' represents the baseline model. 'B_1' and 'B_2' represent the baseline model equipped with CFW and affine bilateral grid, but both w/o $\mathcal{L}_{cl}$. (g)-(j) show intermediate retouching styles generated by our full model w/ $\mathcal{L}_{cl}$ through interpolating the coefficients.

in Figure 7(j). To better show the effectiveness of contrastive learning, we attempt to quantify the influence brought by adjusting each attribute. For each attribute (i), we set the corresponding $\mathbf{c}_i$ to the maximum (1) and minimum (-1) values with other three remaining 0, and generate two extreme results. By calculating the difference in corresponding scores ($\mathbf{s}_i$) of these two results, the adjustable range w.r.t. this attribute can be obtained. The average adjustable ranges for Adobe5K before and after the implementation of contrastive learning are shown in Table 5, and we can see that the adjustable ranges are enlarged for four attributes. The improvements of FID for model B_2 and C in Table 4 further prove that the contrastive learning helps the model better fit the expert-retouched distribution.

Adjustable condition. Our editing mechanism allows users to customize their preferred style by adjusting the coefficients w.r.t. four attributes. Figure 7(g)-(j) show an example of generating the intermediate styles between the style of Expert B and C. As the coefficients are adjusted, we can see that the style changes continuously and in the correct direction for the corresponding attribute.

Model	Adobe5K			PPR10K		
	PSNR↑	SSIM↑	FID↓	PSNR↑	SSIM↑	FID↓
A	19.83	0.618	50.157	20.47	0.694	16.673
B_1	20.97	0.885	26.721	21.79	0.831	14.511
B_2	23.70	0.936	11.476	25.15	0.964	3.091
C	25.20	0.946	8.957	25.85	0.969	2.499

Table 4: Quantitative results of our model and its variants on Adobe5K and PPR10K datasets. 'A' represents the baseline model. 'B_1' and 'B_2' represent the baseline model equipped with CFW and affine bilateral grid, but both w/o $\mathcal{L}_{cl}$. 'C' represents the full model w/ $\mathcal{L}_{cl}$.

Model	Colorfulness↑	Contrast↑	Color Temperature↑	Brightness↑
B_2	10.12	252.7	1141.5	26.46
C	20.60	297.3	1572.9	35.22

Table 5: Comparison of adjustable range w/ and w/o $\mathcal{L}_{cl}$. 'B_2' and 'C' represent the full model w/ and w/o $\mathcal{L}_{cl}$. The adjustable range for each attribute (i) is determined by the changes in the corresponding scores ($\mathbf{s}_i$) of results when varying the coefficient ($\mathbf{c}_i$) from maximum (1) to minimum (-1). A larger adjustable range means more significant influence brought by adjusting the coefficient, proving the effectiveness of the contrastive learning scheme.

Conclusion

In this paper, we propose a diffusion-based method for image retouching, called DiffRetouch. Considering the subjectivity of this task, we leverage the excellent distribution coverage of diffusion to capture the fine-retouched distribution, allowing to sample various visual-pleasing styles. Four adjustable coefficients are provided for users to edit the final results. The affine bilateral grid and contrastive learning scheme are introduced to address the texture distortion and control insensitivity. Extensive experiments have demonstrated the superiority of our DiffRetouch.

Acknowledgments

This work is funded by the National Natural Science Foundation of China (62306153), the Fundamental Research Funds for the Central Universities (Nankai University, 070-63243143), the China Postdoctoral Science Foundation (GZB20240357, 2024M761682), and Shui Mu Tsinghua Scholar (2024SM079). The computational devices of this work is supported by the Supercomputing Center of Nankai University (NKSC).

References

Bychkovsky, V.; Paris, S.; Chan, E.; and Durand, F. 2011. Learning photographic global tonal adjustment with a database of input/output image pairs. In *CVPR*.

Chen, C.; Chen, Q.; Xu, J.; and Koltun, V. 2018a. Learning to see in the dark. In *CVPR*.

Chen, Y.-S.; Wang, Y.-C.; Kao, M.-H.; and Chuang, Y.-Y. 2018b. Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans. In *CVPR*.

Chen, Z.; Zhang, Y.; Gu, J.; Yuan, X.; Kong, L.; Chen, G.; and Yang, X. 2023. Image Super-Resolution with Text Prompt Diffusion. *arXiv preprint arXiv:2311.14282*.

Chung, H.; Sim, B.; Ryu, D.; and Ye, J. C. 2022. Improving diffusion models for inverse problems using manifold constraints. In *NeurIPS*.

Gharbi, M.; Chen, J.; Barron, J. T.; Hasinoff, S. W.; and Durand, F. 2017. Deep bilateral learning for real-time image enhancement. *ACM TOG*.

He, J.; Liu, Y.; Qiao, Y.; and Dong, C. 2020. Conditional sequential modulation for efficient global image retouching. In *ECCV*.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. In *NeurIPS*.

Hou, J.; Zhu, Z.; Hou, J.; Liu, H.; Zeng, H.; and Yuan, H. 2023. Global Structure-Aware Diffusion Process for Low-Light Image Enhancement. *arXiv preprint arXiv:2310.17577*.

Hu, Y.; He, H.; Xu, C.; Wang, B.; and Lin, S. 2018. Exposure: A white-box photo post-processing framework. *ACM TOG*.

Jiang, Y.; Zhang, Z.; Xue, T.; and Gu, J. 2023. AutoDIR: Automatic All-in-One Image Restoration with Latent Diffusion. *arXiv preprint arXiv:2310.10123*.

Kim, H.; Choi, S.-M.; Kim, C.-S.; and Koh, Y. J. 2021. Representative color transform for image enhancement. In *ICCV*.

Kim, H.; and Lee, K. M. 2023. Learning Controllable ISP for Image Enhancement. *IEEE TIP*.

Kim, H.-U.; Koh, Y. J.; and Kim, C.-S. 2020. PieNet: Personalized image enhancement network. In *ECCV*.

Kosugi, S.; and Yamasaki, T. 2020. Unpaired image enhancement featuring reinforcement-learning-controlled image editing software. In *AAAI*.

Li, C.; Guo, C.; Ai, Q.; Zhou, S.; and Loy, C. C. 2020. Flexible piecewise curves estimation for photo enhancement. *arXiv preprint arXiv:2010.13412*.

Li, C.; Guo, C.; Feng, R.; Zhou, S.; and Loy, C. C. 2022. CuDi: Curve Distillation for Efficient and Controllable Exposure Adjustment. *arXiv preprint arXiv:2207.14273*.

Li, C.; Guo, C.; and Loy, C. C. 2021. Learning to enhance low-light image via zero-reference deep curve estimation. *IEEE TPAMI*.

Liang, J.; Zeng, H.; Cui, M.; Xie, X.; and Zhang, L. 2021. Ppr10k: A large-scale portrait photo retouching dataset with human-region mask and group-level consistency. In *CVPR*.

Liu, R.; Ma, L.; Zhang, J.; Fan, X.; and Luo, Z. 2021. Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement. In *CVPR*.

Lugmayr, A.; Danelljan, M.; Romero, A.; Yu, F.; Timofte, R.; and Van Gool, L. 2022. Repaint: Inpainting using denoising diffusion probabilistic models. In *CVPR*.

Lv, X.; Zhang, S.; Wang, C.; Zheng, Y.; Zhong, B.; Li, C.; and Nie, L. 2024. Fourier Priors-Guided Diffusion for Zero-Shot Joint Low-Light Enhancement and Deblurring. In *CVPR*.

Moran, S.; Marza, P.; McDonagh, S.; Parisot, S.; and Slabaugh, G. 2020. Deeplpf: Deep local parametric filters for image enhancement. In *CVPR*.

Moran, S.; McDonagh, S.; and Slabaugh, G. 2021. Curl: Neural curve layers for global image enhancement. In *ICPR*.

Nichol, A. Q.; and Dhariwal, P. 2021. Improved denoising diffusion probabilistic models. In *ICML*.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Ouyang, W.; Dong, Y.; Kang, X.; Ren, P.; Xu, X.; and Xie, X. 2023. RSFNet: A White-Box Image Retouching Approach using Region-Specific Color Filters. In *ICCV*.

Özdenizci, O.; and Legenstein, R. 2023. Restoring vision in adverse weather conditions with patch-based denoising diffusion models. *IEEE TPAMI*.

Ren, M.; Delbracio, M.; Talebi, H.; Gerig, G.; and Milanfar, P. 2023. Multiscale Structure Guided Diffusion for Image Deblurring. In *ICCV*.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *CVPR*.

Saharia, C.; Ho, J.; Chan, W.; Salimans, T.; Fleet, D. J.; and Norouzi, M. 2022. Image super-resolution via iterative refinement. *IEEE TPAMI*.

Shin, S.; Shin, J.; Bae, J.; Shim, I.; and Jeon, H.-G. 2024. Close Imitation of Expert Retouching for Black-and-White Photography. In *CVPR*.

Song, Y.; Qian, H.; and Du, X. 2021. Starenhancer: Learning real-time and style-aware image enhancement. In *ICCV*.

Sun, H.; Li, W.; Liu, J.; Chen, H.; Pei, R.; Zou, X.; Yan, Y.; and Yang, Y. 2023. CoSeR: Bridging Image and Language for Cognitive Super-Resolution. *arXiv preprint arXiv:2311.16512*.

Sun, X.; Li, M.; He, T.; and Fan, L. 2021. Enhance images as you like with unpaired learning. *arXiv preprint arXiv:2110.01161*.

Talebi, H.; and Milanfar, P. 2018. NIMA: Neural image assessment. *IEEE TIP*.

Tseng, E.; Zhang, Y.; Jebe, L.; Zhang, C.; Xia, Z.; Fan, Y.; Heide, F.; and Chen, J. 2022. Neural Photo-Finishing. *ACM TOG*.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*.

Wang, H.; Zhang, J.; Liu, M.; Wu, X.; and Zuo, W. 2022. Learning Diverse Tone Styles for Image Retouching. *arXiv preprint arXiv:2207.05430*.

Wang, J.; Chan, K. C.; and Loy, C. C. 2023. Exploring clip for assessing the look and feel of images. In *AAAI*.

Wang, J.; Yue, Z.; Zhou, S.; Chan, K. C.; and Loy, C. C. 2023a. Exploiting Diffusion Prior for Real-World Image Super-Resolution. *arXiv preprint arXiv:2305.07015*.

Wang, R.; Zhang, Q.; Fu, C.-W.; Shen, X.; Zheng, W.-S.; and Jia, J. 2019. Underexposed photo enhancement using deep illumination estimation. In *CVPR*.

Wang, Y.; Yu, Y.; Yang, W.; Guo, L.; Chau, L.-P.; Kot, A. C.; and Wen, B. 2023b. Exposurediffusion: Learning to expose for low-light image enhancement. In *ICCV*.

Whang, J.; Delbracio, M.; Talebi, H.; Saharia, C.; Dimakis, A. G.; and Milanfar, P. 2022. Deblurring via stochastic refinement. In *CVPR*.

Wu, R.; Yang, T.; Sun, L.; Zhang, Z.; Li, S.; and Zhang, L. 2023. SeeSR: Towards Semantics-Aware Real-World Image Super-Resolution. *arXiv preprint arXiv:2311.16518*.

Xie, S.; Zhang, Z.; Lin, Z.; Hinz, T.; and Zhang, K. 2023. Smartbrush: Text and shape guided object inpainting with diffusion model. In *CVPR*.

Yang, T.; Ren, P.; Xie, X.; and Zhang, L. 2023. Pixel-aware stable diffusion for realistic image super-resolution and personalized stylization. *arXiv preprint arXiv:2308.14469*.

Yin, Y.; Xu, D.; Tan, C.; Liu, P.; Zhao, Y.; and Wei, Y. 2023. CLE Diffusion: Controllable Light Enhancement Diffusion Model. In *ACM MM*.

Yu, F.; Gu, J.; Li, Z.; Hu, J.; Kong, X.; Wang, X.; He, J.; Qiao, Y.; and Dong, C. 2024. Scaling Up to Excellence: Practicing Model Scaling for Photo-Realistic Image Restoration In the Wild. *arXiv preprint arXiv:2401.13627*.

Zeng, H.; Cai, J.; Li, L.; Cao, Z.; and Zhang, L. 2020. Learning image-adaptive 3d lookup tables for high performance photo enhancement in real-time. *IEEE TPAMI*.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*.

Zhu, A.; Zhang, L.; Shen, Y.; Ma, Y.; Zhao, S.; and Zhou, Y. 2020. Zero-shot restoration of underexposed images via robust retinex decomposition. In *ICME*.