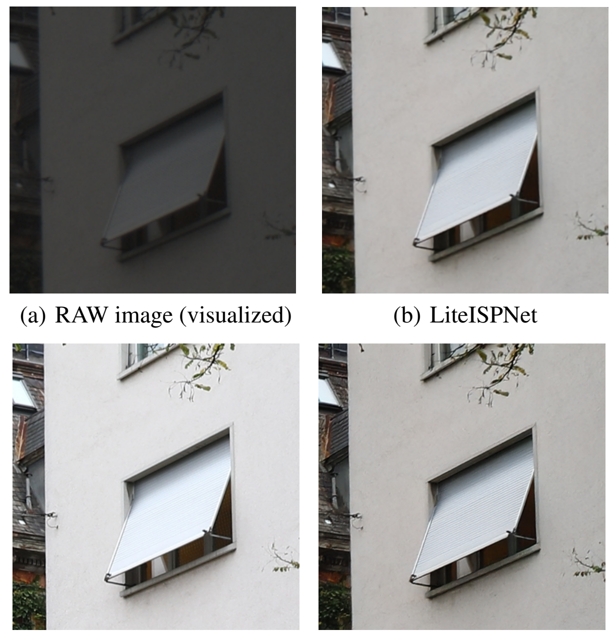
DiffRAW: Leveraging Diffusion Model to Generate DSLR-Comparable Perceptual Quality sRGB from Smartphone RAW Images

Mingxin Yi[1], **Kai Zhang**[1,3*], **Pei Liu**[2], **Tanli Zuo**[2], **Jingduo Tian**[2*]
[1]Tsinghua Shenzhen International Graduate School, Tsinghua University, China
[2]Media Technology Lab, Huawei, China
[3]Research Institute of Tsinghua, Pearl River Delta
ymx21@mails.tsinghua.edu.cn, zhangkai@sz.tsinghua.edu.cn, {liupei55,zuotanli,tianjingduo}@huawei.com

Abstract

Deriving DSLR-quality sRGB images from smartphone RAW images has become a compelling challenge due to discernible detail disparity, color mapping instability, and spatial misalignment in RAW-sRGB data pairs. We present DiffRAW, a novel method that incorporates the diffusion model for the first time in learning RAW-to-sRGB mappings. By leveraging the diffusion model, our approach effectively learns the high-quality detail distribution of DSLR images, thereby enhancing the details of output images. Simultaneously, we use the RAW image as a diffusion condition to maintain image structure information such as contours and textures. To mitigate the interference caused by the color and spatial misalignment in training data pairs, we embed a color-position preserving condition within DiffRAW, ensuring that the output images do not exhibit color biases and pixel shift issues. To accelerate the inference process of DiffRAW, we designed the Domain Transform Diffusion Method, an efficient diffusion process with its corresponding reverse process. The Domain Transform Diffusion Method can reduce the required inference steps for diffusion model-based image restoration/enhancement algorithms while enhancing the quality of the generated images. Through evaluations on the ZRR dataset, DiffRAW consistently demonstrates state-of-the-art performance across all perceptual quality metrics (e.g., LPIPS, FID, MUSIQ), while achieving comparable results in PSNR and SSIM.



(a) RAW image (visualized) (b) LiteISPNet

(c) DSLR sRGB image (d) DiffRAW(ours)

Figure 1: Comparison of results on ZRR dataset. The images generated by our method exhibit richer detail and higher clarity, rivaling the visual quality of DSLR images.

Introduction

To extract natural sRGB images from RAW sensor images, a meticulously engineered image signal processing (ISP) pipeline is usually needed. This encompasses a range of manually crafted low-level vision operations such as demosaicking, white balance, color correction, denoising, gamma correction, among others (Ramanath et al. 2005).

With the rapid advancement of mobile photography, smartphones have become the primary devices for photo capture, owing to their portability. However, due to hardware constraints of smartphone cameras, such as the size of the aperture and sensor, images captured by smartphones exhibit a significant quality gap compared to those taken with professional DSLR cameras. To address this issue, the academic community has begun to explore end-to-end ISP algorithm research based on smartphone RAW to DSLR sRGB data pairs (Ignatov, Van Gool, and Timofte 2020; Liang et al. 2021; Schwartz, Giryes, and Bronstein 2018).

To convert smartphone RAW images into DSLR-quality sRGB images, there are three challenges: First, the inherent hardware constraints of smartphones induce a loss of detail in RAW images relative to DSLR sRGB counterparts, making the task of fully reconstructing DSLR sRGB imagery from smartphone RAW an ill-posed problem. Second, the collection of smartphone RAW images and DSLR sRGB images from different devices inevitably leads to a non-precise alignment problem within the data pairs. Third, as the data pairs are collected under varying environmental conditions

*Kai Zhang and Jingduo Tian are the corresponding authors of this paper.

and camera parameters, the RAW and sRGB images manifest not only color disparities but also an unstable color mapping relationship.

In response to these challenges, we propose the DiffRAW model, which incorporates the diffusion model for the first time in learning RAW-to-sRGB mappings. To address the significant detail disparity between RAW-sRGB data pairs, we leverage the diffusion model to learn the high-quality detail distribution of DSLR images, while using the RAW image as a diffusion condition to retain the structural information (such as contours and textures) of the generated images without relying on the RAW image for details. This combined strategy allows DiffRAW's generated results to maintain the overall image structure of the smartphone RAW image while possessing DSLR-comparable details.

For spatial misalignment and unstable color mapping relationship in the data pairs, we embed a color-position preserving condition in DiffRAW to ensure that the output images do not exhibit color biases and pixel shift issues. This condition also allows for flexible color style transfer.

Moreover, to address the high iteration step issue in the diffusion model's inference process, DiffRAW designs an efficient forward and reverse diffusion process, named the Domain Transform Diffusion Method, which reduces the required iteration steps during the inference phase while enhancing the quality of the generated images. In essence, the primary contributions of our research are as follows:

- We propose a novel and efficient forward and reverse process, named the Domain Transform Diffusion Method, which reduces the iteration steps required during the inference stage while enhancing the quality of the generated images. The Domain Transform Diffusion Method is a universal acceleration approach specifically designed for diffusion model-based image restoration/enhancement algorithms, and can be flexibly transferred to other Diffusion-based image enhancement/restoration algorithms for inference acceleration.
- We introduce the diffusion model into the task of learning RAW-to-sRGB mapping for the first time, proposing the DiffRAW model, achieving state-of-the-art results in perceptual quality metrics.
- We use RAW images as the diffusion condition for the first time, retaining structural information like texture and contours in the generated images.
- Through the specially designed color-position preserving condition, we alleviate the training interference caused by color and spatial misalignment in the training data pairs, ensuring that the model's generated results do not produce color biases and pixel shifts.
- DiffRAW possesses a color pluggable feature. Using different colors of color-position preserving condition for color information injection allows for flexible adjustment of the generated images' color style.

Preliminary

As our approach belongs to the diffusion-based model, we will provide a brief introduction to the background of the diffusion model in this section.

Diffusion Model

The diffusion model includes the forward process and the reverse process. The forward process refers to the procedure of adding noise to the image. Given a real image $y_0 \sim q(y)$, the forward process of the diffusion model accumulates noise through T steps, resulting in $y_1, y_2, y_3, \cdots, y_T$.

Given the variance hyperparameters of the Gaussian noise distribution in the T steps of the noise process $\{\beta_t \in (0,1)\}_{t=1}^{T}$, the definition of the noisy image sequence $y_1, y_2, y_3, \cdots, y_T$ can be given by the following formula:

$$q(y_t|y_{t-1}) = \mathcal{N}(y_t; \sqrt{1-\beta_t}y_{t-1}, \beta_t I) \tag{1}$$

Letting $\alpha_t = 1 - \beta_t$, $\overline{\alpha}_t = \Pi_{i=1}^{t}\alpha_i$, and through derivation, the forward process can be expressed as:

$$q(y_t|y_0) = \mathcal{N}(y_t; \sqrt{\overline{\alpha}_t}y_0, (1-\overline{\alpha}_t)I) \tag{2}$$

In the forward process, $\overline{\alpha}_t$ is a monotonically decreasing sequence, usually pinned to $\overline{\alpha}_0 \approx 1$ and $\overline{\alpha}_T \approx 0$. Thus, as t increases, y_t approaches pure noise. When $T \to \infty$, y_T is complete Gaussian noise.

Next, we will briefly introduce the training process of the diffusion model: first obtain the input image $y_0 \sim q(y)$, randomly select $t \sim Uniform(\{1, 2, 3, \cdots, T\})$, sample a random noise $\epsilon \sim \mathcal{N}(0, I)$, and from Equation 2 it is known that $y_t = \sqrt{\overline{\alpha}_t}y_0 + \sqrt{1-\overline{\alpha}_t}\epsilon$. Using a U-Net(Ronneberger, Fischer, and Brox 2015) network $f_\theta(y_t, t)$ to predict noise ϵ, thereby restoring the noisy image y_t to the original image y_0. Ho et al.(Ho, Jain, and Abbeel 2020) showed that a loss function that works well in practice is a reweighted evidence lower bound (Kingma and Welling 2013):

$$L(\theta) = \mathbb{E}_{y_0,t,\epsilon}\|f_\theta(y_t,t) - \epsilon\|^2 \tag{3}$$

Here, θ represents the learnable parameters of the U-Net network $f_\theta(y_t, t)$ (Ho, Jain, and Abbeel 2020).

The reverse process is the denoising inference process of the diffusion model. The model progressively generates images by reversing the forward process. After the training stage is over, we take the moment of maximum noise strength T as the starting point for the reverse process, sampling $y_T \sim \mathcal{N}(0, I)$ from the standard Gaussian distribution, and use y_T as the generation starting point, iteratively inferring $y_{T-1}, y_{T-2}, y_{T-3}, \cdots$.

Specifically, for any moment t and the current moment's noisy image y_t, the noisy image y_{t-1} at the moment $t-1$ can be inferred using the Bayesian formula(Ho, Jain, and Abbeel 2020):

$$p_\theta(y_{t-1}|y_t) = \mathcal{N}(y_{t-1}; \mu_\theta(y_t, t), \sigma_t^2 I) \tag{4}$$

Here, σ_t is usually a pre-defined constant related to the variance schedule $\{\beta_t \in (0,1)\}_{t=1}^{T}$, and $\mu_\theta(y_t, t)$ can be estimated using the trained denoising network $f_\theta(y_t, t)$ through the following formula:

$$\mu_\theta(y_t, t) = \frac{1}{\sqrt{\alpha_t}}(y_t - \frac{1-\alpha_t}{\sqrt{1-\overline{\alpha}_t}}f_\theta(y_t, t)) \tag{5}$$

Therefore, using the noisy image y_t and the trained denoising network $f_\theta(y_t, t)$, we can estimate the distribution of y_{t-1}. In this way, starting from moment T, using $y_T \sim \mathcal{N}(0, I)$, we iteratively infer $y_{T-1}, y_{T-2}, y_{T-3}, \cdots$. In the final inference step, we directly use the predicted value from Equation 5. Thus, after T iterations, we obtain $\hat{y}_0$.

Image Restoration/Enhancement Algorithm Based on the Diffusion Model

For convenience, we denote x and y as the LQ (Low Quality) image and HQ (High Quality) image, respectively, within the context of image restoration/enhancement algorithms based on the diffusion model. Such algorithms typically construct a noisy image sequence of the HQ image in the forward process as follows: $\{y_t = \sqrt{\overline{\alpha}_t}y + \sqrt{1-\overline{\alpha}_t}\epsilon\}_{t=1}^{T}$.

During training, information about the LQ image x is injected as a condition into the U-Net network $f_\theta(y_t, x, t)$, and the network is utilized to predict the noise ϵ, thereby facilitating learning of the unknown conditional distribution $p(y|x)$.

$$p_\theta(y_{t-1}|y_t) = \mathcal{N}(y_{t-1}; \mu_\theta(y_t, x, t), \sigma_t^2 I) \tag{6}$$

$$\mu_\theta(y_t, x, t) = \frac{1}{\sqrt{\alpha_t}}(y_t - \frac{1-\alpha_t}{\sqrt{1-\overline{\alpha}_t}} f_\theta(y_t, x, t)) \tag{7}$$

In the inference process, $y_T \sim \mathcal{N}(0, I)$ is typically used as the starting point for generation. By applying Equations 6 and 7, the target image y is inferred after T iterative steps.

Methodology

In this section, we begin by outlining how DiffRAW utilizes the information of smartphone RAW images. Subsequently, we elucidate our solution for addressing the misalignment of color and position in training data pairs. Lastly, we detail our novel and efficient diffusion process, along with its corresponding training methods and reverse process.

RAW Condition

We utilize smartphone RAW images as a diffusion condition exclusively for preserving image structural information, such as contours and textures, without dependence on the RAW image for intricate details. This approach facilitates the full exploitation of mobile phone RAW image information without allowing the detail loss in the RAW image to interfere with the model's generated output. By constructing generated images using the structural information from RAW images and high-quality DSLR details learned through the diffusion model, this combined strategy ensures the model's generation results are comparable with DSLRs in image details while preserving the overall content of the smartphone's RAW images.

Color-Position Preserving Condition

For convenience, we denote w as the RAW image captured by the mobile phone, and y as the target sRGB image captured by a DSLR camera. Since w and y have an unstable color mapping relationship and are spatially misaligned, direct usage of the diffusion model to learn the conditional distribution $p(y|w)$ could lead to color biases in the model's output and result in image blurring and pixel shifting. To mitigate the interference caused by the color and spatial misalignment in training data pairs, we embed a color-position preserving condition c within DiffRAW, ensuring that the output images do not exhibit color biases and pixel shift issues.

During training, c^{train} is an sRGB image obtained by degrading y using a high-order degradation model (Wang et al. 2021). During testing, we use a color extraction network $\mathcal{G}(w; \Theta_\mathcal{G})$ to extract a naturally colored sRGB image from w as c^{test}, in order to inject color information into the model:

$$c^{train} = \mathcal{D}^2(y), c^{test} = \mathcal{G}(w; \Theta_\mathcal{G}) \tag{8}$$

Regarding the color extraction network $\mathcal{G}(w; \Theta_\mathcal{G})$, this paper adopts a pre-trained lightweight ISPNet (Zhang et al. 2021). Other pre-trained ISP networks, such as PyNet(Ignatov, Van Gool, and Timofte 2020), MW-ISPNet(Ignatov et al. 2020), etc., are also feasible. In fact, any network capable of extracting color information from w can serve as $\mathcal{G}(w; \Theta_\mathcal{G})$. c^{test} only functions to inject color information into the model. The generated results will maintain color consistency with c^{test}.

We make fine-tuned adjustments to the high-order degradation model $\mathcal{D}^2$ in terms of parameters and degradation methods to ensure strict color consistency between c^{train} and y. Since c^{train} and y are color-consistent and spatially aligned during training, DiffRAW can effectively learn the consistency in color and space between c and y in the conditional distribution $p(y|c, w)$, thus ensuring that the model's generated results maintain color consistency with c^{test}, without producing pixel shifts and blurs.

Domain Transform Diffusion Method

For ease of representation, we introduce an LQ image x, where during the training phase x is the DSLR-degraded image, and during the testing phase x is the output of the color extraction network $\mathcal{G}(w; \Theta_\mathcal{G})$:

$$x^{train} = \mathcal{D}^2(y), x^{test} = \mathcal{G}(w; \Theta_\mathcal{G}) \tag{9}$$

In this way, during the inference process, x_s can be utilized as an approximate estimation of y_s, serving as the starting point for generation. By employing equation 6 and equation 7, the target image y can be inferred through s iterative steps, thus reducing the number of iterations. Here, the definitions of x_s and y_s are given as follows, where $s \in \{1, 2, 3, \cdots, T\}$:

$$x_s = \sqrt{\overline{\alpha}_s}x + \sqrt{1-\overline{\alpha}_s}\epsilon \tag{10}$$

$$y_s = \sqrt{\overline{\alpha}_s}y + \sqrt{1-\overline{\alpha}_s}\epsilon \tag{11}$$

However, when using too small an iteration number s, the domain gap between x_s and y_s could lead to inconsistency between training and testing, consequently diminishing the enhancement in detail. To address this, we construct a new image diffusion sequence m_t with x and y, denoted as the Domain Transform Diffusion Method (DTDM), where $m_0 = y$ and $m_s = x_s$.

In the forward process, each diffusion step involves not only a slight addition of noise but also a minor degradation in the direction from y to x. In the reverse process, we add noise to x for s steps to obtain x_s as the starting point for generation, and then iterate s steps to generate the target image. Since each iteration in the reverse process achieves not only a single denoising but also a detail enhancement in the direction from x to y, we are able to significantly reduce the number of inference steps while enhancing the details more effectively.

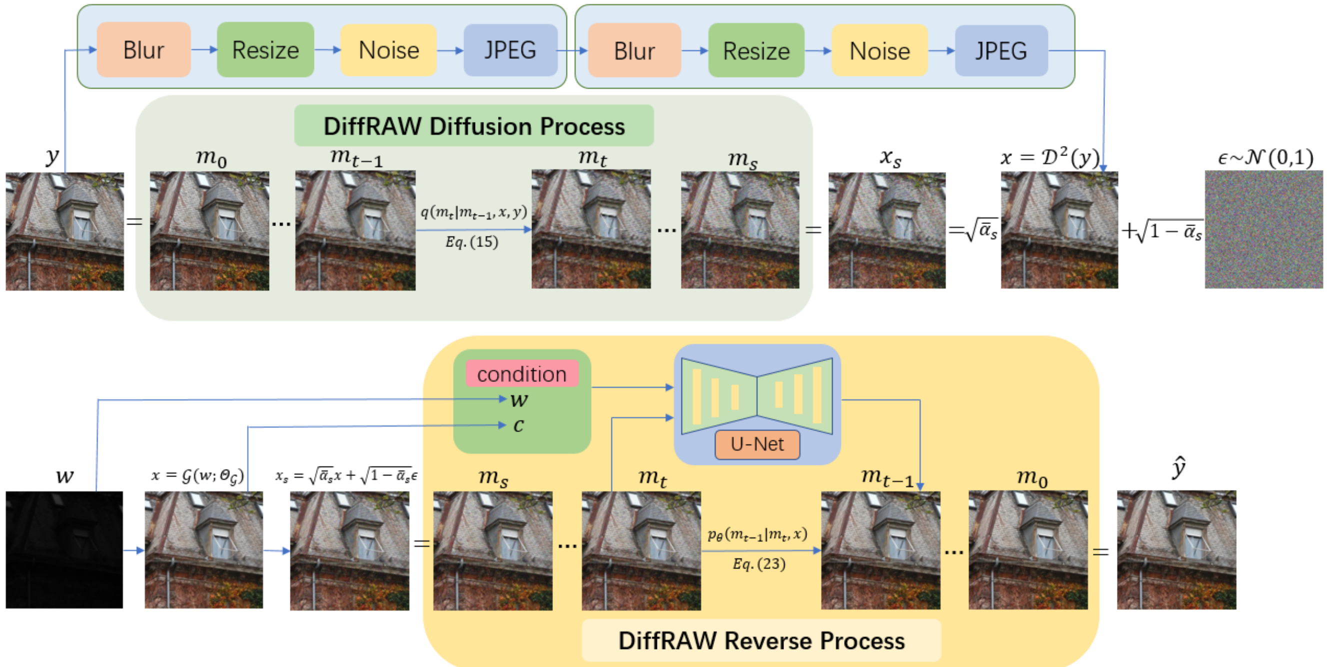


Figure 2: Overall DiffRAW Framework. The comprehensive structure of the proposed DiffRAW is composed of a forward process and an inverse process. In the forward process, we degrade y to x stochastically and construct a sequence m_t with a starting point of y and an endpoint of x_s. In the inverse process, we first extract x from w, add s steps of noise to x to attain the starting point of the inverse process x_s, and then use equation 23 for step-by-step iterative inference until $\hat{y}$ is generated.

Forward Process Suppose we aim to utilize x_s as the starting point for generation in the reverse process, iterating s steps to obtain the target image y. To ensure complete training-test consistency, we accordingly construct an image sequence $\{m_t\}_{t=0}^{s}$ that starts from y and ends at x_s.

In the forward process, a diffusion step from m_{t-1} to m_t is divided into two stages: a minor degradation in the direction from y to x, followed by a slight noise addition. For ease of expression, we let $m_{t-1} = m_{t-1}^{t-1}$, $m_t = m_t^t$, and the intermediate image after the first minor step of degradation from m_{t-1} is denoted as m_{t-1}^t. The process from m_{t-1} to m_t can be represented as follows:

$$m_{t-1}^t = m_{t-1}^{t-1} + \sqrt{\bar{\alpha}_{t-1}}(m_0^t - m_0^{t-1}) \tag{12}$$

$$m_t^t = \sqrt{\alpha_t} m_{t-1}^t + \sqrt{1-\alpha_t}\epsilon \tag{13}$$

Here, $t \in \{1, 2, 3, \cdots, s\}$, and the image sequence $\{m_0^t\}_{t=0}^{s}$ and constant γ_s are determined by the training hyperparameter $s \in \{1, 2, 3, \cdots, T\}$:

$$m_0^t = y + \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}[\gamma_s(x-y)], \gamma_s = \frac{\sqrt{\bar{\alpha}_s}}{\sqrt{1-\bar{\alpha}_s}} \tag{14}$$

We combine equation 12 and equation 13, performing both the degradation in the direction from y to x and the noise addition to m_{t-1} to obtain m_t. Therefore, for a given s, where $s \in \{1, 2, 3, \cdots, T\}$, the diffusion process of the image sequence $\{m_t\}_{t=0}^{s}$ can be represented as:

$$q(m_t|m_{t-1}, x, y) = \mathcal{N}(m_t; \mu_t^{diff}, (1-\alpha_t)I) \tag{15}$$

$$\mu_t^{diff} = \sqrt{\alpha_t} m_{t-1} + \sqrt{\bar{\alpha}_t}(m_0^t - m_0^{t-1}) \tag{16}$$

After recursively applying Equation 15, m_t's distribution can be directly computed from x and y:

$$q(m_t|x, y) = \mathcal{N}(m_t; \sqrt{\bar{\alpha}_t} m_0^t, (1-\bar{\alpha}_t)I) \tag{17}$$

The above can be understood as: applying noise t times to m_0^t results in $m_t^t = m_t$. Substituting equation 14 into equation 17 gives:

$$m_t = \sqrt{\bar{\alpha}_t}y + \sqrt{1-\bar{\alpha}_t}[\gamma_s(x-y)+\epsilon] \tag{18}$$

Thus, starting from $m_0 = y$, and after s times of diffusion, we obtain $m_s = \sqrt{\bar{\alpha}_s}x + \sqrt{1-\bar{\alpha}_s}\epsilon = x_s$.

Training Process We employed a U-Net network $f_\theta(m_t, w, c, t)$ for training, with the learning target being:

$$\frac{m_t - \sqrt{\bar{\alpha}_t}y}{\sqrt{1-\bar{\alpha}_t}} = \gamma_s(x-y) + \epsilon \tag{19}$$

Here, $\gamma_s(x-y)$ characterizes the high-frequency details between x and y, and ϵ represents the random noise of m_t. The loss function of the network can be expressed as:

$$L(\theta) = \mathbb{E}_{x,y,t,\epsilon}\|f_\theta(m_t, w, c, t) - [\gamma_s(x-y)+\epsilon]\|^2 \tag{20}$$

Upon completion of the training, for any moment t and the current image m_t, the estimates for the target image y can be obtained through equation 18 and equation 20, as:

$$\hat{y}(m_t, x, t) = \frac{m_t - \sqrt{1-\bar{\alpha}_t} f_\theta(m_t, w, c, t)}{\sqrt{\bar{\alpha}_t}} \tag{21}$$

Algorithm 1: DiffRAW Training

1: **repeat**
2: $(w, y) \sim q(w, y)$
3: $x = \mathcal{D}^2(y)$
4: $c = x$
5: $t \sim Uniform(\{1, 2, 3, \cdots, s\})$
6: $\epsilon \sim \mathcal{N}(0, I)$
7: $m_t = \sqrt{\overline{\alpha}_t} y + \sqrt{1 - \overline{\alpha}_t}[\gamma_s(x - y) + \epsilon]$
8: Take gradient descent step on $\nabla_\theta \| f_\theta(m_t, w, c, t) - [\gamma_s(x - y) + \epsilon] \|^2$
9: **until** converged

Algorithm 2: DiffRAW Inference

1: $x = \mathcal{G}(w; \Theta_{\mathcal{G}})$
2: $c = x$
3: $m_s \sim \mathcal{N}(m_s; \sqrt{\overline{\alpha}_s} x, (1 - \overline{\alpha}_s)I)$
4: **for** $t = s, \cdots, 1$ **do**
5: $z \sim \mathcal{N}(0, I)$ if $t > 1$,else $z = 0$
6: $m_{t-1} = [\frac{\sqrt{1-\overline{\alpha}_t}}{\sqrt{\overline{\alpha}_t}} \lambda_t - \frac{1-\alpha_t}{\sqrt{\alpha_t}\sqrt{1-\overline{\alpha}_t}}] f_\theta(m_t, w, c, t) + [\frac{1}{\sqrt{\alpha_t}} - \frac{1}{\sqrt{\overline{\alpha}_t}} \lambda_t] m_t + \lambda_t x + \sigma_t z$
7: **end for**
8: **return** m_0

Reverse Process In the reverse process, we utilize $m_s = x_s = \sqrt{\overline{\alpha}_s} x + \sqrt{1 - \overline{\alpha}_s}\epsilon$ as the starting point for generation, progressively iterating to infer $m_{s-1}, m_{s-2}, m_{s-3}, \cdots$. During each iteration, a denoising operation is performed, followed by a domain transform from the x to y direction. After s iterations, we arrive at $m_0 = y$.

Specifically, for any time t and the current image m_t, we can use the Bayes' theorem to simultaneously achieve the denoising of m_t and the domain transform from x to y direction, and directly infer m_{t-1} from m_t:

$$q(m_{t-1}|m_t, x, y) = q(m_t|m_{t-1}, x, y) \frac{q(m_{t-1}|x, y)}{q(m_t|x, y)} \quad (22)$$

By substituting equation 15, equation 17 and equation 21 into equation 22, we can obtain:

$$p_\theta(m_{t-1}|m_t, x) = \mathcal{N}(m_{t-1}; \hat{\mu}_\theta^{bayes}(m_t, x), \sigma_t^2 I) \quad (23)$$

$$\hat{\mu}_\theta^{bayes}(m_t, x) = [\frac{\sqrt{1 - \overline{\alpha}_t}}{\sqrt{\overline{\alpha}_t}} \lambda_t - \frac{1 - \alpha_t}{\sqrt{\alpha_t}\sqrt{1 - \overline{\alpha}_t}}] f_\theta(m_t, w, c, t) + [\frac{1}{\sqrt{\alpha_t}} - \frac{1}{\sqrt{\overline{\alpha}_t}} \lambda_t] m_t + \lambda_t x \quad (24)$$

$$\lambda_t = [\sqrt{1 - \overline{\alpha}_{t-1}}(1 - \sqrt{\alpha_t} \frac{\sqrt{1 - \overline{\alpha}_{t-1}}}{\sqrt{1 - \overline{\alpha}_t}})]\gamma_s \quad (25)$$

The training and sample process of DTDM are shown in algorithm1 and algorithm2. In previous diffusion-based image restoration/enhancement algorithms, if we use x_s as the starting point instead of y_s during inference for generation, the model merely denoises x_s at each step of the generation process. However, DTDM not only denoises x_s at each step of the generation process but also performs a domain transfer from x to y at each step, allowing DTDM to transform x_s into y with fewer iterations while also enhancing the quality of the generated images.

Experiments

Implementation Details

Datasets We conduct experiments on Zurich RAW to RGB (ZRR) dataset(Ignatov, Van Gool, and Timofte 2020). In the ZRR dataset, 20 thousand image pairs are collected and roughly aligned via SIFT keypoints (Lowe 2004) and the RANSAC algorithm (Vedaldi and Fulkerson 2010), and the cropped patches with cross-correlation < 0.9 are discarded, resulting in 48,043 RAW-sRGB pairs of size 448 ×448. We follow the official division to train our DiffRAW with 46.8k pairs, and report the quantitative results on the remaining 1.2k pairs.

Training Details We train our DiffRAW model for 1M training steps with a batch size of 32. Consistent with (Ho, Jain, and Abbeel 2020), we use the Adam optimizer with a linear warmup schedule over 10k training steps, followed by a fixed learning rate of 1e-4. The training hyperparameters T and s, which determine the noise scheduling and the distribution of the DTDM image sequence, are respectively set to 2000 and 100. We did not conduct more engineering attempts on the training hyperparameters T and s, and only set $s = 100$, $T = 2000$ to verify the effects of inference acceleration and improved image quality by DTDM. If more training hyperparameter trials are conducted on s and T, better experimental metric results might be achieved.

Evaluation Metrics For benchmarks with paired data, we employ various perceptual metrics including LPIPS (Zhang et al. 2018), FID (Heusel et al. 2017), MUSIQ (Ke et al. 2021) and CLIPIQA+ (Wang, Chan, and Loy 2023) to evaluate the perceptual quality of generated images. PSNR (Hore and Ziou 2010), SSIM (Zhou 2004), NIQE (Mittal, Soundararajan, and Bovik 2012) and IL-NIQE (Zhang, Zhang, and Bovik 2015) scores are also reported for reference. It should be noted specifically, in table 1, MUSIQ-K refers to `'musiq-koniq'`, MUSIQ-S refers to `'musiq-spaq'`, and CLIPIQA+RN50 refers to `'clipiqa+_rn50_512'`.

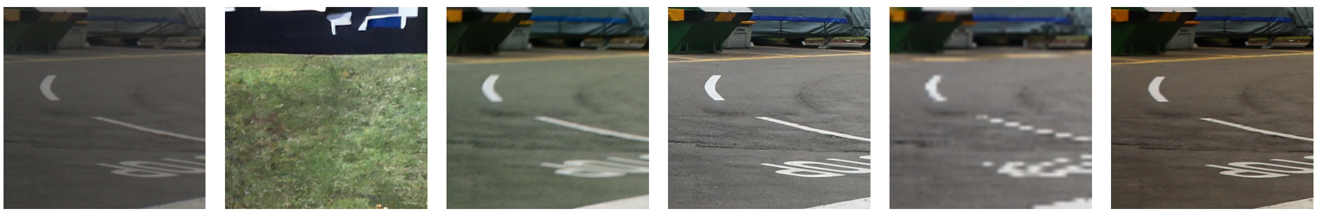
Testing Details Reducing the number of iterations during the inference process appropriately will lower the performance of the generated results on no-reference metrics, but enhance their performance on full-reference metrics. Therefore, we balance these two types of metrics and set the number of denoising steps and iteration steps during the inference process to 93, achieving the metric results shown in tables 1 and 2. If the number of denoising steps and iteration steps during the inference process is set to $s = 100$, the performance of the generated results on no-reference metrics will be better, which is also consistent with the human eye's observation of image details and image quality.

Method	MUSIQ-K↑	MUSIQ-S↑	CLIPIQA+↑	CLIPIQA+RN50↑	NIQE↓	ILNIQE↓
PyNet	43.56	46.4990	0.5353	0.3196	7.6856	50.55
MW-ISPNet	43.34	45.5973	0.5230	0.3097	7.9001	55.19
LiteISPNet	48.52	50.4763	0.5377	0.3063	7.4839	53.50
DiffRAW (ours)	**56.67**	**57.3660**	**0.5596**	**0.3739**	**7.0072**	**42.65**
DSLR(Reference)	56.62	57.4589	0.5622	0.3895	7.0181	44.13

Table 1: No Reference Metric Experimental Results on ZRR Dataset

Method	Original GT				Align GT with result			
	LPIPS↓	FID↓	PSNR↑	SSIM↑	LPIPS↓	FID↓	PSNR↑	SSIM↑
PyNet	0.193	18.69	21.19	0.7471	0.152	17.11	22.96	0.8510
MW-ISPNet	0.213	20.41	21.42	**0.7544**	0.164	18.48	23.31	0.8578
LiteISPNet	0.187	17.04	**21.55**	0.7487	0.133	15.30	**23.87**	**0.8737**
DiffRAW (ours)	**0.145**	**15.10**	21.31	0.7433	**0.118**	**14.61**	23.54	0.8682

Table 2: Full Reference Metric Experimental Results on ZRR Dataset



(a) RAW image (visualized) (b) Without condition (c) w condition (d) $w + c$ condition (e) image x (f) DSLR sRGB

Figure 3: Fig3(a) is the result of visualizing the RAW image using a simple ISP pipeline. Fig3(b) represents the generated result without condition. Fig3(c) represents the generated result using condition w. Fig3(d) represents the result using both w and c as conditions. Fig3(e) illustrates the image x utilized in these experiments. Fig3(f) represents the DSLR sRGB image.

(a) DDPM 1500 steps (b) DDPM 500 steps (c) DDPM 100 steps (d) DTDM 100 steps

Figure 4: A comparative analysis of the experimental results between DDPM and DTDM. Please zoom in for better observation.

Experimental Results on ZRR Dataset

To evaluate the effectiveness of the DiffRAW, we compare our model with three state-of-the-art methods, i.e., PyNet(Ignatov, Van Gool, and Timofte 2020), MW-ISPNet (Ignatov et al. 2020) and LiteISPNet(Zhang et al. 2021). As shown in table 1 and table 2, DiffRAW exceeds the competing methods on all perceptual quality metrics, while achieving comparable results in PSNR and SSIM.

Ablation Study

Diffusion Condition We introduced two diffusion conditions w and c to achieve exact control over the generated results. The w condition can stably generate image structure information, ensuring that the generated results maintain the original image's contours and textures. And the c condition can control over the color of the generated images while ensuring that there are no pixel shifts and blurring.

As illustrated in figures 3, we initiate the generation process by subjecting x to an eight-fold downsampling degradation followed by the addition of noise over 1500 steps, which serves as our starting point. It can be observed that with the incorporation of w, the contours and textures of the image are preserved. Furthermore, after the introduction of c, the image no longer exhibits any color bias or blurry shifts.

For a more detailed demonstration of the individual functionalities of the w and c conditions, such as the flexible manipulation of the generated results' color through the infusion of the c condition with various color representations, please refer to the supplementary material.

Diffusion Process and Inference Process We experimented with two types of diffusion processes for network training as described in equations 1 and 15. We use DDPM to represent the existing method, with its diffusion and the reverse process described by equations 1 and 4. And the DTDM, our improved method, is characterized by its diffusion and the reverse process through Equations 15 and 23. As shown in figure 4, an increase in the noise addition and iterative steps during the inference process leads to a corresponding enhancement in the detail of the generated results. Notably, our enhanced DTDM diffusion and inference process is capable of using only 100 steps of iteration to achieve detail enhancement surpassing that of 1500 iterative steps in DDPM.

Related Works

Deep Learning-based ISP Networks

To overcome the hardware limitations of mobile cameras, a significant number of attempts have been made in recent years towards the deep learning-based ISP methods. Ignatov et al. (Ignatov, Van Gool, and Timofte 2020) harnessed a RAW-sRGB dataset drawn from Huawei P20 smartphone and Canon 5D Mark IV DSLR, devising an end-to-end ISP network to supplant the conventional built-in ISP pathway of the smartphone. AWNet (Dai et al. 2020) incorporated the global context block (Cao et al. 2019) to mitigate the impact of image misalignment. Zhang et al. (Zhang et al. 2019) conceived a contextual bilateral (CoBi) loss, facilitating the discovery of the best matching patch for supervision and partly ameliorating data misalignment. However, this approach did not fully resolve the spatial displacement stemming from depth variations between objects. LiteISP-Net (Zhang et al. 2021) engineered a color-shift-resistant GCM module to contend with inconsistencies in color and pixel position shifts within data pairs, introducing a light-flow alignment module to synchronize the DSLR sRGB image with the mobile coordinate system. This alignment effectively attenuated the blurring and shifting complications in the output image, resulting from the misalignment in training data pairs. Further, Tripathi et al. (Shekhar Tripathi et al. 2022) tackled the pronounced color disparity between mobile RAW images and DSLR images through the utilization of a color prediction network grounded in the Perceiver architecture (Jaegle et al. 2021).

Diffusion Model

Over recent years, the diffusion model, distinguished by its superior ability for intricate detail generation, has outperformed Generative Adversarial Networks (GANs), positioning itself as the state-of-the-art methodology within the realm of image generation and editing. Deriving inspiration from non-equilibrium statistical physics, Sohl-Dickstein et al. (Sohl-Dickstein et al. 2015) were the pioneers in propounding the diffusion model as a tool to fit intricate distributions. Subsequently, Ho et al. (Ho, Jain, and Abbeel 2020) established a novel nexus between the diffusion model and denoising score matching. In a subsequent development, Song et al. (Song et al. 2020) advanced a unified framework to articulate the diffusion model through the lens of stochastic differential equations (SDEs). Several concurrent works have also leveraged analogous diffusion processes to DiffRAW. Although motivated by similar objectives, these efforts have embraced distinct mathematical formulations to realize this ambition. For instance, Delbracio and Milanfar (Delbracio and Milanfar 2023) employed Inversion by Direct Iteration (InDI) to model the process, while Luo et al. (Luo et al. 2023) and Liu et al. (Liu et al. 2023) sought to express it as an SDE.

Conclusion

In this work, we introduced DiffRAW, a novel method that adeptly addresses the challenges of converting smartphone RAW images to DSLR-quality sRGB images. DiffRAW's design incorporates the use of RAW images to maintain structural details and color-position preserving conditions to control color, coupled with an efficient diffusion process to enhance output quality and reduce inference steps. Evaluated on the ZRR dataset, DiffRAW consistently outperforms existing solutions in perceptual quality metrics, while achieving comparable results in PSNR and SSIM. Notably, DiffRAW marks the first achievement in reaching a level comparable to DSLR images on no-reference IQA metrics.

Acknowledgments

This work was supported by the Project from Science and Technology Innovation Committee of Shenzhen (KCXST20221021111201002) and the Key-Area Research and Development Program of Guangdong Province (2020B0909050003)

References

Cao, Y.; Xu, J.; Lin, S.; Wei, F.; and Hu, H. 2019. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*, 0–0.

Dai, L.; Liu, X.; Li, C.; and Chen, J. 2020. Awnet: Attentive wavelet network for image isp. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, 185–201. Springer.

Delbracio, M.; and Milanfar, P. 2023. Inversion by direct iteration: An alternative to denoising diffusion for image restoration. *arXiv preprint arXiv:2303.11435*.

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. *Advances in neural information processing systems*, 30.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Hore, A.; and Ziou, D. 2010. Image quality metrics: PSNR vs. SSIM. In *2010 20th international conference on pattern recognition*, 2366–2369. IEEE.

Ignatov, A.; Timofte, R.; Zhang, Z.; Liu, M.; Wang, H.; Zuo, W.; Zhang, J.; Zhang, R.; Peng, Z.; Ren, S.; et al. 2020. Aim 2020 challenge on learned image signal processing pipeline. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, 152–170. Springer.

Ignatov, A.; Van Gool, L.; and Timofte, R. 2020. Replacing mobile camera isp with a single deep learning model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 536–537.

Jaegle, A.; Gimeno, F.; Brock, A.; Vinyals, O.; Zisserman, A.; and Carreira, J. 2021. Perceiver: General perception with iterative attention. In *International conference on machine learning*, 4651–4664. PMLR.

Ke, J.; Wang, Q.; Wang, Y.; Milanfar, P.; and Yang, F. 2021. Musiq: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5148–5157.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Liang, Z.; Cai, J.; Cao, Z.; and Zhang, L. 2021. Cameranet: A two-stage framework for effective camera isp learning. *IEEE Transactions on Image Processing*, 30: 2248–2262.

Liu, G.-H.; Vahdat, A.; Huang, D.-A.; Theodorou, E. A.; Nie, W.; and Anandkumar, A. 2023. I^2SB: Image-to-Image Schrödinger Bridge. *arXiv*.

Lowe, D. G. 2004. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60: 91–110.

Luo, Z.; Gustafsson, F. K.; Zhao, Z.; Sjölund, J.; and Schön, T. B. 2023. Image restoration with mean-reverting stochastic differential equations. *arXiv preprint arXiv:2301.11699*.

Mittal, A.; Soundararajan, R.; and Bovik, A. C. 2012. Making a "completely blind" image quality analyzer. *IEEE Signal processing letters*, 20(3): 209–212.

Ramanath, R.; Snyder, W. E.; Yoo, Y.; and Drew, M. S. 2005. Color image processing pipeline. *IEEE Signal processing magazine*, 22(1): 34–43.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, 234–241. Springer.

Schwartz, E.; Giryes, R.; and Bronstein, A. M. 2018. Deepisp: Toward learning an end-to-end image processing pipeline. *IEEE Transactions on Image Processing*, 28(2): 912–923.

Shekhar Tripathi, A.; Danelljan, M.; Shukla, S.; Timofte, R.; and Van Gool, L. 2022. Transform your smartphone into a DSLR camera: Learning the ISP in the wild. In *European Conference on Computer Vision*, 625–641. Springer.

Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2256–2265. PMLR.

Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2020. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.

Vedaldi, A.; and Fulkerson, B. 2010. VLFeat: An open and portable library of computer vision algorithms. In *Proceedings of the 18th ACM international conference on Multimedia*, 1469–1472.

Wang, J.; Chan, K. C.; and Loy, C. C. 2023. Exploring clip for assessing the look and feel of images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 2555–2563.

Wang, X.; Xie, L.; Dong, C.; and Shan, Y. 2021. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1905–1914.

Zhang, L.; Zhang, L.; and Bovik, A. C. 2015. A feature-enriched completely blind image quality evaluator. *IEEE Transactions on Image Processing*, 24(8): 2579–2591.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 586–595.

Zhang, X.; Chen, Q.; Ng, R.; and Koltun, V. 2019. Zoom to learn, learn to zoom. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3762–3770.

Zhang, Z.; Wang, H.; Liu, M.; Wang, R.; Zhang, J.; and Zuo, W. 2021. Learning raw-to-srgb mappings with inaccurately aligned supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4348–4358.

Zhou, W. 2004. Image quality assessment: from error measurement to structural similarity. *IEEE transactions on image processing*, 13: 600–613.