YeTI: You Only Need Two Noisy Images for Real-World sRGB Noise Generation

Jaekyun Ko[1,2*], Byung Wan Lim[1*], Soomin Lee[1], Dongjin Kim[1], and Tae Hyun Kim[1†]

[1] Department of Computer Science, Hanyang University
{pook0612,min001017,dongjinkim,taehyunkim}@hanyang.ac.kr
[2] Mobile Experience (MX) Division, Samsung Electronics
jkko1124.ko@samsung.com

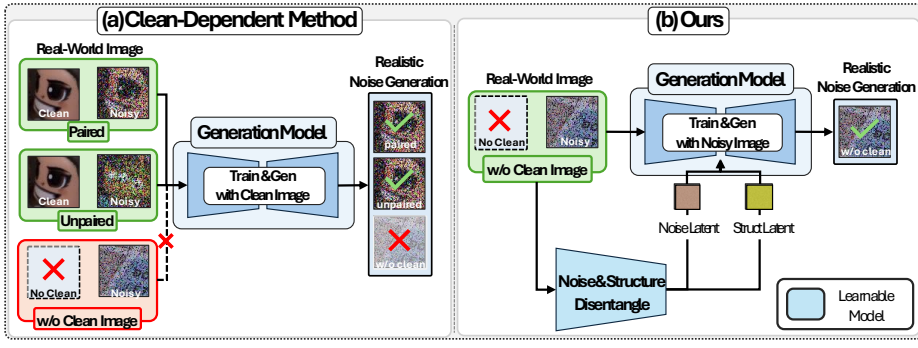

Fig. 1: Noise Generation Comparison. (a) Conventional clean-image-dependent approaches. (b) Our clean-image-free approach.

Abstract. Real-world sRGB image denoising remains challenging due to the nonlinear characteristics of sensor noise and the difficulty of acquiring aligned clean-noisy image pairs. Supervised denoisers often overfit to limited paired datasets, while self-supervised methods still depend on sufficiently diverse noisy observations. These limitations motivate scalable noise synthesis methods that can model real-world noise without clean ground truth or camera metadata. We propose YeTI, a real-world sRGB noise generation framework that learns from only two noisy observations of the same scene. YeTI uses a Reconstruction Autoencoder to disentangle scene structure and noise characteristics, and models the latent noise distribution with a one-step Conditional Diffusion Transformer trained using consistency objectives. Given a single noisy input at inference time, YeTI generates realistic, signal-dependent noise while preserving the underlying scene content. Extensive experiments demonstrate the effectiveness of YeTI across real-world benchmarks. We evaluate noise generation on SIDD and further assess generalization on SIDD+, MAI2021, and SID, covering smartphone and diverse consumer-camera sensors. Downstream denoising results on DND further show that denoisers trained with YeTI-synthesized images achieve strong real-world performance, highlighting the practical value of clean-image-free and metadata-free noise generation. Code is available at: https://github.com/ByungWanLim/YeTI-You-Only-Need-Two-Noisy-Images-for-Real-World-sRGB-Noise-Generation

* Equal contribution. † Corresponding author.

Table 1: Comparison between real-world sRGB noise modeling methods.

Generative Model	No Need for Clean?	Realistic Noise?	No Need for Metadata?
(a) NeCA, NAFlow	✗	✓	✗
(b) C2N	✗	✗	✓
(c) SeNM-VAE	✗	✓	✓
(d) YeTI (Ours)	✓	✓	✓

1 Introduction

Image denoising is a fundamental problem in computer vision and plays a critical role in various real-world applications such as photography [6,17], surveillance [58], and autonomous driving [25, 43, 48]. Despite the remarkable progress of deep learning-based denoisers [7, 8, 16, 36, 55, 61, 62, 66], achieving robust performance in various imaging conditions remains a challenging problem.

In particular, real-world sRGB denoising poses unique difficulties due to the complex characteristics of sensor noise and the nonlinear transformations applied during the image signal processing (ISP) pipeline [12, 13, 46]. The raw sensor noise becomes highly distorted after passing through demosaicing, tone mapping, gamma correction, and other camera-specific operations. Consequently, the noise distribution in the real-world sRGB domain deviates significantly from simple Poisson-Gaussian (PG) assumptions [10, 32], making it difficult for supervised denoising models to generalize well to real-world settings.

Moreover, the scarcity of real-world training datasets leads to overfitting problems in supervised denoising methods. Although datasets such as SIDD [3] and MIDD [9] attempt to collect real-world noisy–clean image pairs, the process is labor-intensive and expensive. This process requires capturing more than a thousand short- and long-exposure images for each scene under fixed camera settings and controlled environmental conditions, followed by additional post-processing to construct high-quality ground truth images.

To reduce the dependence on large-scale real-world data collection, recent studies [2, 11, 23, 27, 32, 34, 69] have modeled real-world noise distributions to synthesize additional noisy samples. Nevertheless, many of these methods require paired clean-noisy images or hardware-specific metadata, such as ISO and shutter speed, which are often unavailable in practical scenarios.

To address these limitations, we introduce YeTI, a clean-image-free framework for real-world sRGB noise generation. As illustrated in Fig. 1 and summarized in Tab. 1, our framework eliminates the need for paired clean-noisy images and camera metadata while synthesizing realistic sRGB noise. Existing sRGB noise modeling methods differ substantially in their data requirements and generation realism. Prior methods such as NeCA [11] and NAFlow [27] can synthesize realistic noise, but require either camera metadata or paired clean-noisy images for training or noise synthesis. Although C2N [23] learns noise generation in an unpaired setting, producing realistic noise remains challenging due to the instability of GAN-based unpaired training [4, 49]. SeNM-VAE [69] improves the realism of generated noise without relying on metadata, but still depends on paired datasets during training. In contrast, YeTI synthesizes realistic sRGB

noise using only two noisy burst observations during training and a single noisy image during inference, making it substantially more practical for real-world data acquisition.

At the core of our approach is a Reconstruction Autoencoder (RAE) that learns a compact latent space for disentangling scene structure and noise characteristics from input noisy images using only two consecutively captured burst observations. The RAE consists of two specialized encoders, namely a Structure Encoder and a Noise Encoder, together with a shared decoder. The Structure Encoder is trained with a contrastive objective [44] to extract noise-invariant content representations, where the two burst noisy images from the same scene form a positive pair and images from other scenes within the mini-batch serve as negatives. Conditioned on these structural representations, the Noise Encoder captures residual noise-specific information in a dedicated noise latent. The shared decoder then reconstructs the noisy input using both structural and noise latent features, preserving the underlying scene content while retaining signal-dependent and spatially correlated noise characteristics.

To model the distribution of the noise latent, we train a Conditional Diffusion Transformer (C-DiT) with a one-step consistency objective [41, 42, 51, 52]. The C-DiT is conditioned on the structure and noise latent features extracted from one noisy observation in the burst pair, and learns to predict the noise latent extracted from the other observation. This conditional training strategy provides sensor-specific noise guidance without requiring clean images or camera metadata, enabling the model to synthesize novel noise samples rather than merely reproducing a particular noise instance.

During inference, we sample a latent code from a simple prior and transform it with C-DiT into a noise latent conditioned on the structure and noise representations extracted from a single noisy image. The RAE decoder then maps the generated latent back to the image space, producing a synthesized noisy image that preserves the underlying scene structure while exhibiting realistic noise characteristics.

Through extensive experiments, we demonstrate that YeTI achieves state-of-the-art noise modeling performance across diverse real-world sensor domains. Furthermore, downstream denoising experiments show that denoisers trained with YeTI-synthesized noisy images achieve strong real-world performance, confirming the practical value of clean-image-free and metadata-free noise generation.

2 Related Works

2.1 Real-World Image Denoising

Image denoising has long been a fundamental problem in low-level vision and image processing. With the rapid progress of deep learning, recent approaches have largely shifted toward data-driven frameworks that learn complex noise-to-clean mappings from large-scale training data. Existing denoising methods can be broadly grouped into supervised and self-supervised paradigms.

Supervised methods train denoisers using paired clean–noisy images [7, 8, 15, 16, 28, 36, 55, 61, 62, 65–68]. Representative models, including NAFNet [7] and

MambaIR [16], adopt expressive architectures based on attention mechanisms or state-space modeling to enlarge the receptive field and capture complex degradation patterns. Despite their strong restoration performance, these methods often depend heavily on the training noise distribution, which can lead to degraded performance under unseen camera devices or imaging conditions. Such sensitivity limits their generalization ability in diverse real-world environments.

Self-supervised methods alleviate the need for clean ground truth images by exploiting internal noise statistics or spatial redundancy [33, 35, 38, 57, 59, 64]. Blind-spot-based approaches, such as AP-BSN [35] and LG-BPN [57], have shown promising performance on real-world noisy images by breaking spatial noise correlations during training. However, their effectiveness still largely depends on the diversity and representativeness of the available noisy data. When the noisy observations are limited or biased toward specific imaging conditions, self-supervised denoisers may overfit or fail to capture the full real-world noise distribution. As a result, both supervised and self-supervised paradigms remain constrained by the difficulty of acquiring diverse, high-quality real-world denoising datasets.

2.2 Real-World sRGB Noise Generation

To mitigate overfitting caused by limited real-world paired datasets, recent studies have explored sRGB noise modeling methods that synthesize realistic noisy images. These methods aim to capture the statistical, spatial, and signal-dependent characteristics of camera sensor noise, thereby enabling more robust denoiser training without requiring large-scale additional data collection.

NeCA [11] introduces a neighboring correlation-aware model that explicitly considers signal dependency and local noise correlations to synthesize realistic noise. NAFlow [27] further models sRGB noise distributions using a noise-aware normalizing flow, enabling realistic noise sampling without camera-specific metadata during inference. C2N [23] formulates real-world noise generation as an unpaired image-to-image translation problem, removing the need for paired supervision and metadata. SeNM-VAE [69] adopts a variational inference framework that supports paired and partially unpaired training schemes. Despite these advances, existing methods still face practical limitations. NeCA, NAFlow, and SeNM-VAE require either metadata or paired clean-noisy images during training, while C2N often struggles to faithfully model real-world sRGB noise due to the instability of GAN-based unpaired learning. These requirements limit their practicality and scalability in real-world data acquisition scenarios.

In contrast, our method relaxes these constraints by learning from only two noisy burst images of the same scene. This design removes the dependency on clean references and camera metadata during both training and inference, enabling scalable, metadata-free, and realistic sRGB noise synthesis.

3 Proposed Method

3.1 Preliminaries: One-step Diffusion

Diffusion Models. Diffusion models [20, 50, 53] generate data by reversing a forward corruption process in which *i.i.d.* Gaussian noise is added to a clean

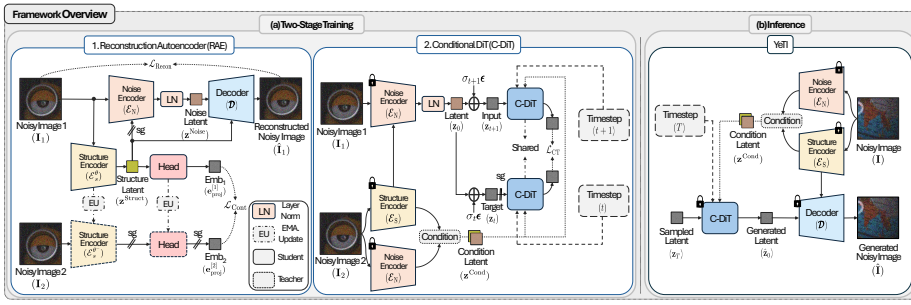

Fig. 2: Overview of the proposed method. (a) Two-stage training phase. (b) Inference phase.

sample $\mathbf{x}_0$. Using reparameterization, the noisy sample at timestep t is written as:

$$\mathbf{x}_t = \alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{\mathcal{I}}), \quad t \in \{0, 1, \ldots, T\}, \tag{1}$$

where $\boldsymbol{\epsilon}$ denotes Gaussian random noise, σ_t controls the noise level and we set $\alpha_t = 1$ following the formulation of EDM [26].

The reverse process is learned by solving the probability flow ODE [26], which transports $\mathbf{x}_T$ toward $\mathbf{x}_0$. Solving this ODE typically requires many update steps, which makes the sampling process computationally demanding.

Consistency Models (CM). To enable faster generation, consistency models [51,52] learn a direct single-step mapping that produces predictions consistent with the clean sample x_0 across different noise levels σ_t. Specifically, a consistency model predicts:

$$f^{\theta}(\mathbf{x}_t, \sigma_t) = \mathbf{x}_t + \int_{\sigma_t}^{\sigma_0} \frac{d\mathbf{x}_u}{du}\, du \approx \mathbf{x}_0, \tag{2}$$

and is trained with the consistency loss as:

$$\mathcal{L}_{\mathrm{CT}} = \mathbb{E}\left[\lambda(\sigma_t)\, d\left(f^{\theta}(\mathbf{x}_{t+1}, \sigma_{t+1}) - \mathrm{sg}(f^{\theta^-}(\mathbf{x}_t, \sigma_t))\right)\right], \tag{3}$$

where $d(\cdot)$ is a distance metric, $\lambda(\cdot)$ is a weighting function, and the stop-gradient operator (sg) stabilizes the training. Specifically, the teacher model f^{θ^-} provides a stable distillation target to ensure the student network f^{θ} learns consistent mappings along the probability flow ODE trajectory. Following prior work [31,51,52], we do not apply exponential moving average (EMA) updates from the student to the teacher, and instead share parameters between them such that $\theta^- = \theta$. The hyperparameters follow the setting in EDM [26] and iCT [51], and additional details are provided in the Supplementary Material Sec. S1.

3.2 Overall Flow

In this work, we propose YeTI to model real-world noise distributions from only a single noisy input during inference. To achieve this, the model must disentangle noise characteristics from the underlying scene structure in a given noisy image. This task is highly challenging, and existing approaches often rely on external information, such as camera metadata or paired clean images, to guide the separation.

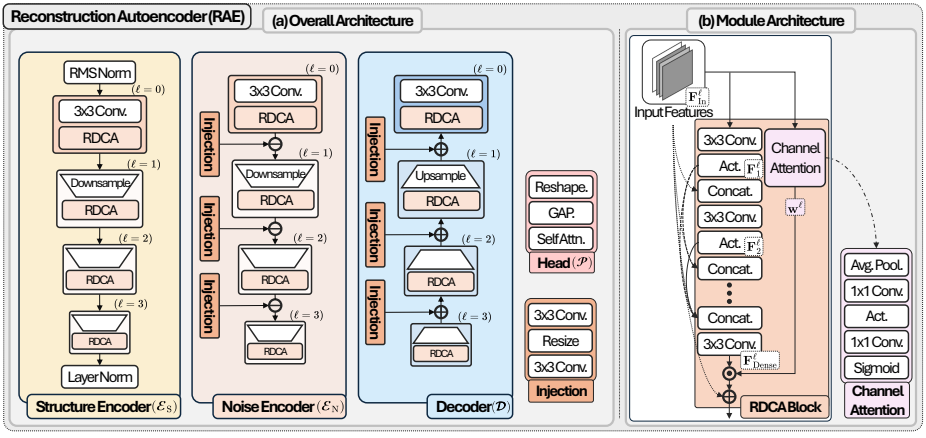

Fig. 3: Detailed configuration. (a) Reconstruction Autoencoder (RAE). (b) Residual Dense Channel Attention (RDCA) block.

In contrast, our approach learns to disentangle noise characteristics from the underlying scene structure by using burst noisy images captured from the same scene under identical camera settings during training. This design exploits the signal consistency shared across burst observations, allowing transient noise components to be isolated without clean references or camera metadata.

As illustrated in Fig. 2(a), YeTI consists of two main components: a Reconstruction Autoencoder (RAE) and a Conditional Diffusion Transformer (C-DiT). Following common latent diffusion practice [47], the framework is trained in two stages. First, the RAE learns a disentangled latent space that separates scene structure from stochastic noise using burst noisy observations. Then, with the RAE fixed, the C-DiT models the distribution of the noise latent through a one-step diffusion process, following the consistency training framework [42]. During inference, C-DiT transforms a randomly initialized latent into a realistic noise latent conditioned on a single noisy image, which is subsequently decoded by the RAE to synthesize a new noisy image while preserving the original scene content.

3.3 Reconstruction Autoencoder (RAE)

As shown in Fig. 3, the key challenge of clean-image-free noise modeling is to disentangle scene structure from stochastic noise using only noisy observations. To address this challenge, we propose a Reconstruction Autoencoder (RAE) consisting of a Structure Encoder $\mathcal{E}_s$, a Noise Encoder $\mathcal{E}_n$, and a Decoder $\mathcal{D}$.

Unlike previous approaches that rely on synthetic augmentations or restrictive prior assumptions [39], our framework exploits the physical redundancy of burst noisy images. Since two burst observations captured from the same static scene share identical scene content while containing independent noise realizations, the RAE can naturally separate invariant structural information from transient noise components. Consequently, the learned latent representations better reflect real-world sensor characteristics and provide an effective representation space for subsequent noise modeling.

The RAE first extracts scene-invariant structural representations, then suppresses redundant structural information to obtain a compact noise representation, and finally reconstructs the noisy image by combining both latent representations.

Structure Encoder ($\mathcal{E}_s$). The Structure Encoder $\mathcal{E}_s$ serves as the foundation of the RAE by extracting scene-invariant representations shared across burst observations. Let $\mathbf{I}_1$ and $\mathbf{I}_2$ denote two burst noisy images captured from the same static scene, where the subscripts indicate the burst indices. Although $\mathbf{I}_1$ and $\mathbf{I}_2$ contain independent noise realizations, they share identical underlying scene content. We exploit this property through contrastive learning, encouraging representations extracted from the same scene to be aligned while separating those from different scenes. As a result, the $\mathcal{E}_s$ preserves structural information while suppressing noise-specific variations.

As shown in Fig. 3(a), $\mathcal{E}_s$ first applies RMSNorm [28, 63] to stabilize the feature distribution across diverse noise conditions. Multi-scale features are then extracted using Residual Dense Channel Attention (RDCA) blocks, producing the structure latent representations $\mathbf{z}_1^{\text{Struct}}$ and $\mathbf{z}_2^{\text{Struct}}$. The extracted structure latents are subsequently shared with both the $\mathcal{E}_n$ and the $\mathcal{D}$. Specifically, they are used to suppress redundant scene information during noise encoding and restore structural information during image reconstruction, enabling explicit structure-noise disentanglement throughout the RAE.

Noise Encoder ($\mathcal{E}_n$). We introduce a Noise Encoder $\mathcal{E}_n$ to extract noise-specific information from $\mathbf{I}_1$ while suppressing scene structure. As in $\mathcal{E}_s$, $\mathcal{E}_n$ follows the multi-scale design and employs a stack of RDCA blocks to maintain feature compatibility across spatial resolutions. To encourage noise-focused representations, we remove structure-related information from $\mathcal{E}_n$ through subtractive injection at each scale, and later restore it in the decoder through additive injection. Unlike simple feature concatenation, this complementary injection strategy encourages the $\mathcal{E}_n$ to focus on stochastic noise characteristics while allowing the Decoder to recover the removed structural information, leading to improved structure-noise disentanglement.

Specifically, the structure latent $\mathbf{z}_1^{\text{Struct}}$ extracted by $\mathcal{E}_s$ is transformed into a scale-specific structure-guidance feature:

$$\mathbf{G}_\ell = \text{Conv}_{3\times3}\left(\text{Resize}_\ell\left(\text{Conv}_{3\times3}\left(\mathbf{z}_1^{\text{Struct}}\right)\right)\right), \tag{4}$$

where Resize_ℓ denotes spatial interpolation used to match the feature resolution at scale level $\ell \in \{1, 2, 3\}$. At each scale, $\mathbf{G}_\ell$ is subtracted from the $\mathcal{E}_n$ features to suppress structural information and isolate stochastic noise residuals. Finally, Layer Normalization (LN) [5] is applied to stabilize the noise feature distribution across diverse imaging conditions, improving optimization stability and producing the noise latent $\mathbf{z}_1^{\text{Noise}}$.

Decoder ($\mathcal{D}$). The Decoder $\mathcal{D}$ adopts a symmetric multi-scale architecture similar to the two encoders. Given the noise latent $\mathbf{z}_1^{\text{Noise}}$ as input, it progressively reconstructs the noisy image through a multi-scale decoding process. Starting from the coarsest level (*i.e.*, $\ell = 3$), it progressively upsamples the features and refines

them with RDCA blocks. Since structural information is intentionally suppressed in $\mathcal{E}_n$, $\mathcal{D}$ restores it through additive injection at each scale using the corresponding structure-guidance feature $\mathbf{G}_\ell$. This complementary design preserves scene fidelity while allowing $\mathcal{E}_n$ to remain dedicated to modeling stochastic noise.

Residual Dense Channel Attention (RDCA) Block. To encode scene structure and noise statistics into compact latent representations, we introduce a Residual Dense Channel Attention (RDCA) block. The RDCA block captures local spatial dependencies through dense convolutional connections and global channel relationships through channel attention.

As shown in Fig. 3 (b), the RDCA block first extracts dense spatial features using a stack of convolutional layers [21]. Let $\mathbf{F}_{\text{In}}^{\ell} \in \mathbb{R}^{\frac{H}{2^\ell} \times \frac{W}{2^\ell} \times C^\ell}$ denote the input feature at scale level $\ell \in \{0, 1, 2, 3\}$, where H and W are the spatial resolution of the input image and C^ℓ is the number of channels at scale ℓ. The dense feature embedding is defined as:

$$\mathbf{F}_{\text{Dense}}^{\ell} = \phi\left([\mathbf{F}_1^{\ell}, \mathbf{F}_2^{\ell}, \ldots, \mathbf{F}_K^{\ell}]\right), \tag{5}$$

where $\mathbf{F}_i^{\ell} = \delta(\text{Conv}_{3\times3}(\mathbf{F}_{i-1}^{\ell}))$ denotes the i-th intermediate feature, $\delta(\cdot)$ is a LeakyReLU activation, $[\cdot]$ indicates channel-wise concatenation, and $\phi(\cdot)$ is a 3×3 convolution used for feature refinement.

To incorporate global contextual information, we apply channel attention [7, 54, 61] to the dense feature:

$$\mathbf{F}_{\text{Attn}}^{\ell} = \mathbf{w}^{\ell} \odot \mathbf{F}_{\text{Dense}}^{\ell}, \tag{6}$$

where $\odot$ denotes element-wise multiplication. The channel-wise attention weight $\mathbf{w}^{\ell}$ is computed as:

$$\mathbf{w}^{\ell} = \sigma\left(\text{Conv}_{1\times1}\left(\delta\left(\text{Conv}_{1\times1}(\text{GAP}(\mathbf{F}_{\text{In}}^{\ell}))\right)\right)\right), \tag{7}$$

where GAP denotes global average pooling and σ is the sigmoid function. Finally, a residual connection is used to stabilize training and produce the output feature:

$$\mathbf{F}_{\text{Out}}^{\ell} = \mathbf{F}_{\text{In}}^{\ell} \oplus \mathbf{F}_{\text{Attn}}^{\ell}, \tag{8}$$

where $\oplus$ denotes element-wise addition. By combining dense spatial feature extraction with channel attention, the RDCA block effectively captures both local and global contextual information.

Training Pipeline. The RAE is optimized with a multi-task objective that reconstructs the input noisy image while encouraging the latent space to separate scene structure from noise characteristics. The main supervision is the reconstruction loss $\mathcal{L}_{\text{Recon}}$, defined as the Mean Absolute Error (MAE) between the reconstructed image $\hat{\mathbf{I}}_1$ and the input noisy image $\mathbf{I}_1$. This loss preserves pixel-level fidelity by recovering both the underlying signal and its corresponding noise realization.

To learn noise-invariant structural representations, we employ a contrastive loss $\mathcal{L}_{\text{Cont}}$ based on InfoNCE [44]. Specifically, we compute two embeddings,

$\mathbf{u}_1 = \mathcal{P}(\mathcal{E}_s(\mathbf{I}_1))$ and $\mathbf{u}_2 = \mathcal{P}(\mathcal{E}_s(\mathbf{I}_2))$, where $\mathcal{P}(\cdot)$ denotes a projection head applied to the output of the $\mathcal{E}_s$. Since $\mathbf{I}_1$ and $\mathbf{I}_2$ are captured from the same scene, $(\mathbf{u}_1, \mathbf{u}_2)$ forms a positive pair, while embeddings from other samples in the mini-batch are treated as negatives. This objective pulls representations of the same scene closer together despite different noise realizations, while pushing apart representations from different scenes. As a result, $\mathcal{E}_s$ is encouraged to capture structural information rather than noise-specific variations. To stabilize contrastive learning, we adopt a momentum teacher encoder $\mathcal{E}_s^{\theta^-}$ updated by an exponential moving average (EMA) of the online $\mathcal{E}_s^{\theta}$ and projection head, following common practice in self-supervised representation learning [14,18]. This momentum update provides slowly evolving target representations, facilitating the learning of scene-invariant structural representations from burst observations with different noise realizations. During training, $\mathcal{E}_s^{\theta}$ is optimized by backpropagation, while $\mathcal{E}_s^{\theta^-}$ provides target representations. As illustrated in Fig. 2(a), the stop-gradient (sg) operation prevents gradients from propagating through the teacher branch.

In addition, we apply a total variation loss $\mathcal{L}_{\text{TV}}$ [24] to the structure latent $\mathbf{z}_1^{\text{Struct}}$ to promote spatial coherence and suppress residual noise. This regularization helps obtain a cleaner structure representation while preserving the underlying scene content. We also apply an $\mathcal{L}_2$ regularization loss $\mathcal{L}_{\text{Norm}}$ to the noise latent $\mathbf{z}_1^{\text{Noise}}$ extracted by the $\mathcal{E}_n$, which constrains the latent scale and improves optimization stability.

The final RAE objective is defined as:

$$\mathcal{L}_{\text{RAE}} = \mathcal{L}_{\text{Recon}} + \lambda_{\text{Cont}}\mathcal{L}_{\text{Cont}} + \lambda_{\text{TV}}\mathcal{L}_{\text{TV}} + \lambda_{\text{Norm}}\mathcal{L}_{\text{Norm}}, \tag{9}$$

where λ_{Cont}, λ_{TV}, and λ_{Norm} control the contribution of each loss term. This objective jointly optimizes faithful reconstruction, robust structure-noise disentanglement, and stable latent representations, providing an effective representation space for the subsequent diffusion-based noise generation.

3.4 C-DiT: Learning Real-World Noise Distribution

Using the compact latent space learned by the RAE, we model the real-world noise distribution directly in the latent domain. Since the $\mathcal{E}_n$ extracts representations that capture the noise characteristics of the input image, learning a generative model in this space enables realistic noise synthesis.

To this end, we adopt a one-step diffusion framework based on consistency models, inspired by [30]. This framework learns a direct mapping from a sampled latent to its target noise latent, enabling efficient generation without iterative sampling.

Conditioning and Target Latent Features. Given a noisy observation $\mathbf{I}_1$, we first extract the target noise latent using the pretrained $\mathcal{E}_n$. Specifically, $\mathbf{I}_1$ is mapped to

$$\mathbf{z}_0 = \text{LN}(\mathcal{E}_n(\mathbf{I}_1)), \tag{10}$$

which serves as the target latent for the diffusion model. Layer Normalization (LN) is applied to align the latent distribution and improve the stability of diffusion training.

To guide noise synthesis, we construct a conditioning feature $\mathbf{z}^{\text{Cond}}$ from the other noisy observation $\mathbf{I}_2$ in the burst pair. This condition combines the noise latent $\mathbf{z}_2^{\text{Noise}} = \mathcal{E}_n(\mathbf{I}_2)$ and the structure latent $\mathbf{z}_2^{\text{Struct}} = \mathcal{E}_s(\mathbf{I}_2)$ through channel-wise concatenation:

$$\mathbf{z}^{\text{Cond}} = \left[\mathbf{z}_2^{\text{Noise}}, \mathbf{z}_2^{\text{Struct}}\right]. \tag{11}$$

Here, $\mathbf{z}_2^{\text{Noise}}$ is used without LN to preserve raw noise-related statistics, such as variations induced by sensor type and ISO. Meanwhile, $\mathbf{z}_2^{\text{Struct}}$ provides a structural anchor for the underlying scene content. By conditioning on both noise and structure information, C-DiT learns the signal-dependent characteristics of real-world noise and synthesizes realistic noise latent features.

Training Pipeline. We apply the diffusion process described in Sec. 3.1 to the target latent $\mathbf{z}_0$, producing noisy latent variables $\mathbf{z}_t$ for $t \in \{0, 1, \ldots, T\}$. At each timestep t, the forward process is defined as:

$$\mathbf{z}_t = \mathbf{z}_0 \oplus \sigma_t \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{\mathcal{I}}), \tag{12}$$

where σ_t denotes the noise level. Following the consistency objective in Eq. (3), C-DiT is trained to map different points along the same probability flow ODE trajectory to a consistent target state:

$$\text{C-DiT}_\theta(\mathbf{z}_{t+1}, \sigma_{t+1} \mid \mathbf{z}^{\text{Cond}}) \approx \text{C-DiT}_\theta(\mathbf{z}_t, \sigma_t \mid \mathbf{z}^{\text{Cond}}) \approx \mathbf{z}_0. \tag{13}$$

This training objective encourages C-DiT to serve as a stable single-step generation operator across different noise levels.

3.5 sRGB Noise Generation with Single Noisy Image

With the fully trained RAE and C-DiT, YeTI synthesizes burst noisy images from a single noisy input. As illustrated in Fig. 2 (b), we first sample a random latent $\mathbf{z}_T$ from a predefined normal distribution. Conditioned on latent features extracted from the input noisy image $\mathbf{I}$, C-DiT transforms $\mathbf{z}_T$ into a latent code $\hat{\mathbf{z}}_0$ that captures realistic, signal-dependent noise characteristics. The decoder $\mathcal{D}$ then maps $\hat{\mathbf{z}}_0$ back to the image space to generate a synthesized noisy image $\hat{\mathbf{I}}$. This process enables YeTI to generate diverse noisy images from a single input while preserving the underlying scene content and matching the input noise distribution.

4 Experiments

4.1 Experimental Setup

Implementation Details. The RAE is trained with the Adam optimizer [29] by minimizing the joint objective $\mathcal{L}_{\text{RAE}}$ defined in Eq. (9). The weights of the individual loss terms are provided in the Supplementary Material Sec S2. We initialize the learning rate to 1×10^{-4} and use a cosine annealing scheduler [40] to gradually decrease it to 1×10^{-6} over 200k iterations. During training, we use randomly cropped 256×256 patches with a mini-batch size of 64.

The C-DiT model is trained from scratch without distillation. We optimize it with the RAdam optimizer [37] using a fixed learning rate of 2×10^{-4} for 100k

iterations. Following iCT [51], we use the pseudo-Huber loss to optimize the consistency objective in Eq. (3). The input images are randomly cropped to 256×256, producing latent codes of size 32×32, and the mini-batch size is set to 64.

For downstream denoising with the synthesized data, we adopt blind-spot networks (BSNs), which are trained using only noisy images and therefore align well with our noise generation setting. Specifically, we use AP-BSN [35] and MM-BSN [64] with their official training configurations.

All YeTI models are trained on four NVIDIA RTX A6000 GPUs and inference is evaluated on a single A6000 GPU.

Metrics. We evaluate the quality of the generated noise using the Kullback-Leibler divergence (KLD) and the Average KLD (AKLD) [60]. For denoising performance, we report PSNR and SSIM [56] values.

Dataset. We train YeTI on the SIDD training set [3], which covers 34 camera configurations. Following prior works [27, 69], we use the SIDD Medium split, which consists of 320 noisy-clean image pairs captured by five smartphones: Google Pixel (GP), iPhone 7 (IP), Samsung Galaxy S6 Edge (S6), Motorola Nexus 6 (N6), and LG G4 (G4). Although clean images are provided, YeTI does not use them for training; instead, it only exploits the two noisy burst observations available for each scene, which naturally satisfy the input requirement of our framework. We use the SIDD validation set to evaluate both generated noise quality and downstream denoising performance. To assess cross-dataset robustness, we further evaluate noise generation on SIDD+ [1], MAI2021 [22], and the See-in-the-Dark (SID) dataset [6], which includes images captured by Fujifilm X-T2 and Sony α7S II cameras. Finally, we use the DND benchmark [45] to evaluate downstream denoising performance on unseen real-world domains.

Table 2: Quantitative results of synthetic noise on the SIDD validation subset across different camera devices. The results are computed with KLD↓ and AKLD↓. The best and second-best results are shown in **bold** and underline, respectively.

Camera	G4		GP		IP		N6		S6		**Average**	
Methods	KLD↓	AKLD↓	KLD↓	AKLD↓	KLD↓	AKLD↓	KLD↓	AKLD↓	KLD↓	AKLD↓	KLD↓	AKLD↓
NAFlow (100%)	0.0254	0.1367	0.0352	0.1180	0.0339	0.1522	0.0309	0.1108	**0.0272**	0.1355	0.0305	0.1306
SeNM-VAE (0.01%)	0.0942	0.1509	0.0490	0.1204	0.0687	**0.1169**	0.0731	0.1244	0.0512	0.1350	0.0672	0.1295
C2N (0%)	0.1660	0.2007	0.1315	0.1968	0.0581	0.2929	0.3524	0.2919	0.4517	0.4190	0.2129	0.2802
Ours (0%)	**0.0226**	**0.1171**	**0.0218**	**0.1017**	**0.0328**	0.1176	**0.0224**	**0.1024**	0.0284	**0.1155**	**0.0256**	**0.1108**

4.2 sRGB Noise Generation Performance

Device-Specific Noise Quality Assessment. In Tab. 2, we evaluate device-specific noise generation quality using KLD and AKLD. We compare YeTI with three representative methods under different supervision settings: NAFlow [27] and SeNM-VAE [69], which use paired noisy-clean supervision, and C2N [23], which follows an unpaired setting. For a fair comparison, we synthesize noise using SIDD validation pairs whose metadata, such as device type and ISO, matches that of the training set.

Compared with existing methods, YeTI achieves highly competitive performance across nearly all device types and obtains the lowest average KLD and

AKLD scores. The visual results in Fig. 4 further show that YeTI synthesizes realistic noise patterns that closely match real-world noise in magnitude, spatial correlation, and signal dependency. These quantitative and qualitative results demonstrate that YeTI effectively models device-specific noise distributions with high fidelity.

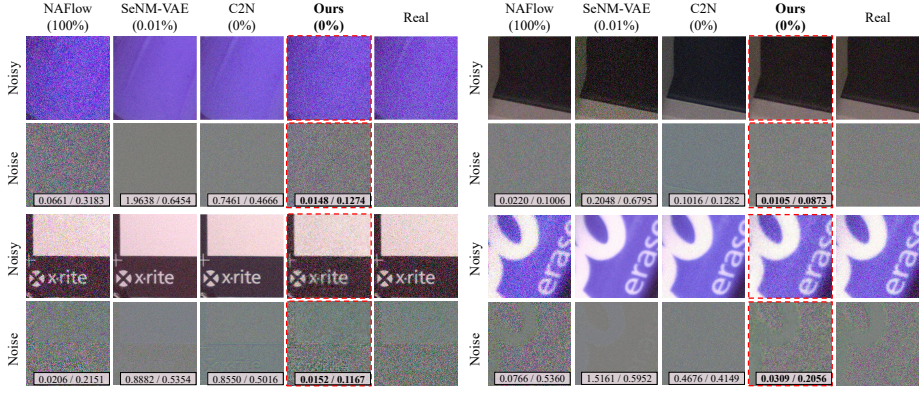

Fig. 4: Visualization of synthetic noisy images on the SIDD validation set. From left to right: NAFlow (100%), SeNM-VAE (0.01%), C2N (0%), Ours (0%, YeTI), and real noisy images. The percentage indicates the amount of paired data used to train each noise generation model. Metrics below each image denote KLD↓/AKLD↓.

Robustness on Various Real-World Datasets. To further validate the robustness and generalization ability of YeTI, we evaluate the quality of generated noise on multiple real-world datasets, including SIDD+, MAI2021, and SID. As reported in Tab. 3, YeTI consistently achieves lower KLD and AKLD scores than NAFlow across all evaluated datasets. This is particularly notable since NAFlow requires paired noisy-clean images and camera metadata during training, whereas YeTI does not rely on either. These results demonstrate the robustness of YeTI across diverse camera environments and highlight its potential for augmenting real-world datasets with realistic synthetic noisy images without requiring paired ground truth or metadata.

Table 3: Quantitative results of synthetic noise on the SIDD+, MAI2021, and SID datasets (Fujifilm and Sony). All methods are trained with the SIDD training set. The results are computed with KLD↓ and AKLD↓. The best results are shown in **bold**.

Methods	SIDD+		MAI2021		Fujifilm X-T2		Sony α7S II		Average	
	KLD↓	AKLD↓	KLD↓	AKLD↓	KLD↓	AKLD↓	KLD↓	AKLD↓	KLD↓	AKLD↓
NAFlow (100%)	0.0493	0.2914	0.7304	2.6465	0.3850	1.4640	0.2970	1.3860	0.3654	1.4470
Ours (0%)	**0.0251**	**0.1451**	**0.0795**	**0.2860**	**0.1510**	**0.2410**	**0.1070**	**0.3690**	**0.0907**	**0.2603**

4.3 Real-World Denoising Performance

Denoising Results on SIDD. To evaluate the effectiveness of generated noise in a downstream denoising task, we train AP-BSN and MM-BSN on synthetic noisy datasets generated from the SIDD training set. In Tab. 4, we compare self-supervised denoisers trained with YeTI-generated images against those trained

Table 4: Quantitative denoising results on the SIDD validation dataset using different synthetic noisy data generation strategies. We report PSNR and SSIM values for two denoising backbones, AP-BSN and MM-BSN. The best and second-best results are denoted as **bold** and underline, respectively.

Methods	AP-BSN		MM-BSN	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Real Noisy	36.58	0.8865	37.01	0.8915
NAFlow (100%)	36.25	0.8912	36.66	0.8941
SeNM-VAE (0.01%)	36.58	0.8911	36.83	0.8868
C2N (0%)	34.06	0.8519	34.15	0.8486
Ours (0%)	36.86	0.8946	37.01	**0.8950**
SeNM-VAE-Mixed (50%)	36.71	0.8927	36.91	0.8894
Ours-Mixed (50%)	**37.09**	**0.8953**	**37.09**	**0.8950**

Table 5: Quantitative denoising results on the DND benchmark dataset using different synthetic noisy data generation strategies. We report PSNR and SSIM values for two denoising backbones, AP-BSN and MM-BSN. The best and second-best results are denoted as **bold** and underline, respectively.

Methods	AP-BSN		MM-BSN	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Real Noisy	38.20	0.939	38.61	0.941
Ours (0%)	38.37	0.940	38.62	**0.942**
Ours-Mixed (50%)	**38.55**	**0.941**	**38.70**	**0.942**

with images synthesized by competing methods, including NAFlow, SeNM-VAE, and C2N, as well as a *Real Noisy* baseline. The percentages in parentheses indicate the proportion of paired clean-noisy data used to train each noise generation model.

YeTI consistently outperforms competing noise generators and achieves performance comparable to, or slightly better than, the *Real Noisy* baseline. This indicates that YeTI synthesizes realistic and diverse noise patterns that help reduce overfitting by exposing the denoiser to a broader range of noise realizations than the original training set. Moreover, our *Mixed* strategy, which combines real and synthetic noisy images, further improves denoising performance and surpasses both the *Real Noisy* baseline and other mixed-data approaches. In contrast, BSN models trained with C2N-generated data show noticeably lower performance, suggesting that its synthesized noise distribution is less aligned with the target real-world noise characteristics required for effective denoising. Qualitative results for both denoisers are shown in Fig. 5.

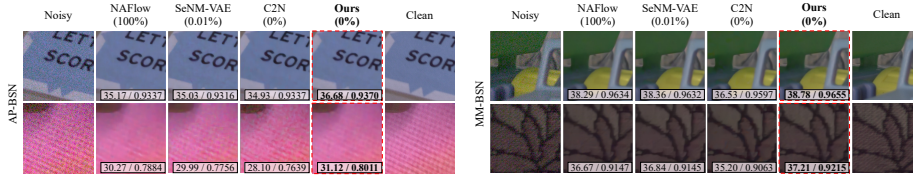

Fig. 5: Visual comparison. Denoising results on SIDD validation set from AP-BSN and MM-BSN trained on each method. From left to right: noisy image, NAFlow (100%), SeNM-VAE (0.01%), C2N (0%), Ours (0%, YeTI), and clean GT. The percentage indicates the amount of paired data used to train each noise generation model. Numbers below each image denote PSNR↑/SSIM↑.

Denoising Results on External Dataset. To evaluate the generalization of synthetic noise beyond the original training distribution, we further conduct downstream denoising experiments on the DND benchmark. Since DND does not provide clean-noisy pairs for training, YeTI synthesizes noisy images using only the available noisy observations. As shown in Tab. 5, AP-BSN and MM-BSN

trained on YeTI-generated data achieve performance comparable to, or slightly better than, the *Real Noisy* baseline. In addition, the *Mixed* strategy, which combines real and synthetic noisy images, provides further performance gains. These results indicate that YeTI can generate effective training data for unseen real-world domains and improve denoising robustness without requiring clean ground truth.

Table 6: Quantitative results of metadata classification on SIDD validation. We evaluate the impact of Noise Encoder components on classification accuracy ↑.

Methods	Camera Sensor (%)	Camera Sensor + ISO	
		Top-1 (%)	Top-3 (%)
Noisy Image	49.98	41.66	77.14
w/o $\mathcal{L}_{\text{Norm}}$	56.65	47.20	83.73
w/o Head	<u>59.70</u>	<u>51.20</u>	<u>85.66</u>
Ours	**76.20**	**65.22**	**94.87**

Table 7: Evaluation of $\mathbf{z}^{\text{Struct}}$ purity on SIDD validation. We decode images using only $\mathbf{z}^{\text{Struct}}$ (*i.e.*, $\mathbf{z}^{\text{Noise}} = \mathbf{0}$) and compare against Clean GT.

Methods	PSNR↑	SSIM↑
w/o $\mathcal{L}_{\text{Cont}}$	15.49	0.4855
w/o Head	17.21	0.6769
w/o $\mathcal{L}_{\text{TV}}$	<u>21.49</u>	<u>0.7090</u>
Ours	**23.06**	**0.7597**

4.4 Ablation Study

Noise Latent Quality Evaluation. To examine whether $\mathbf{z}^{\text{Noise}}$ captures intrinsic noise characteristics, we conduct a metadata classification task. Specifically, we train a ResNet-based classifier [19] to classify 16 noise profiles defined by five camera sensors and their corresponding ISO levels. As shown in Tab. 6, our full model achieves the highest accuracy, outperforming the *Noisy Image* baseline. Removing either the projection head or $\mathcal{L}_{\text{Norm}}$ decreases the classification accuracy, demonstrating their importance in learning a discriminative noise representation. In particular, $\mathcal{L}_{\text{Norm}}$ constrains the noise latent and discourages the $\mathcal{E}_n$ from encoding redundant scene structure, guiding it to focus on stochastic noise statistics. These results indicate that $\mathcal{E}_n$ extracts noise-specific attributes that are useful for accurate real-world sRGB noise modeling.

Structure Latent Quality Evaluation. To evaluate the quality of the structure latent $\mathbf{z}^{\text{Struct}}$, we perform a reconstruction test by feeding only the structure latent into the RAE decoder while setting $\mathbf{z}^{\text{Noise}}$ to zero. The reconstructed image is then compared with the clean ground truth. As shown in Tab. 7, removing either $\mathcal{L}_{\text{Cont}}$ or the projection head $\mathcal{P}$ leads to lower PSNR and SSIM, indicating weaker structure-noise disentanglement. Without these components, the model tends to rely more on $\mathbf{z}^{\text{Noise}}$ for reconstruction, leaving $\mathbf{z}^{\text{Struct}}$ less informative. In contrast, our full model preserves scene-specific structural information more effectively while encouraging $\mathcal{E}_n$ to focus on sensor-specific noise statistics. In addition, $\mathcal{L}_{\text{TV}}$ further improves the structure representation by suppressing residual high-frequency noise.

Analysis on Conditioning Strategy. We compare our dual-conditioning strategy with a Single Encoder baseline, which uses the Noise Encoder architecture $\mathcal{E}_n$ without structural suppression through subtractive injection. As shown in Tab. 8, our method consistently achieves lower KLD and AKLD scores across all evaluated datasets. This improvement comes from the $\mathcal{E}_s$, which separates

invariant scene information from noise-related features, allowing C-DiT to model the target noise distribution with less structural interference. By using $\mathbf{z}^{\text{Struct}}$ as a stable structural anchor and $\mathbf{z}^{\text{Noise}}$ as statistical noise guidance, YeTI produces more realistic noisy images. The resulting improvement in noise fidelity also leads to higher downstream PSNR for AP-BSN, demonstrating that disentangled conditioning is beneficial for real-world noise modeling and restoration.

Table 8: Ablation study on the conditioning mechanism of C-DiT on the SIDD validation, SIDD+, and MAI2021 datasets. We evaluate the efficacy of disentangled conditioning by comparing the baseline using a single latent representation against our strategy utilizing $\mathbf{z}^{\text{Struct}}$ and $\mathbf{z}^{\text{Noise}}$.

Method	SIDD		SIDD+		MAI2021	
	KLD/AKLD(↓)	PSNR/SSIM(↑)	KLD/AKLD(↓)	PSNR/SSIM(↑)	KLD/AKLD(↓)	PSNR/SSIM(↑)
Single Enc	0.0275/0.1118	36.82/0.8942	0.0322/0.1624	35.74/0.9037	0.1584/0.4309	33.73/0.8589
Ours	**0.0256**/**0.1108**	**36.86**/**0.8946**	**0.0251**/**0.1451**	**35.81**/**0.9104**	**0.0793**/**0.2860**	**34.40**/**0.8798**

5 Conclusion

We presented YeTI, a clean-image-free and metadata-free framework for real-world sRGB noise generation. Unlike prior approaches that rely on paired clean-noisy images or camera metadata, YeTI learns from only two noisy observations of the same scene during training and requires only a single noisy image at inference. By disentangling scene structure and noise characteristics with a Reconstruction Autoencoder (RAE) and modeling the latent noise distribution using a one-step Conditional DiT (C-DiT) trained with consistency objectives, YeTI synthesizes realistic, signal-dependent noise while preserving the underlying scene content. Extensive experiments demonstrate that YeTI effectively captures device-specific noise distributions and generalizes across diverse real-world sensor domains. Moreover, self-supervised denoisers trained with YeTI-synthesized data achieve competitive performance compared to those trained on real noisy images, confirming the practical value of scalable noise generation without clean references or metadata. These results suggest that two-noisy-observation-based noise modeling is a promising direction for building robust real-world image restoration systems.

Acknowledgments

This research was supported by Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2026(Project Name: Development of AI-Based Animation Production Technology to Ensure Character and Scene Consistency and Continuity for Enhanced Efficiency and Quality, Project Number: RS-2026-25525207, Contribution Rate: 30%). This work was also supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2022-0-00156, Fundamental research on continual meta-learning for quality enhancement of casual videos and their 3D metaverse transformation), IITP grant funded by the Korea government (MSIT) (No.RS-2020-II201373, Artificial Intelligence Graduate School Program (Hanyang University)).

References

1. Abdelhamed, A., Afifi, M., Timofte, R., Brown, M.S.: Ntire 2020 challenge on real image denoising: Dataset, methods and results. In: CVPR (2020)
2. Abdelhamed, A., Brubaker, M.A., Brown, M.S.: Noise flow: Noise modeling with conditional normalizing flows. In: ICCV (2019)
3. Abdelhamed, A., Lin, S., Brown, M.S.: A high-quality denoising dataset for smartphone cameras. In: CVPR (2018)
4. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: ICML (2017)
5. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv (2016)
6. Chen, C., Chen, Q., Xu, J., Koltun, V.: Learning to see in the dark. In: CVPR (2018)
7. Chen, L., Chu, X., Zhang, X., Sun, J.: Simple baselines for image restoration. In: ECCV (2022)
8. Chen, L., Lu, X., Zhang, J., Chu, X., Chen, C.: Hinet: Half instance normalization network for image restoration. In: CVPR (2021)
9. Flepp, R., Ignatov, A., Timofte, R., Van Gool, L.: Real-world mobile image denoising dataset with efficient baselines. In: CVPR (2024)
10. Foi, A., Trimeche, M., Katkovnik, V., Egiazarian, K.: Practical poissonian-gaussian noise modeling and fitting for single-image raw-data. TIP (2008)
11. Fu, Z., Guo, L., Wen, B.: srgb real noise synthesizing with neighboring correlation-aware noise model. In: CVPR (2023)
12. Gow, R.D., Renshaw, D., Findlater, K., Grant, L., McLeod, S.J., Hart, J., Nicol, R.L.: A comprehensive tool for modeling cmos image-sensor-noise performance. IEEE T-ED (2007)
13. Green, P.: Fundamentals and Applications of Colour Engineering. John Wiley & Sons (2023)
14. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al.: Bootstrap your own latent-a new approach to self-supervised learning. NeurIPS (2020)
15. Guo, H., Guo, Y., Zha, Y., Zhang, Y., Li, W., Dai, T., Xia, S.T., Li, Y.: Mambairv2: Attentive state space restoration. In: CVPR (2025)
16. Guo, H., Li, J., Dai, T., Ouyang, Z., Ren, X., Xia, S.T.: Mambair: A simple baseline for image restoration with state-space model. In: ECCV (2024)
17. Hasinoff, S.W., Sharlet, D., Geiss, R., Adams, A., Barron, J.T., Kainz, F., Chen, J., Levoy, M.: Burst photography for high dynamic range and low-light imaging on mobile cameras. TOG (2016)
18. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: CVPR. pp. 9729–9738 (2020)
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
20. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. NeurIPS (2020)
21. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: CVPR (2017)
22. Ignatov, A., Byeoung-su, K., Timofte, R., Pouget, A.: Fast camera image denoising on mobile gpus with deep learning, mobile ai 2021 challenge: Report. In: CVPRW (2021)

23. Jang, G., Lee, W., Son, S., Lee, K.M.: C2n: Practical generative noise modeling for real-world denoising. In: ICCV (2021)
24. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: ECCV (2016)
25. Kamann, C., Rother, C.: Benchmarking the robustness of semantic segmentation models. In: CVPR (2020)
26. Karras, T., Aittala, M., Aila, T., Laine, S.: Elucidating the design space of diffusion-based generative models. NeurIPS (2022)
27. Kim, D., Jung, D., Baik, S., Kim, T.H.: srgb real noise modeling via noise-aware sampling with normalizing flows. In: ICLR (2024)
28. Kim, D., Ko, J., Ali, M.K., Kim, T.H.: Idf: Iterative dynamic filtering networks for generalizable image denoising. In: ICCV (2025)
29. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv (2014)
30. Ko, J., Kim, D., Lee, S., Wang, G., Kim, T.H.: Diffusion-based srgb real noise generation via prompt-driven noise representation learning. In: CVPR (2026)
31. Kong, F., Duan, J., Sun, L., Cheng, H., Xu, R., Shen, H., Zhu, X., Shi, X., Xu, K.: Act-diffusion: Efficient adversarial consistency training for one-step diffusion models. In: CVPR (2024)
32. Kousha, S., Maleky, A., Brown, M.S., Brubaker, M.A.: Modeling srgb camera noise with normalizing flows. In: CVPR (2022)
33. Laine, S., Karras, T., Lehtinen, J., Aila, T.: High-quality self-supervised deep image denoising. NeurIPS (2019)
34. Lee, S., Kim, T.H.: Noisetransfer: Image noise generation with contrastive embeddings. In: ACCV (2022)
35. Lee, W., Son, S., Lee, K.M.: Ap-bsn: Self-supervised denoising for real-world images via asymmetric pd and blind-spot network. In: CVPR (2022)
36. Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Swinir: Image restoration using swin transformer. In: ICCV (2021)
37. Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., Han, J.: On the variance of the adaptive learning rate and beyond. In: ICLR (2020)
38. Liu, P., Zhang, H., Zhang, K., Lin, L., Zuo, W.: Multi-level wavelet-cnn for image restoration. In: CVPRW (2018)
39. Liu, Y., Anwar, S., Qin, Z., Ji, P., Caldwell, S., Gedeon, T.: Disentangling noise from images: A flow-based image denoising neural network. Sensors (2022)
40. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. arXiv (2017)
41. Lu, C., Song, Y.: Simplifying, stabilizing and scaling continuous-time consistency models. In: ICLR (2025)
42. Luo, S., Tan, Y., Huang, L., Li, J., Zhao, H.: Latent consistency models: Synthesizing high-resolution images with few-step inference. arXiv (2023)
43. Michaelis, C., Mitzkus, B., Geirhos, R., Rusak, E., Bringmann, O., Ecker, A.S., Bethge, M., Brendel, W.: Benchmarking robustness in object detection: Autonomous driving when winter is coming. arXiv (2019)
44. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv (2018)
45. Plotz, T., Roth, S.: Benchmarking denoising algorithms with real photographs. In: CVPR (2017)
46. Ramanath, R., Snyder, W.E., Yoo, Y., Drew, M.S.: Color image processing pipeline. IEEE Signal processing magazine (2005)
47. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022)

48. Sakaridis, C., Dai, D., Gool, L.V.: Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In: ICCV (2019)
49. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. NeurIPS (2016)
50. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: ICML (2015)
51. Song, Y., Dhariwal, P.: Improved techniques for training consistency models. In: ICLR (2024)
52. Song, Y., Dhariwal, P., Chen, M., Sutskever, I.: Consistency models. In: ICML (2023)
53. Song, Y., Ermon, S.: Generative modeling by estimating gradients of the data distribution. NeurIPS (2019)
54. Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q.: Eca-net: Efficient channel attention for deep convolutional neural networks. In: CVPR (2020)
55. Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., Li, H.: Uformer: A general u-shaped transformer for image restoration. In: CVPR (2022)
56. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. TIP (2004)
57. Wang, Z., Fu, Y., Liu, J., Zhang, Y.: Lg-bpn: Local and global blind-patch network for self-supervised real-world denoising. In: CVPR (2023)
58. Wei, C., Wang, W., Yang, W., Liu, J.: Deep retinex decomposition for low-light enhancement. arXiv (2018)
59. Wu, X., Liu, M., Cao, Y., Ren, D., Zuo, W.: Unpaired learning of deep image denoising. In: ECCV (2020)
60. Yue, Z., Zhao, Q., Zhang, L., Meng, D.: Dual adversarial network: Toward real-world noise removal and noise generation. In: ECCV (2020)
61. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H.: Restormer: Efficient transformer for high-resolution image restoration. In: CVPR (2022)
62. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H., Shao, L.: Learning enriched features for real image restoration and enhancement. In: ECCV (2020)
63. Zhang, B., Sennrich, R.: Root mean square layer normalization. NeurIPS (2019)
64. Zhang, D., Zhou, F., Jiang, Y., Fu, Z.: Mm-bsn: Self-supervised image denoising for real-world with multi-mask based on blind-spot network. In: CVPRW (2023)
65. Zhang, K., Li, Y., Zuo, W., Zhang, L., Van Gool, L., Timofte, R.: Plug-and-play image restoration with deep denoiser prior. TPAMI (2021)
66. Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L.: Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. TIP (2017)
67. Zhang, K., Zuo, W., Gu, S., Zhang, L.: Learning deep cnn denoiser prior for image restoration. In: CVPR (2017)
68. Zhang, K., Zuo, W., Zhang, L.: Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. TIP (2018)
69. Zheng, D., Zou, Y., Zhang, X., Bao, C.: Senm-vae: Semi-supervised noise modeling with hierarchical variational autoencoder. In: CVPR (2024)