Effective Cloud Removal for Remote Sensing Images by an Improved Mean-Reverting Denoising Model with Elucidated Design Space

Yi Liu[1], Wengen Li[1*], Jihong Guan[1*], Shuigeng Zhou[2], Yichao Zhang[1]
[1] Tongji University, [2] Fudan University
{liuyi61,lwengen,jhguan,yichaozhang}@tongji.edu.cn, sgzhou@fudan.edu.cn

Abstract

Cloud removal (CR) remains a challenging task in remote sensing image processing. Although diffusion models (DM) exhibit strong generative capabilities, their direct applications to CR are suboptimal, as they generate cloudless images from random noise, ignoring inherent information in cloudy inputs. To overcome this drawback, we develop a new CR model ***EMRDM*** *based on mean-reverting diffusion models (MRDMs) to establish a direct diffusion process between cloudy and cloudless images. Compared to current MRDMs, EMRDM offers a modular framework with updatable modules and an elucidated design space, based on a reformulated forward process and a new ordinary differential equation (ODE)-based backward process. Leveraging our framework, we redesign key MRDM modules to boost CR performance, including restructuring the denoiser via a preconditioning technique, reorganizing the training process, and improving the sampling process by introducing deterministic and stochastic samplers. To achieve multi-temporal CR, we further develop a denoising network for simultaneously denoising sequential images. Experiments on mono-temporal and multi-temporal datasets demonstrate the superior performance of EMRDM. Our code is available at* https://github.com/Ly403/EMRDM.

1. Introduction

Satellite imagery, as a fundamental remote sensing product [68, 72], enables diverse applications including environmental monitoring [59], land cover classification [34], and agricultural monitoring [48]. However, cloud coverage severely affects the usability of satellite imagery. Data analysis for the Moderate Resolution Imaging Spectroradiometer (MODIS) on the Terra and Aqua satellites indicates that about 67% of the Earth's surface experiences cloud coverage [33]. Hence, cloud removal (CR) is a critical preliminary step in processing satellite imagery.

*Corresponding author.

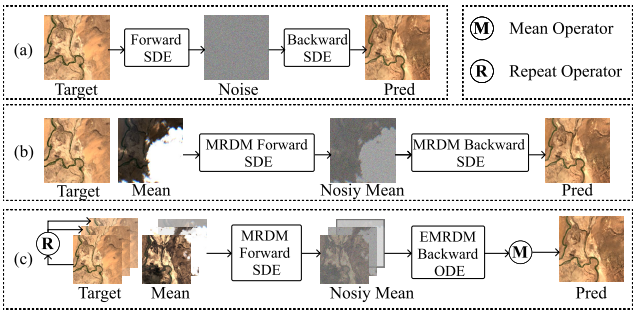


Figure 1. Comparison of EMRDM (c) with generative DMs (a) and MRDMs (b). Here, *target* is the cloudless image, *pred* is the CR prediction result, *mean* is the cloudy image, and *noisy mean* is the noisy cloudy image. The forward processes of (a), (b), and (c) generate diffused images approximated by *noise* (for DMs) and *noisy mean* (for EMRDM and MRDMs), respectively.

Recent advances in deep learning have driven the progress of CR [62], with generative adversarial networks (GANs) [20] becoming a predominant approach. However, the effectiveness of GANs in CR is undermined by training instability [51] and mode collapse [3]. In comparison, diffusion models (DMs) [23, 55, 56] can overcome these limitations via enhanced training stability and output diversity, setting new benchmarks in image synthesis [11] and restoration [36]. Such advantages of DMs also extend to CR tasks [29, 58, 71, 74].

Existing diffusion-based CR methods typically employ vanilla DM frameworks that start the diffusion process from pure noise (Fig. 1 (a)). However, this is unnecessary as cloudy images contain substantial unexploited information. Even worse, noise-initiated generation lacks pixel-level consistency, inducing distortion [6] in restored images due to poor fine-grained controllability. To resolve this, we propose the integration of mean-reverting diffusion models (MRDMs) [41] into CR. MRDMs start the diffusion process directly from noisy cloudy images (Fig. 1 (b)), intrinsically preserving structural fidelity through pixel-level consistency constraints. Specifically, the forward process pro-

gressively diffuses the target image by injecting noise while maintaining the cloudy image as the distribution mean, yielding a noisy cloudy image (*noisy mean*). Subsequent denoising in the backward process reconstructs the cloudless image (*pred*) while preserving structural consistency.

However, current MRDMs exhibit limitations due to their intricately coupled modules and opaque relationships among modules, impeding their application. Inspired by the successful designs of EDM [31] in image generation, we conduct an in-depth analysis of the underlying mathematical principles of MRDMs to clarify the roles and interrelationships of modules within the MRDM framework. Based on these insights, we **E**lucidate the design space of **MRDM**s and propose a novel MRDM-based CR model, termed **EMRDM**. EMRDM offers a modular framework by reformulating the forward process through a stochastic differential equation (SDE) with simplified parameters and introducing an ordinary differential equation (ODE)-based backward process, as illustrated in Fig. 1 (c). The framework offers two critical advantages: (1) an elucidated and flexible design space enabling orthogonal module modifications, and (2) seamless compatibility with generative DMs. Leveraging the advantages of our framework, we further redesign key MRDM modules to boost CR performance, focusing on the following enhancements: (1) We restructure the denoiser via a preconditioning technique, inspired by image generation methods [31, 57], to adaptively scale inputs and outputs of the denoising network according to noise levels. (2) We reorganize the training process and improve the sampling process. For practical sampling of CR results, we introduce novel deterministic and stochastic samplers based on the improved sampling process.

To achieve multi-temporal CR, we further develop a denoising network that processes arbitrary-length image sequences. Specifically, for L sequential cloudy images, our architecture employs L weight-sharing encoders and bottleneck modules, compresses temporal features through a novel attention block, and reconstructs outputs via a single decoder. The generated attention masks are preserved and upsampled to various resolutions, serving as adaptive weights to fuse temporal skip feature maps. The preconditioning and training methods are modified to accommodate multi-temporal scenarios through sequential input compatibility optimization. During sampling, to ensure temporal restoration consistency, we independently restore each temporal instance under mono-temporal conditions and aggregate results through a mean fusion operator (Fig. 1 (c)).

Our **contributions** are summarized as follows:

1) We propose a novel CR model **EMRDM** that offers a modular framework with updatable modules and an elucidated design space.
2) We develop a multi-temporal network with a temporal fusion method to denoise arbitrary-length image sequences.
3) We restructure the denoiser via a preconditioning method, improve training and sampling processes, and propose novel stochastic and deterministic samplers.
4) Experiments on mono-temporal and multi-temporal cases demonstrate the superior CR performance of EMRDM.

2. Related Work

Cloud Removal. CR methods are primarily divided into traditional methods [37, 64, 65] and deep learning-based methods, with the former offering better interpretability but generally inferior performance compared to data-driven methods. Deep learning-based methods are further categorized into mono-temporal [4, 15–17, 21, 35, 43, 45, 63, 74] and multi-temporal [14, 15, 24, 52, 71, 74] paradigms based on single-image or sequential inputs. Mono-temporal methods commonly employ vanilla conditional GANs (cGANs) [20, 44] in early applications [4, 16, 21], with improvements including spatial attention [45] and transformer architectures [35]. Alternative frameworks include DMs [74] and non-generative models [15, 43, 63]. Multi-temporal strategies mainly use temporal cGAN [24, 52], temporal fusion attention (*e.g.*, L-TAE [18, 19]) as in [15], and sequential DMs [71]. CR methods are also classified as mono-modal [16, 45, 74] or multi-modal, depending on the use of auxiliary modalities, including infrared (IR) and synthetic aperture radar (SAR) images. Multi-modal methods involve modality concatenation [4, 14, 15, 21, 24, 43, 52] and specialized fusion modules [17, 63, 71].

Diffusion Models. Recent advances in generative modeling have witnessed DMs [23, 55, 56] surpass GANs [11] in image synthesis. Notable improvements to DMs [9, 31, 46, 47] have also been proposed, with EDM [31] and HDiT [9] most crucial to our work. EDM presents a framework that delineates the specific design decisions for DM components, while HDiT introduces an efficient hourglass diffusion transformer. Inspired by the success of DMs in image generation, extensive studies have investigated their applications in image restoration [36]. These methods can be categorized as supervised learning [1, 10, 38, 39, 41, 42, 49, 50, 61, 69] or zero-shot learning [8, 32, 40, 54, 60]. In the first category, several methods focus on generating images directly from noiseless or noisy corrupted images, such as IR-SDE [41], InDI [10], ResShift [69], RDDM [39], and I2SB [38]. Considering that starting with pure noise is inefficient, IR-SDE, InDI, ResShift, and RDDM all integrate the corrupted image and noise within the diffusion process. We extend this paradigm and apply it to CR.

3. Methodology

As illustrated in Fig. 2, we introduce the EMRDM framework in Sec. 3.2, propose a novel multi-temporal denoising network in Sec. 3.3, restructure the denoiser by the precon-

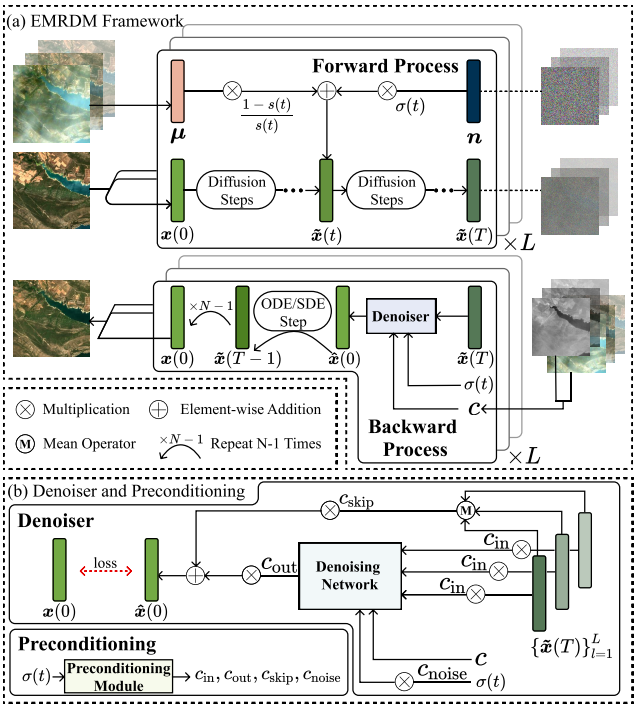


Figure 2. (a) The EMRDM framework comprises a forward process and a backward process that contains a denoiser. (b) The denoiser consists primarily of a denoising network, where the preconditioning module generates reparameterized factors $c_{\text{in}}(\sigma), c_{\text{out}}(\sigma), c_{\text{skip}}(\sigma), c_{\text{noise}}(\sigma)$ based on noise level $\sigma(t)$. We show the multi-temporal condition with the sequence length L.

ditioning technique in Sec. 3.4, and present our redesigned training and sampling process in Sec. 3.5.

3.1. Preliminary

The forward process of DMs can be expressed as an SDE proposed by Song *et al*. (Eq. 5 in [56]), as follows:

$$\mathrm{d}\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x}, t)\mathrm{d}t + g(t)\mathrm{d}\boldsymbol{\omega}_t, \tag{1}$$

where $\boldsymbol{\omega}_t$ is a standard Brownian motion, $\boldsymbol{x} \in \mathbb{R}^d$ is an Itô process, $\boldsymbol{f}(\cdot, t) : \mathbb{R}^d \to \mathbb{R}^d$ and $g(\cdot) : \mathbb{R} \to \mathbb{R}$ are the drift and diffusion coefficients, respectively, and d is the dimensionality of images. Song *et al*. further derive a reverse probability flow ODE (Eq. 13 in [56]) for sampling as

$$\mathrm{d}\boldsymbol{x} = \left[\boldsymbol{f}(\boldsymbol{x}, t) - \frac{1}{2}g(t)^2\nabla_{\boldsymbol{x}}\log p_t(\boldsymbol{x})\right]\mathrm{d}t, \tag{2}$$

where $p_t(\boldsymbol{x})$ is the probability density function (pdf) of $\boldsymbol{x}$ at time t. The score function $\nabla_{\boldsymbol{x}}\log p_t(\boldsymbol{x})$ is predicted by a neural network. Therefore, the models proposed by [56], as well as our models, are *score matching* models.

3.2. The EMRDM Framework

We reformulate the forward process of MRDMs to construct a stochastic process $\{\boldsymbol{x}(t)\}_{t=0}^{T}$ that transforms a target image into its noisy cloudy counterpart. The new ODE-based backward process iteratively denoises the corrupted images.

Forward Process. We transform the SDE in Eq. (1) into

$$\mathrm{d}\boldsymbol{x} = f(t)(\boldsymbol{x} - \boldsymbol{\mu})\mathrm{d}t + g(t)\mathrm{d}\boldsymbol{\omega}_t, \tag{3}$$

where $\boldsymbol{\mu} \in \mathbb{R}^d$ is the cloudy image, and the stochastic process $\boldsymbol{x}(t)$ is simplified to $\boldsymbol{x}$. According to [41], Eq. (3) can be viewed as a special case of Eq. (1) by defining $\boldsymbol{f}(\boldsymbol{x}, t) = f(t)(\boldsymbol{x} - \boldsymbol{\mu})$. This setting yields a solution for the pdf of $\boldsymbol{x}(t)$ given $\boldsymbol{x}(0)$ and $\boldsymbol{\mu}$:

$$p_{0t}\big(\boldsymbol{x}(t) \mid \boldsymbol{x}(0), \boldsymbol{\mu}\big) = s(t)^{-d}\tilde{p}_{0t}\big(\tilde{\boldsymbol{x}}(t) \mid \tilde{\boldsymbol{x}}_0(t)\big), \tag{4}$$

$$\tilde{p}_{0t}\big(\tilde{\boldsymbol{x}}(t) \mid \tilde{\boldsymbol{x}}_0(t)\big) = \mathcal{N}\left(\frac{\boldsymbol{x}(t)}{s(t)}; \tilde{\boldsymbol{x}}_0(t), \sigma(t)^2\boldsymbol{I}\right), \tag{5}$$

$$\tilde{\boldsymbol{x}}_0(t) = \boldsymbol{x}(0) + \frac{1 - s(t)}{s(t)}\boldsymbol{\mu}, \tag{6}$$

where $\mathcal{N}(\boldsymbol{x}; \boldsymbol{m}, \boldsymbol{\Sigma})$ denotes the Gaussian pdf evaluated at $\boldsymbol{x}$, with mean $\boldsymbol{m}$ and covariance $\boldsymbol{\Sigma}$. We define $\tilde{\boldsymbol{x}}(t) = \boldsymbol{x}(t)/s(t)$. The values of $s(t)$ and $\sigma(t)$ are as follows:

$$s(t) = \exp\left(\int_0^t f(\xi)\mathrm{d}\xi\right), \sigma(t) = \sqrt{\int_0^t \frac{g(\xi)^2}{s(\xi)^2}\mathrm{d}\xi}. \tag{7}$$

In our framework, $s(t)$ and $\sigma(t)$ are used instead of $f(t)$ and $g(t)$ for the design simplicity. By introducing the mean-adding term, *i.e.*, $\frac{1-s(t)}{s(t)}\boldsymbol{\mu}$, in Eq. (6), the mean of $\tilde{\boldsymbol{x}}(t)$ approximately shifts to $\boldsymbol{\mu}$, unlike generative DMs with a final mean of zero. Hence, the SDE in Eq. (3) is named the mean-reverting SDE. Concretely:

- At $t = 0$, it is obvious that $s(0) = 1$ and $\sigma(0) = 0$, ensuring $\tilde{\boldsymbol{x}}(0) = \tilde{\boldsymbol{x}}_0(0) = \boldsymbol{x}(0)$.
- At a large $t = T$, we require $\frac{1-s(T)}{s(T)}$ to be large enough to obscure $\boldsymbol{x}(0)$, ensuring that $\tilde{\boldsymbol{x}}(T)$ has a mean almost proportional to $\boldsymbol{\mu}$ and a standard variance equal to $\sigma(T)$.

With the techniques above, we establish a diffusion process that bridges the target image $\boldsymbol{x}(0)$ and the cloudy image $\boldsymbol{\mu}$ with noise $\boldsymbol{n}$, ensuring pixel-level fidelity in CR outputs. Notably, by omitting the mean-adding term (*i.e.*, setting $s(t) = 1$), the EMRDM framework reduces to the generative DM in [31]. Hence, our framework expands the boundary of generative DMs.

See Appendix A.1 for derivations of the forward process.

Backward Process. We use $s(t)$ and $\sigma(t)$ to derive the backward ODE. Based on Eq. (2), we have

$$\mathrm{d}\tilde{\boldsymbol{x}}(t) = \left[-\frac{\dot{s}(t)}{s(t)^2}\boldsymbol{\mu} - \dot{\sigma}(t)\sigma(t)s_\theta\big(\tilde{\boldsymbol{x}}(t)\big)\right]\mathrm{d}t, \tag{8}$$

where $s_\theta\big(\tilde{\boldsymbol{x}}(t)\big) = \nabla_{\tilde{\boldsymbol{x}}(t)}\log p_t\big(\tilde{\boldsymbol{x}}(t)\big)$ is the score function [27], a vector field pointing to the higher density of

data, with θ as its parameters. As $s_\theta\big(\tilde{\boldsymbol{x}}(t)\big)$ does not depend on the intractable form of $\log p_t\big(\tilde{\boldsymbol{x}}(t)\big)$ [27], it can be easily calculated. We use a denoiser function $D_\theta(\boldsymbol{x};\sigma;c)$ to predict it, with $\boldsymbol{x}$ as the image input, σ as the noise level input, and c as the conditioning input. By training D_θ as follows:

$$\begin{aligned} L\big(D_\theta,\sigma(t)\big) =& \mathbb{E}_{\tilde{\boldsymbol{x}}_0(t)\sim p_{\text{data}}}\mathbb{E}_{\boldsymbol{n}\sim\mathcal{N}\left(0,\sigma(t)^2\boldsymbol{I}\right)} \\ & \left\|D_\theta\big(\tilde{\boldsymbol{x}}_0(t)+\boldsymbol{n};\sigma(t);c\big)-\tilde{\boldsymbol{x}}_0(0)\right\|_2^2, \end{aligned} \tag{9}$$

with p_{data} as the distribution of $\tilde{\boldsymbol{x}}_0(t)$, we can acquire

$$s_\theta\big(\tilde{\boldsymbol{x}}(t)\big)=\frac{D_\theta\big(\tilde{\boldsymbol{x}}(t);\sigma(t);c\big)+\frac{1-s(t)}{s(t)}\boldsymbol{\mu}-\tilde{\boldsymbol{x}}(t)}{\sigma(t)^2}. \tag{10}$$

Though it is common to directly use a neural network as the denoiser D_θ, it is suboptimal for stable and effective training, as explained in Sec. 3.3. Hence, as shown in Fig. 2, we restructure D_θ by training a different network F_θ via the preconditioning technique. Sec. 3.3 provides details on F_θ, while the relationship between F_θ and D_θ is discussed in Sec. 3.4. By substituting Eq. (10) into Eq. (8), we obtain

$$\begin{aligned} \mathrm{d}\tilde{\boldsymbol{x}}(t)=&\left[-\frac{\dot{s}(t)}{s(t)^2}\boldsymbol{\mu}-\frac{\dot{\sigma}(t)}{\sigma(t)}\times\right. \\ &\left.\left(D_\theta\big(\tilde{\boldsymbol{x}}(t);\sigma(t);c\big)+\frac{1-s(t)}{s(t)}\boldsymbol{\mu}-\tilde{\boldsymbol{x}}(t)\right)\right]\mathrm{d}t. \end{aligned} \tag{11}$$

See Appendix A.2 for the proof of Eqs. (8) to (11). We redesign the samplers based on this ODE, detailed in Sec. 3.5. Generally, as depicted in Fig. 2 (a), at time T, the samplers iteratively use D_θ to estimate $\boldsymbol{x}(0)$. The output $\hat{\boldsymbol{x}}(0)$ is used in Eq. (11) to compute the next-step image $\tilde{\boldsymbol{x}}(T-1)$ for N steps, ultimately restoring the image.

Choices of $s(t)$ and $\sigma(t)$. It is essential to ensure $\sigma(0)=0$ and $\lim_{t\to 0}\frac{1-s(t)}{s(t)}=0$. We adopt the linear choice $\sigma(t)=t$ according to [31], and set $s(t)=\frac{1}{1+\alpha t}$, where α controls the mean reversion rate. Such settings yield a simpler SDE parameterization compared to prior MRDMs [41].

3.3. Multi-temporal Denoising Network

Network Architecture. For mono-temporal CR, previous improvements to the denoising network [2, 9, 46] can be directly used, as it is orthogonal to other modules. We choose HDiT [9] for effectiveness and efficiency. To adapt HDiT to CR tasks, we reset the input channels and remove the non-leaking augmentation [30] and classifier-free guidance [22], as they are unsuitable for restoration. Following [49], we concatenate the noisy cloudy image $\tilde{\boldsymbol{x}}_0(t)+n$ with the condition c. The condition includes cloudy images and optional auxiliary modal images (*e.g.*, SAR or IR images).

To extend HDiT to multi-temporal CR tasks, we propose a new denoising network based on UTAE [19] to denoise sequential images. As shown in Fig. 3 (a), we retain the main architecture of HDiT and create L weight-sharing copies of the encoder and middle HDiT blocks 3, while keeping the decoder unchanged. In the bottleneck module, we introduce a temporal HDiT block (THDiT), allowing sequential feature maps to be condensed into one map. Attention masks are generated from THDiT and used to collapse the temporal dimension of the skip feature maps per resolution:

$$o^i=\text{Concat}\left[\sum_{l=1}^{L}\text{bilinear}(a_l^g,i)\odot e_l^{i,g}\right]_{g=1}^{G}, \tag{12}$$

where o^i is the output skipping feature map to the decoder at resolution level i, a_l^g is the attention mask at head g and time l, $e_l^{i,g}$ is the input feature map from the encoder at head g, time l and resolution level i, G is the number of heads, $\odot$ is the element-wise multiplication, and bilinear$(\cdot,i)$ indicates upsampling the map from the lowest resolution to level i.

Temporal HDiT Block. THDiT is modified from the original HDiT block. As shown in Fig. 3 (b), we replace spatial attention with our proposed temporal fusion self-attention (TFSA) to merge sequential feature maps and generate attention masks. We also introduce rearrangement layers to ensure that the feature maps have the correct shape before entering different blocks. As the temporal dimension collapses after TFSA, we remove the residual connection.

Temporal Fusion Self-Attention. As shown in Fig. 3 (c), TFSA adopts vanilla multi-head self-attention. Following L-TAE [18], we define query, key and value matrices as $\boldsymbol{Q}\in\mathbb{R}^{1\times d_k}, \boldsymbol{K}=\boldsymbol{X}\boldsymbol{W}\in\mathbb{R}^{L\times d_k}, \boldsymbol{V}=\boldsymbol{X}\in\mathbb{R}^{L\times C}$, respectively. Here, we consider a single-head scenario and omit the batch size dimension for simplicity. The feature map $\boldsymbol{X}$ has a sequence length of L and C channels. Both $\boldsymbol{Q}$ and $\boldsymbol{K}$ have d_k channels. We use $\boldsymbol{X}$ as $\boldsymbol{V}$, and project it to $\boldsymbol{K}$ with weights $\boldsymbol{W}\in\mathbb{R}^{C\times d_k}$. $\boldsymbol{Q}$ is set as a learnable parameter and initialized from a normal distribution, with a sequence length of 1 to condense the temporal information.

3.4. Preconditioning

In this section, we restructure the denoiser via the preconditioning technique to adaptively scale inputs and outputs according to noise variance $\sigma(t)$, focusing on multi-temporal CR, with the mono-temporal case covered by setting $L=1$. We use the superscript l to represent the time point.

For training a network, it is advisable to maintain both inputs and outputs with unit variance [5, 25], thus stabilizing and enhancing the training process. While directly training denoiser D_θ is not ideal for this purpose, we train a network F_θ instead via the preconditioning technique to scale inputs and outputs to unit variance, following EDM [31]. As shown in Fig. 2 (b), the relation between D_θ and F_θ is:

$$\begin{aligned} D_\theta\left(\{\tilde{\boldsymbol{x}}^l\}_{l=1}^L;\sigma;c\right)=&\,\text{mean}\left(\{c_{\text{skip}}(\sigma)\,\tilde{\boldsymbol{x}}^l\}_{l=1}^L\right) \\ &+c_{\text{out}}(\sigma)\,F_\theta\left(\{c_{\text{in}}(\sigma)\,\tilde{\boldsymbol{x}}^l\}_{l=1}^L;c_{\text{noise}}(\sigma);c\right), \end{aligned} \tag{13}$$

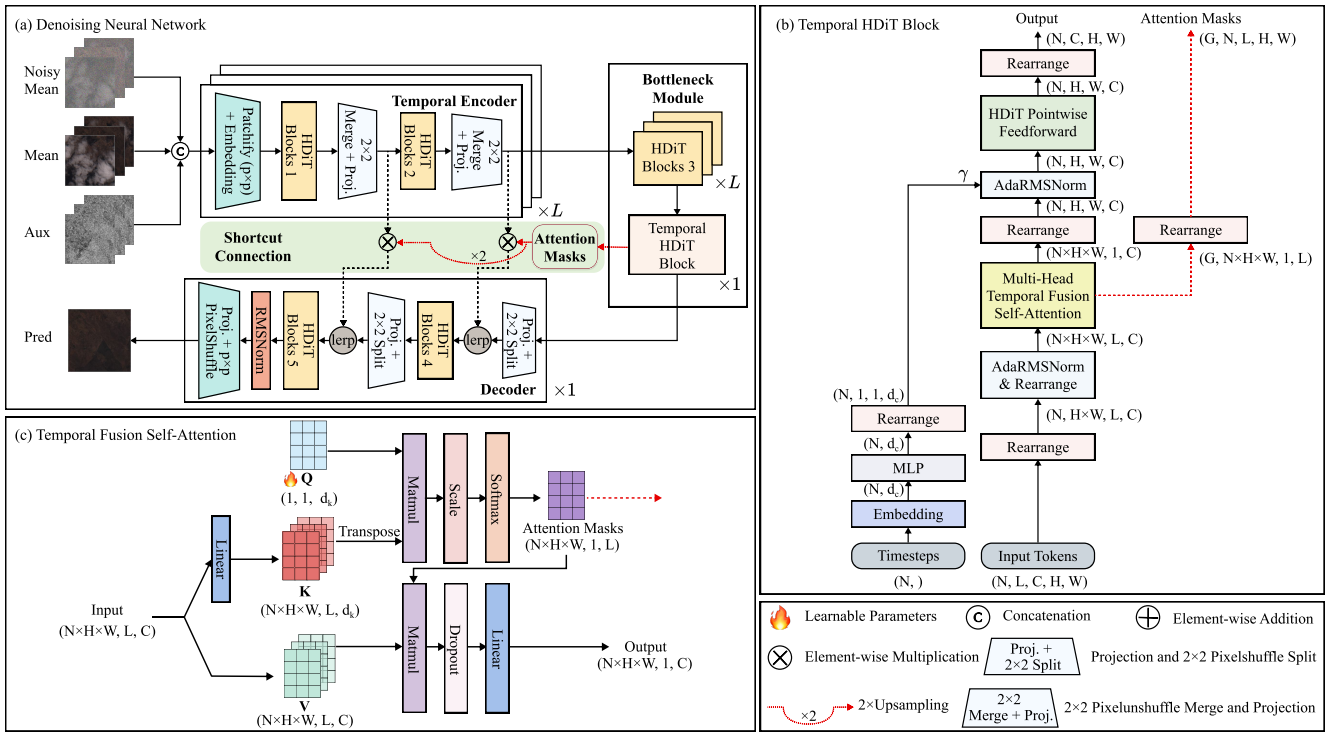


Figure 3. Illustration of the denoising network. (a) The network concurrently denoises sequences of noisy cloudy images (*noisy mean*), cloudy images (*mean*), and optional auxiliary modal images (*aux*) to generate results (*pred*). The notation $\times L$ indicates L weight-sharing copies. (b) We extend the original HDiT Blocks to THDiT Blocks to integrate temporal information. (c) TFSA collapses the temporal dimension of inputs and generates the attention masks. For simplicity, we present a single-head scenario. Feature map dimensions are indicated below each block, where N is the batch size, H is the height, W is the width, L is the sequence length, C is the channels of feature maps, G is the number of heads, d_c is the channels of condition vectors, and d_k is the channels of query and key matrices.

where $\sigma(t)$ is simplified to σ and $\tilde{\boldsymbol{x}}(t)^l$ is simplified to $\tilde{\boldsymbol{x}}^l$. The output shape of F_θ differs from the input shape, which requires a mean operator to reduce the temporal dimension of $\{c_{\text{skip}}(\sigma)\,\tilde{\boldsymbol{x}}^l\}_{l=1}^L$. As our network can process sequential images, $\{c_{\text{in}}(\sigma)\,\tilde{\boldsymbol{x}}^l\}_{l=1}^L$ does not need the mean operator. To ensure that inputs and targets have unit variance, we introduce four factors $c_{\text{in}}(\sigma)$, $c_{\text{skip}}(\sigma)$, $c_{\text{out}}(\sigma)$ and $c_{\text{noise}}(\sigma)$ to scale the inputs and outputs governed by four hyperparameters: σ_{data} (the variance of target images), σ_{mu} (the variance of cloudy images), σ_{cov} (the covariance between target and cloudy images), and L (sequence length):

$$c_{\text{in}}(\sigma) = \frac{1}{\sqrt{\sigma_{\text{data}}^2 + k^2\sigma_{\text{mu}}^2 + \sigma^2 + 2k\sigma_{\text{cov}}}}, \tag{14}$$

$$c_{\text{skip}}(\sigma) = \frac{\sigma_{\text{data}}^2 + k\sigma_{\text{cov}}}{\sigma_{\text{data}}^2 + k^2\sigma_{\text{mu}}^2 + \frac{\sigma^2}{L} + 2k\sigma_{\text{cov}}}, \tag{15}$$

$$c_{\text{out}}(\sigma) = \sqrt{\frac{k^2\sigma_{\text{mu}}^2\sigma_{\text{data}}^2 + \frac{\sigma^2}{L}\sigma_{\text{data}}^2 - k^2\sigma_{\text{cov}}^2}{\sigma_{\text{data}}^2 + k^2\sigma_{\text{mu}}^2 + \frac{\sigma^2}{L} + 2k\sigma_{\text{cov}}}}, \tag{16}$$

$$c_{\text{noise}}(\sigma) = \frac{1}{4}\ln(\sigma), \tag{17}$$

where k represents $k(t)$, and $k(t) = \frac{1-s(t)}{s(t)}$. Notably, setting $\sigma_{\text{mu}} = \sigma_{\text{cov}} = 0$ reverts Eqs. (14) to (17) to their original form in EDM. See Appendix A.3 for derivations.

3.5. Training and Sampling

This section details the training and sampling processes under the multi-temporal scenario, with the mono-temporal case covered by setting $L = 1$.

Training. The training process is detailed in Algorithm 1. We retain the training distribution of σ in [31] (line 2). Sequential images are then independently perturbed (lines 4 to 6) and denoised jointly (line 7). We further introduce a parameter $\lambda(\sigma)$ to adjust the loss function at different noise levels during training (line 9):

$$\mathbb{E}_{\sigma,\boldsymbol{x}(0),\boldsymbol{n}}\left[\lambda\left\|D_\theta\left(\{\tilde{\boldsymbol{x}}_0^l + \boldsymbol{n}\}_{l=1}^L;\sigma,c\right) - \boldsymbol{x}(0)\right\|_2^2\right], \tag{18}$$

where λ and $\tilde{\boldsymbol{x}}_0^l$ represent $\lambda(\sigma)$ and $\tilde{\boldsymbol{x}}_0^l(t)$, respectively. We set $\lambda(\sigma) = \frac{1}{c_{\text{out}}(\sigma)^2}$, in accordance with EDM [31].

Sampling. As outlined in Algorithm 2, we design a stochastic sampler. It begins with the sequential sampling of noisy images (lines 2 to 3). Within the sampling loop, γ_i is computed (line 5) to perturb the time t_i to a higher noise level $\hat{t}_i$

Algorithm 1 Our training step with $s(t) = 1/(1+\alpha t)$ and $\sigma(t) = t$.

procedure TRAINSTEP($\boldsymbol{x}(0), \{\boldsymbol{\mu}^l\}_{l=1}^L, c, D_\theta$)
  sample $\ln(\sigma) \sim \mathcal{N}(P_{\text{mean}}, P_{\text{std}}^2)$
  $\sigma \leftarrow \exp\left(\ln(\sigma)\right)$
  for $l \in \{1, 2, \cdots, L\}$ **do**
    sample $\boldsymbol{n}^l \sim \mathcal{N}(0, \boldsymbol{I})$
    $\tilde{\boldsymbol{x}}_0^l(t) \leftarrow \boldsymbol{x}(0) + \alpha\sigma\boldsymbol{\mu}^l, \tilde{\boldsymbol{x}}^l(t) \leftarrow \tilde{\boldsymbol{x}}_0^l(t) + \sigma\boldsymbol{n}^l$
  $\hat{\boldsymbol{x}}(0) \leftarrow D_\theta\left(\{\tilde{\boldsymbol{x}}^l(t)\}_{l=1}^L; \sigma; c\right)$ ▷ Eq. (13)
  Take gradient descent step on
    $\nabla_{\boldsymbol{x}} \mathbb{E}_{\sigma, \boldsymbol{x}(0), \boldsymbol{n}} \left[\lambda(\sigma)\|\hat{\boldsymbol{x}}(0) - \boldsymbol{x}(0)\|_2^2\right]$ ▷ Eq. (18)

Algorithm 2 Our stochastic sampler with $s(t) = 1/(1+\alpha t)$ and $\sigma(t) = t$.

1: **procedure** STOCHASTICSAMPLER($\{\boldsymbol{\mu}^l\}_{l=1}^L, c, D_\theta$)
2: **for** $l \in \{1, 2, \cdots, L\}$ **do**
3: **sample** $\boldsymbol{x}_0^l \sim \mathcal{N}(\alpha\sigma\boldsymbol{\mu}^l, \sigma^2\boldsymbol{I})$
4: **for** $i \in \{0, 1, \cdots, N-1\}$ **do**
5: $\gamma_i \leftarrow S_{\text{churn}}/N$ **if** $t_i \in [S_{\text{tmin}}, S_{\text{tmax}}]$ **else** 0
6: $\hat{t}_i \leftarrow t_i + \gamma_i t_i$
7: **for** $l \in \{1, 2, \cdots, L\}$ **do**
8: **sample** $\boldsymbol{\epsilon}_i^l \in \mathcal{N}\left(0, S_{\text{noise}}^2\boldsymbol{I}\right)$
9: $\hat{\boldsymbol{x}}_i^l \leftarrow \boldsymbol{x}_i^l + \alpha(\hat{t}_i - t_i)\boldsymbol{\mu}^l + \sqrt{\hat{t}_i^2 - t_i^2}\boldsymbol{\epsilon}_i^l$ ▷ Eq. (19)
10: **for** $l \in \{1, 2, \cdots, L\}$ **do**
11: $\boldsymbol{d}_i^l \leftarrow \left(\hat{\boldsymbol{x}}_i^l - D_\theta(\{\hat{\boldsymbol{x}}_i^l\}_{l=1}^L; \sigma; c)\right) / \hat{t}_i$ ▷ Eq. (11)
12: $\boldsymbol{x}_{i+1}^l \leftarrow \hat{\boldsymbol{x}}_i^l + (t_{i+1} - \hat{t}_i)\boldsymbol{d}_i^l$
13: $\boldsymbol{x}_N \leftarrow \text{mean}\left(\{\boldsymbol{x}_N^l\}_{l=1}^L\right)$
14: **return** $\boldsymbol{x}_N$

(line 6). Updated samples $\hat{\boldsymbol{x}}_i^l$ at noise level $\hat{t}_i$ are obtained:

$$\hat{\boldsymbol{x}}_i^l = \boldsymbol{x}_i^l + \left(k(\hat{t}_i) - k(t_i)\right)\boldsymbol{\mu}^l + \sqrt{\sigma(\hat{t}_i)^2 - \sigma(t_i)^2}\boldsymbol{\epsilon}_i^l, \quad (19)$$

where $\boldsymbol{\epsilon}_i^l$ denotes Gaussian noise. The Euler step (lines 10 to 12) based on Eq. (11) computes the next sample $\boldsymbol{x}_{i+1}^l$ for each l. The loop ends with a mean operator to collapse the temporal dimension of $\{\boldsymbol{x}_N^l\}_{l=1}^L$. The method includes following hyperparameters: N, S_{churn}, S_{tmin}, S_{tmax} and S_{noise}, as in EDM. N is the number of sample steps. S_{churn}, S_{tmin} and S_{tmax} control γ_i, while S_{noise} regulates the variance of $\boldsymbol{\epsilon}_i^l$. The stochastic sampler becomes deterministic when setting $S_{\text{churn}} = 0$. In addition, we should set a range for σ when sampling. In other words, $\sigma(t_{N-1}) = \sigma_{\text{max}}$ and $\sigma(t_0) = \sigma_{\text{min}}$. Both σ_{max} and σ_{min} are also hyperparameters. The intermediate σ values are interpolated following EDM (Eq. 5 in [31]). See Appendix A.4 for more details.

4. Performance Evaluation

4.1. Implementation Details

We conduct experiments on four datasets: CUHK-CR1 [58], CUHK-CR2 [58] and SEN12MS-CR [13] for mono-temporal CR tasks; and Sen2_MTC_New [24] for multi-temporal CR tasks with $L = 3$. MAE, PSNR, SSIM, SAM, and LPIPS are used as evaluation metrics. We move more implementation details to Appendix C.1.

Table 1. Quantitative results on (a) CUHK-CR1, (b) CUHK-CR2, (c) SEN12MS-CR, and (d) Sen2_MTC_New datasets. The metrics align with those used in prior studies on these datasets. The symbols ↑/↓ indicate that higher/lower values correspond to better performance. The best results are highlighted in red **bold** underline, while the second-best results are marked in blue **bold**. Dashed lines separate diffusion-based approaches from others.

Method	(a) CUHK-CR1			(b) CUHK-CR2		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
SpA-GAN [45]	20.999	0.5162	0.0830	19.680	0.3952	0.1201
AMGAN-CR [66]	20.867	0.4986	0.1075	20.172	0.4900	0.093
CVAE [12]	24.252	0.7252	0.1075	22.631	0.6302	0.0489
MemoryNet [70]	26.073	0.7741	0.0315	24.224	0.6838	0.0403
MSDA-CR [67]	25.435	0.7483	0.0374	23.755	0.6661	0.0433
DE-MemoryNet [58]	**26.183**	**0.7746**	**0.0290**	**24.348**	**0.6843**	**0.0369**
DE-MSDA [58]	25.739	0.7592	0.0321	23.968	0.6737	0.0372
Ours (EMRDM)	**27.281**	**0.8007**	**0.0218**	**24.594**	**0.6951**	**0.0301**

(c) SEN12MS-CR	PSNR↑	SSIM↑	MAE↓	SAM↓
McGAN [16]	25.14	0.744	0.048	15.676
SAR-Opt-cGAN [21]	25.59	0.764	0.043	15.494
SAR2OPT [4]	25.87	0.793	0.042	14.788
SpA GAN [45]	24.78	0.754	0.045	18.085
Simulation-Fusion GAN [17]	24.73	0.701	0.045	16.633
DSen2-CR [43]	27.76	0.874	0.031	9.472
GLF-CR [63]	28.64	0.885	0.028	8.981
UnCRtainTS L2 [15]	28.90	0.880	0.027	8.320
ACA-Net [26]	29.78	0.896	0.025	7.770
DiffCR [74]	**31.77**	**0.902**	**0.019**	**5.821**
Ours (EMRDM)	**32.14**	**0.924**	**0.018**	**5.267**

(d) Sen2_MTC_New	PSNR↑	SSIM↑	LPIPS↓
McGAN [16]	17.448	0.513	0.447
Pix2Pix [28]	16.985	0.455	0.535
AE [53]	15.100	0.441	0.602
STNet [7]	16.206	0.427	0.503
DSen2-CR [43]	16.827	0.534	0.446
STGAN [52]	18.152	0.587	0.513
CTGAN [24]	18.308	0.609	0.384
SEN12MS-CR-TS Net [14]	18.585	0.615	0.342
PMAA [73]	18.369	0.614	0.392
UnCRtainTS [15]	18.770	0.631	0.333
DDPM-CR [29]	18.742	0.614	0.329
DiffCR [74]	**19.150**	**0.671**	**0.291**
Ours (EMRDM)	**20.067**	**0.709**	**0.255**

4.2. Performance Comparison

All quantitative results are illustrated in Tab. 1 using the optimal configuration for each model for a fair comparison. EMRDM surpasses all previous methods across all datasets and metrics, demonstrating its superiority. On the SEN12MS-CR dataset containing multi-spectral optical and auxiliary SAR images, EMRDM achieves significant improvements over existing methods. This validates its ca-

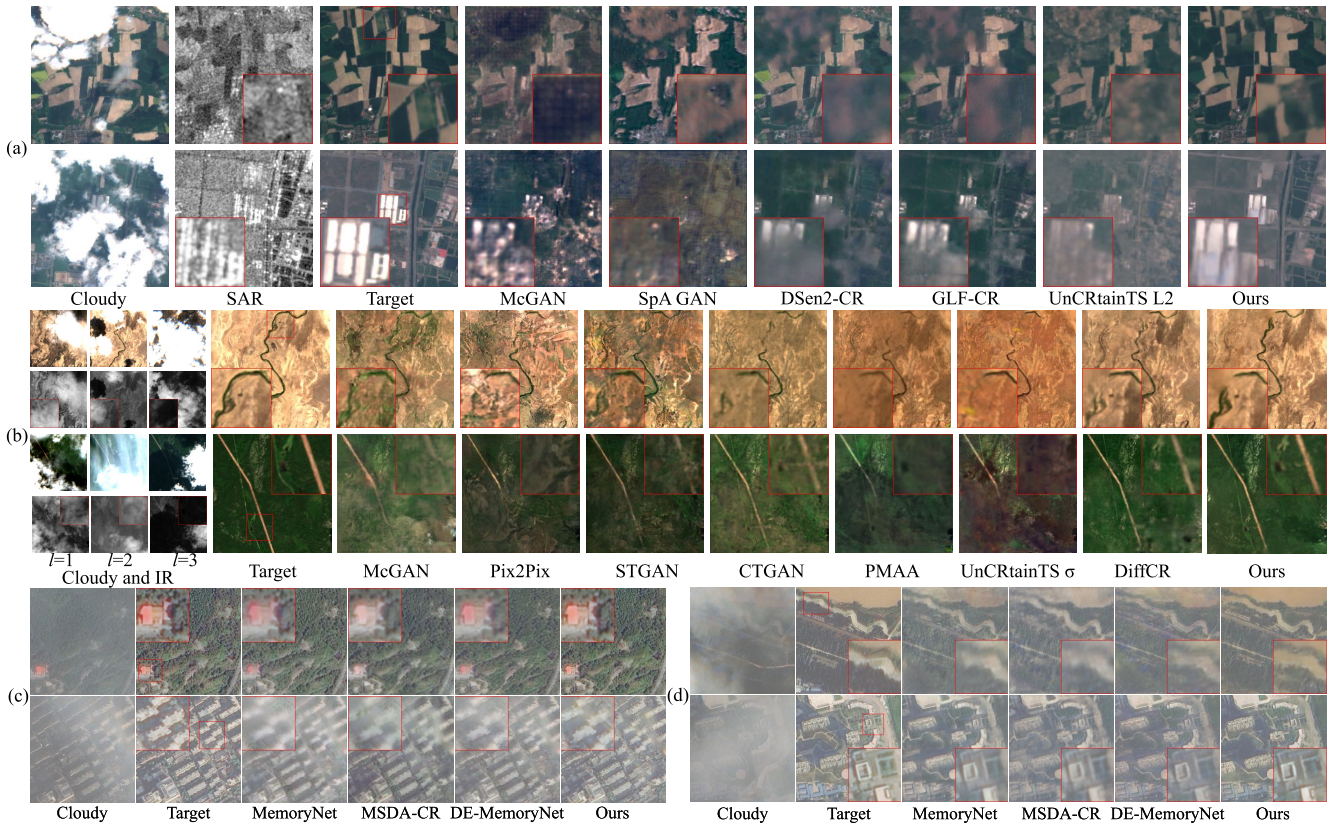


Figure 4. (a) SEN12MS-CR dataset results: RGB channels for optical imagery (linearly enhanced for visualization) and VV channel for SAR imagery. GLF-CR results are obtained by combining four separately processed subimages as it processes 128×128 images (256×256 for others). (b) Sen2_MTC_New dataset results. (c,d) RGB channel results on CUHK-CR1 and CUHK-CR2 datasets, respectively.

pability to exploit SAR's all-weather imaging characteristics and effectively process multi-spectral inputs. On the CUHK-CR1, CUHK-CR2, and Sen2_MTC_New datasets that mainly consist of RGB channels, EMRDM attains remarkable results across perceptual quality (LPIPS) and structural consistency metrics (SSIM, PSNR). Notably, it maintains performance superiority on the CUHK-CR1/CR2 datasets without auxiliary modalities, demonstrating robust CR capabilities with limited information. EMRDM further exhibits strong multi-temporal processing capability, as evidenced by leading metrics on the Sen2_MTC_New dataset. The visual results in Fig. 4 further prove the superior CR quality of EMRDM. In particular, when the input images are heavily cloud-covered, our model restores better textures, crucial for subsequent tasks after CR.

4.3. Ablation Study& Parameter Effect

Effects of Modules. We conduct ablation studies on key modules, as outlined in Tab. 2, using models trained for 500 epochs with a deterministic sampler, setting $N = 5$, $\sigma_{\text{data}} = 1.0$, $\sigma_{\text{min}} = 0.001$ and $\sigma_{\text{max}} = 100$ for a fair comparison. The baseline (config A) sets $s(t) = 1$, reducing our method to generative DMs, with only noise images as inputs. Config B and C incorporate cloudy and IR images, respectively. The results demonstrate their essential roles as conditioning inputs. Config D verifies the effectiveness of the EMRDM framework in Sec. 3.2 with $s(t) = \frac{1}{1+t}$ and $\sigma_{\text{mu}} = \sigma_{\text{cov}} = 0$. Incorporating preconditioning techniques proposed in Sec. 3.4 in config E, with $\sigma_{\text{mu}} = 1.0$, $\sigma_{\text{cov}} = 0.9$, results in improved performance.

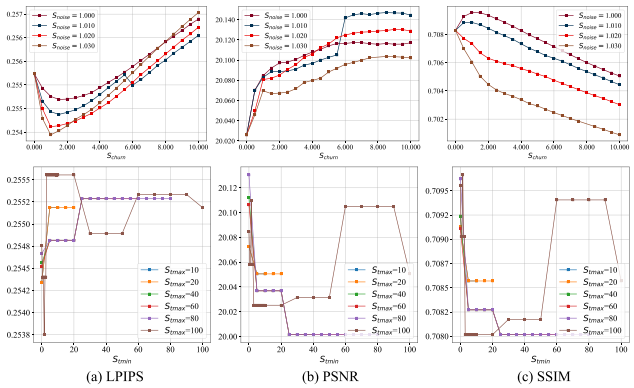
Table 2. We conducted an ablation study on the Sen2_MTC_New dataset to evaluate our method by incrementally adding modules.

Training configuration	PSNR↑	SSIM↑	MAE↓	SAM↓	LPIPS↓
A Baseline ($s(t) = 1$)	12.81	0.342	0.204	13.005	0.718
B + Corrupted images	18.26	0.649	0.109	6.526	0.311
C + IR images	19.31	0.677	0.095	6.547	0.279
D + Our MRDM framework	**19.52**	0.679	**0.092**	6.551	0.278
E + Our preconditioning	19.47	**0.693**	0.093	**6.390**	**0.267**

Effects of α, σ_{max} and N. Tab. 3 presents the results while varying key parameters. Each model is trained for 500 epochs, with $\sigma_{\text{data}} = \sigma_{\text{mu}} = 1.0$ and $\sigma_{\text{cov}} = 0.9$. We use a deterministic sampler with $\sigma_{\text{min}} = 0.001$. For α, which controls the ratio of $\boldsymbol{\mu}$ and $\boldsymbol{n}$ in the forward process, it yields the

Table 3. Hyperparameter analysis on the Sen2_MTC_New dataset.

Configurations			Metrics				
α	σ_{max}	N	PSNR↑	SSIM↑	MAE↓	SAM↓	LPIPS↓
0.2	100.0	5	19.34	0.692	0.095	6.306	0.269
0.5			19.14	0.675	0.097	6.580	0.283
0.8			19.90	0.689	0.088	6.249	0.260
1.0			19.44	0.688	0.091	6.367	0.273
2.0			19.77	0.704	0.087	5.922	0.262
3.0			**20.00**	**0.708**	**0.084**	**5.710**	**0.255**
4.0			19.76	0.695	0.087	5.821	0.263
3.0	40	5	19.58	0.701	0.087	5.764	0.260
	60		19.88	0.706	0.085	5.733	0.257
	80		19.96	0.707	**0.084**	5.726	0.256
	100		20.00	**0.708**	**0.084**	**5.710**	**0.255**
	150		**20.03**	0.707	0.085	5.730	0.256
	200		**20.03**	0.707	0.085	5.723	0.256
	300		20.02	0.705	0.086	5.728	0.257
3.0	100	4	19.98	0.702	0.085	5.744	0.259
		5	**20.00**	**0.708**	**0.084**	5.710	**0.255**
		6	19.97	0.705	0.084	5.710	0.257
		8	19.89	0.700	0.085	**5.695**	0.257
		10	19.89	0.700	0.085	**5.695**	0.257
		15	19.55	0.672	0.088	5.715	0.261
		50	19.19	0.641	0.091	5.857	0.270



(a) LPIPS (b) PSNR (c) SSIM

Figure 5. Analysis of our samplers on the Sen2_MTC_New dataset. When $S_{\text{churn}} = 0$, the sampler reduces to be deterministic. The upper row shows the effects of S_{churn} and S_{noise} by fixing $S_{\text{tmin}} = 0$ and $S_{\text{tmax}} \geq 100$. The lower row examines the effects of S_{tmin} and S_{tmax} with fixed $S_{\text{churn}} = 1$ and $S_{\text{noise}} = 1$. Note that $S_{\text{tmin}} >= S_{\text{tmax}}$ is excluded as this leads to a deterministic sampler.

optimal results across all metrics when set to 3. For σ_{max}, the results show that a moderate value (*e.g.*, 100) produces almost all the best metrics. For N, surprisingly, contrary to the expectations in generative DMs, a large N yields poor results, while using only five steps delivers superior results across most metrics. This finding aligns with [69].

Effect of Samplers. We examine our samplers in Fig. 5 using the $\alpha = 3.0$ configuration in Tab. 3, and setting $\sigma_{\text{min}} = 0.001$, $\sigma_{\text{max}} = 100$, and $N = 5$. According to the upper row of Fig. 5, the stochastic sampler consistently outperforms the deterministic one in PSNR, with $S_{\text{noise}} \in [1.000, 1.020]$ and $S_{\text{churn}} \geq 6.0$ achieving superior scores. However, high S_{churn} can negatively affect LPIPS and SSIM. While LPIPS is relatively insensitive to S_{noise}, SSIM declines at higher S_{noise}. We suggest using $S_{\text{noise}} \approx 1.000$ and $S_{\text{churn}} \approx 1.0$ for balanced metric performance. According to the lower row of Fig. 5, the optimal results are achieved across all metrics when $S_{\text{tmin}} \approx 0$. Generally, S_{tmax} should be relatively large, such as 80 and 100.

Table 4. Analysis of L on the Sen2_MTC_New dataset.

Sequence Length	PSNR↑	SSIM↑	MAE↓	SAM↓	LPIPS↓
$L = 1$	16.09	0.493	0.146	7.773	0.440
$L = 2$	18.10	0.623	0.106	7.313	0.344
$L = 3$	**20.07**	**0.709**	**0.084**	**5.670**	**0.255**

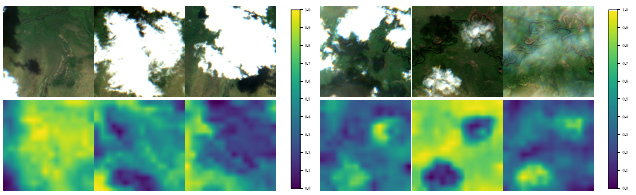


Figure 6. Visualizations of attention masks and their corresponding cloudy images from two cases on the Sen2_MTC_New dataset. Each mask at different time points is normalized to the range $[0, 1]$ and upsampled using bilinear interpolation to match the size of the cloudy images for clarity. The left panel shows a case from head 0, while the right panel displays a case from head 15.

Effect of the Network. We analyze the impact of L on our network (see Tab. 4), with models trained using the $\alpha = 3.0$ configuration in Tab. 3 and evaluated via a deterministic sampler ($\sigma_{\text{min}} = 0.001$, $\sigma_{\text{max}} = 100$, and $N = 5$). Increasing L consistently boosts performance across all metrics, highlighting the benefits of multi-temporal inputs and our network's ability to process them. Fig. 6 visualizes TFSA attention masks, with high attention scores for cloudless regions and low scores for cloudy ones. Regions occluded by clouds, characterized by low attention scores, correspondingly exhibit elevated scores in cloudless temporal counterparts. This validates TFSA's capacity to compensate for corrupted information by integrating information from spatially equivalent regions across the temporal dimension.

5. Conclusion

We propose a novel MRDM-based CR model named **EMRDM**. It offers a modular framework with updatable modules and an elucidated design space. With this advantage, we redesign core MRDM modules to boost CR performance, including restructuring the denoiser via a preconditioning technique and improving training and sampling processes. To achieve multi-temporal CR, a new network is devised to process sequential images in parallel. These improvements enable EMRDM to achieve superior results on mono-temporal and multi-temporal CR benchmarks.

6. Acknowledgments

This work was supported in part by National Natural Science Foundation of China (No. 62202336, No. 62172300, No. 62372326), and the Fundamental Research Funds for the Central Universities (No. 2024-4-YB-03).

References

[1] Arpit Bansal, Eitan Borgnia, Hong-Min Chu, Jie Li, Hamid Kazemi, Furong Huang, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Cold diffusion: Inverting arbitrary image transforms without noise. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[2] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22669–22679, 2023. 4

[3] David Bau, Jun-Yan Zhu, Jonas Wulff, William Peebles, Hendrik Strobelt, Bolei Zhou, and Antonio Torralba. Seeing what a gan cannot generate. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 1

[4] Jose D Bermudez, Patrick Nigri Happ, Dario Augusto Borges Oliveira, and Raul Queiroz Feitosa. Sar to optical image synthesis for cloud removal with generative adversarial networks. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4:5–11, 2018. 2, 6

[5] Christopher M Bishop. *Neural networks for pattern recognition*. Oxford university press, 1995. 4

[6] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6228–6237, 2018. 1

[7] Yang Chen, Qihao Weng, Luliang Tang, Xia Zhang, Muhammad Bilal, and Qingquan Li. Thick clouds removing from multitemporal landsat images using spatiotemporal neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2020. 6

[8] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14367–14376, 2021. 2

[9] Katherine Crowson, Stefan Andreas Baumann, Alex Birch, Tanishq Mathew Abraham, Daniel Z Kaplan, and Enrico Shippole. Scalable high-resolution pixel-space image synthesis with hourglass diffusion transformers. In *Forty-first International Conference on Machine Learning*, 2024. 2, 4

[10] Mauricio Delbracio and Peyman Milanfar. Inversion by direct iteration: An alternative to denoising diffusion for image restoration. *Transactions on Machine Learning Research*, 2023. Featured Certification. 2

[11] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 1, 2

[12] Haidong Ding, Yue Zi, and Fengying Xie. Uncertainty-based thin cloud removal network via conditional variational autoencoders. In *Proceedings of the Asian Conference on Computer Vision*, pages 469–485, 2022. 6

[13] Patrick Ebel, Andrea Meraner, Michael Schmitt, and Xiao Xiang Zhu. Multisensor data fusion for cloud removal in global and all-season sentinel-2 imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 59(7):5866–5878, 2020. 6

[14] Patrick Ebel, Yajin Xu, Michael Schmitt, and Xiao Xiang Zhu. Sen12ms-cr-ts: A remote-sensing data set for multimodal multitemporal cloud removal. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2022. 2, 6

[15] Patrick Ebel, Vivien Sainte Fare Garnot, Michael Schmitt, Jan Dirk Wegner, and Xiao Xiang Zhu. Uncrtaints: Uncertainty quantification for cloud removal in optical satellite time series. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2086–2096, 2023. 2, 6

[16] Kenji Enomoto, Ken Sakurada, Weimin Wang, Hiroshi Fukui, Masashi Matsuoka, Ryosuke Nakamura, and Nobuo Kawaguchi. Filmy cloud removal on satellite imagery with multispectral conditional generative adversarial nets. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 48–56, 2017. 2, 6

[17] Jianhao Gao, Qiangqiang Yuan, Jie Li, Hai Zhang, and Xin Su. Cloud removal with fusion of high resolution optical and sar images using generative adversarial networks. *Remote Sensing*, 12(1):191, 2020. 2, 6

[18] Vivien Sainte Fare Garnot and Loic Landrieu. Lightweight temporal self-attention for classifying satellite images time series. In *Advanced Analytics and Learning on Temporal Data: 5th ECML PKDD Workshop, AALTD 2020, Ghent, Belgium, September 18, 2020, Revised Selected Papers 6*, pages 171–181. Springer, 2020. 2, 4

[19] Vivien Sainte Fare Garnot and Loic Landrieu. Panoptic segmentation of satellite image time series with convolutional temporal attention networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4872–4881, 2021. 2, 4

[20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 1, 2

[21] Claas Grohnfeldt, Michael Schmitt, and Xiaoxiang Zhu. A conditional generative adversarial network to fuse sar and multispectral optical data for cloud removal from sentinel-2 images. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 1726–1729. IEEE, 2018. 2, 6

[22] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 4

[23] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 2

[24] Gi-Luen Huang and Pei-Yuan Wu. Ctgan: Cloud transformer generative adversarial network. In *2022 IEEE In-*

ternational Conference on Image Processing (ICIP), pages 511–515. IEEE, 2022. 2, 6

[25] Lei Huang, Jie Qin, Yi Zhou, Fan Zhu, Li Liu, and Ling Shao. Normalization techniques in training dnns: Methodology, analysis and application. *IEEE transactions on pattern analysis and machine intelligence*, 45(8):10173–10196, 2023. 4

[26] Wenli Huang, Ye Deng, Yang Wu, and Jinjun Wang. Attentive contextual attention for cloud removal. *IEEE Transactions on Geoscience and Remote Sensing*, 2024. 6

[27] Aapo Hyvärinen and Peter Dayan. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4), 2005. 3, 4

[28] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 6

[29] Ran Jing, Fuzhou Duan, Fengxian Lu, Miao Zhang, and Wenji Zhao. Denoising diffusion probabilistic feature-based network for cloud removal in sentinel-2 imagery. *Remote Sensing*, 15(9):2217, 2023. 1, 6

[30] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in Neural Information Processing Systems*, 33:12104–12114, 2020. 4

[31] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022. 2, 3, 4, 5, 6

[32] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. *Advances in Neural Information Processing Systems*, 35:23593–23606, 2022. 2

[33] Michael D King, Steven Platnick, W Paul Menzel, Steven A Ackerman, and Paul A Hubanks. Spatial and temporal distribution of clouds observed by modis onboard the terra and aqua satellites. *IEEE transactions on geoscience and remote sensing*, 51(7):3826–3852, 2013. 1

[34] Nataliia Kussul, Mykola Lavreniuk, Sergii Skakun, and Andrii Shelestov. Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geoscience and Remote Sensing Letters*, 14(5):778–782, 2017. 1

[35] Congyu Li, Xinxin Liu, and Shutao Li. Transformer meets gan: Cloud-free multispectral image reconstruction via multi-sensor data fusion in satellite images. *IEEE Transactions on Geoscience and Remote Sensing*, 2023. 2

[36] Xin Li, Yulin Ren, Xin Jin, Cuiling Lan, Xingrui Wang, Wenjun Zeng, Xinchao Wang, and Zhibo Chen. Diffusion models for image restoration and enhancement–a comprehensive survey. *arXiv preprint arXiv:2308.09388*, 2023. 1, 2

[37] Chao-Hung Lin, Po-Hung Tsai, Kang-Hua Lai, and Jyun-Yuan Chen. Cloud removal from multitemporal satellite images using information cloning. *IEEE transactions on geoscience and remote sensing*, 51(1):232–241, 2012. 2

[38] Guan-Horng Liu, Arash Vahdat, De-An Huang, Evangelos A Theodorou, Weili Nie, and Anima Anandkumar. I2sb: image-to-image schrödinger bridge. In *Proceedings of the 40th International Conference on Machine Learning*, pages 22042–22062, 2023. 2

[39] Jiawei Liu, Qiang Wang, Huijie Fan, Yinong Wang, Yandong Tang, and Liangqiong Qu. Residual denoising diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2773–2783, 2024. 2

[40] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11461–11471, 2022. 2

[41] Ziwei Luo, Fredrik K. Gustafsson, Zheng Zhao, Jens Sjölund, and Thomas B. Schön. Image restoration with mean-reverting stochastic differential equations. In *Proceedings of the 40th International Conference on Machine Learning*, pages 23045–23066. PMLR, 2023. 1, 2, 3, 4

[42] Ziwei Luo, Fredrik K Gustafsson, Zheng Zhao, Jens Sjölund, and Thomas B Schön. Refusion: Enabling large-size realistic image restoration with latent-space diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1680–1691, 2023. 2

[43] Andrea Meraner, Patrick Ebel, Xiao Xiang Zhu, and Michael Schmitt. Cloud removal in sentinel-2 imagery using a deep residual neural network and sar-optical data fusion. *ISPRS Journal of Photogrammetry and Remote Sensing*, 166:333–346, 2020. 2, 6

[44] Mehdi Mirza. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 2

[45] Heng Pan. Cloud removal for remote sensing imagery via spatial attention generative adversarial network. *arXiv preprint arXiv:2009.13015*, 2020. 2, 6

[46] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 2, 4

[47] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2

[48] Marc Rußwurm and Marco Korner. Temporal vegetation modelling using long short-term memory networks for crop identification from medium-resolution multi-spectral satellite images. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 11–19, 2017. 1

[49] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 conference proceedings*, pages 1–10, 2022. 2, 4

[50] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4713–4726, 2022. 2

[51] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016. 1

[52] Vishnu Sarukkai, Anirudh Jain, Burak Uzkent, and Stefano Ermon. Cloud removal from satellite images using spatiotemporal generator networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1796–1805, 2020. 2, 6

[53] Wassana Sintarasirikulchai, Teerasit Kasetkasem, Tsuyoshi Isshiki, Thitiporn Chanwimaluang, and Preesan Rakwatin. A multi-temporal convolutional autoencoder neural network for cloud removal in remote sensing images. In *2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pages 360–363. IEEE, 2018. 6

[54] Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023. 2

[55] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 1, 2

[56] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. 1, 2, 3

[57] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning*, pages 32211–32252. PMLR, 2023. 2

[58] Jialu Sui, Yiyang Ma, Wenhan Yang, Xiaokang Zhang, Man-On Pun, and Jiaying Liu. Diffusion enhancement for cloud removal in ultra-resolution remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 2024. 1, 6

[59] Maria Vakalopoulou, Konstantinos Karantzalos, Nikos Komodakis, and Nikos Paragios. Building detection in very high resolution multispectral data with deep learning features. In *2015 IEEE international geoscience and remote sensing symposium (IGARSS)*, pages 1873–1876. IEEE, 2015. 1

[60] Yinhuai Wang, Jiwen Yu, and Jian Zhang. Zero-shot image restoration using denoising diffusion null-space model. In *The Eleventh International Conference on Learning Representations*, 2023. 2

[61] Bin Xia, Yulun Zhang, Shiyin Wang, Yitong Wang, Xinglong Wu, Yapeng Tian, Wenming Yang, and Luc Van Gool. Diffir: Efficient diffusion model for image restoration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13095–13105, 2023. 2

[62] Quan Xiong, Guoqing Li, Xiaochuang Yao, and Xiaodong Zhang. Sar-to-optical image translation and cloud removal based on conditional generative adversarial networks: Literature survey, taxonomy, evaluation indicators, limits and future directions. *Remote Sensing*, 15(4):1137, 2023. 1

[63] Fang Xu, Yilei Shi, Patrick Ebel, Lei Yu, Gui-Song Xia, Wen Yang, and Xiao Xiang Zhu. Glf-cr: Sar-enhanced cloud removal with global–local fusion. *ISPRS Journal of Photogrammetry and Remote Sensing*, 192:268–278, 2022. 2, 6

[64] Meng Xu, Mark Pickering, Antonio J Plaza, and Xiuping Jia. Thin cloud removal based on signal transmission principles and spectral mixture analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 54(3):1659–1669, 2015. 2

[65] Meng Xu, Xiuping Jia, Mark Pickering, and Sen Jia. Thin cloud removal from optical remote sensing images using the noise-adjusted principal components transform. *ISPRS Journal of Photogrammetry and Remote Sensing*, 149:215–225, 2019. 2

[66] Meng Xu, Furong Deng, Sen Jia, Xiuping Jia, and Antonio J Plaza. Attention mechanism-based generative adversarial networks for cloud removal in landsat images. *Remote sensing of environment*, 271:112902, 2022. 6

[67] Weikang Yu, Xiaokang Zhang, and Man-On Pun. Cloud removal in optical remote sensing imagery using multiscale distortion-aware networks. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2022. 6

[68] Qiangqiang Yuan, Huanfeng Shen, Tongwen Li, Zhiwei Li, Shuwen Li, Yun Jiang, Hongzhang Xu, Weiwei Tan, Qianqian Yang, Jiwen Wang, et al. Deep learning in environmental remote sensing: Achievements and challenges. *Remote sensing of Environment*, 241:111716, 2020. 1

[69] Zongsheng Yue, Jianyi Wang, and Chen Change Loy. Resshift: Efficient diffusion model for image super-resolution by residual shifting. *Advances in Neural Information Processing Systems*, 36:13294–13307, 2023. 2, 8

[70] Xiao Feng Zhang, Chao Chen Gu, and Shan Ying Zhu. Memory augment is all you need for image restoration. *arXiv preprint arXiv:2309.01377*, 2023. 6

[71] Xiaohu Zhao and Kebin Jia. Cloud removal in remote sensing using sequential-based diffusion models. *Remote Sensing*, 15(11):2861, 2023. 1, 2

[72] Xiao Xiang Zhu, Devis Tuia, Lichao Mou, Gui-Song Xia, Liangpei Zhang, Feng Xu, and Friedrich Fraundorfer. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine*, 5(4):8–36, 2017. 1

[73] Xuechao Zou, Kai Li, Junliang Xing, Pin Tao, and Yachao Cui. Pmaa: A progressive multi-scale attention autoencoder model for high-performance cloud removal from multi-temporal satellite imagery. In *ECAI*, pages 3165–3172, 2023. 6

[74] Xuechao Zou, Kai Li, Junliang Xing, Yu Zhang, Shiying Wang, Lei Jin, and Pin Tao. Diffcr: A fast conditional diffusion framework for cloud removal from optical satellite images. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–14, 2024. 1, 2, 6