Inpainting Radar Missing Data via Denoising Iterative Restoration

Wei Zhang, *Member, IEEE*, Xinyu Zhang, Zhuyu Jin, Youqi Wen, and Jie Liu

***Abstract*—The issue of missing radar data has long presented a challenge for meteorologists. Meteorological radars often suffer from data gaps and poor quality due to various factors, such as random partial beam blockage (PBB), instrument malfunctions, or irregular blind spots in detection locations. Therefore, it is essential that low-quality weather radar observations be remedied. Traditional deep learning methods are not well suited to this task, as they are incompatible with pixel-level remote sensing data and struggle to identify discrete features, leading to inaccurate repairs in areas of high reflectivity. In light of the aforementioned challenges, in this article, we propose a method for radar data restoration using a noise-guided diffusion probability model driven by known information. The aim of this approach is to effectively address missing and poor-quality radar data in various scenarios. We performed experiments in the central region of the United States using seamlessly blended scanning reflectivity radar data provided by the National Oceanic and Atmospheric Administration. We simulated two data gap types: random PBB and short radar outage periods. We tested our method on data from different years and compared its performance to those of other deep learning models under the same missing data conditions. The experimental results indicate that, compared with other methods, our method improves the inpainting accuracy index by approximately 30% and the inpainting quality index by approximately 10% on the missing inpainting of outage. Moreover, it shows excellent inpainting effects on beam blockage, highlighting its considerable potential to improve quantitative applications and various tasks under inclement weather conditions.**

***Index Terms*—Diffusion model, high echo, inpainting, radar reflectivity data.**

I. Introduction

THE absence and poor quality of data from the scanning area of meteorological radar are common issues that are influenced by various factors including weather conditions, such as heavy fog, thick clouds, strong winds, or interference from solar radiation during dawn and dusk, which can affect radar echo results and impact precipitation forecasting [1]. Radar malfunctioning or maintenance may also result in missing areas in the scanning region [2].

Topographical obstacles, such as high mountains, deep canyons, the topography of the scanned area, and ground clutter, caused by anomalous propagation of radar signals can result in regions with missing data [3], while echoes from wind farms can also interfere with radar returns [4].

The most common interferences are beam blockage or partial beam blockage (PBB) [5] and short outages [6]. This article's primary focus is the recovery of random and widespread beam blockage and short radar outage periods caused by physical obstacles and multipath effects. Researchers have proposed various traditional data correction and completion methods to address the low-quality or missing radar data caused by the factors mentioned above. Correction methods typically involve the use of digital elevation models [7] or polarimetric radar [8] to correct low-quality data, while completion methods often rely on low-level data features, such as texture and color, to infer missing pixel values, commonly using interpolation [9], [10], [11] or nearest-neighbor filling methods [12]. Correction and completion methods are generally simple and computationally fast, but due to the lack of high-level semantic information, their restoration results may not be sufficiently accurate.

Simultaneously, radar data may be approximated as a form of image data; image inpainting is an important research direction in computer vision, aimed at automatically restoring missing or low-quality regions in images [13]. As deep learning techniques continue to evolve, the emergence of convolutional neural network architectures, such as UNet [14], and self-attention mechanism-based neural network architectures, such as Transformer [15], has significantly advanced the computer vision field, and such architectures have been widely applied to image inpainting tasks. In 2016, Pathak et al. [16] proposed a deep neural network-based image inpainting algorithm that utilized a CNN-based autoencoder to learn low-level features and high-level semantics of images and achieved automatic completion by alternately training known and unknown regions. In 2017, Iizuka et al. [17] introduced a generative adversarial network (GAN)-based image inpainting algorithm, which employed a GAN framework with global and local discriminators to generate high-quality and diverse inpainted images. In 2021, Deng et al. [18] proposed a fully convolutional network with attention modules that better modeled the affinity between different regions in an image and improved the effectiveness and quality of image inpainting. In the field of radar-data inpainting, researchers have

Manuscript received 14 January 2024; revised 19 April 2024; accepted 7 May 2024. Date of publication 13 May 2024; date of current version 14 June 2024. This work was supported by the National Natural Science Foundation of China under Grant 42276202. *(Wei Zhang, Xinyu Zhang, and Zhuyu Jin contributed equally to this work.) (Corresponding authors: Wei Zhang; Zhuyu Jin.)*

Wei Zhang, Xinyu Zhang, Youqi Wen, and Jie Liu are with the College of Information Science and Technology, Ocean University of China, Qingdao 266100, China (e-mail: weizhang@ouc.edu.cn).

Zhuyu Jin is with the Qingdao Joint Institute of Marine Meteorology, Chinese Academy of Meteorological Sciences, Qingdao 266034, China (e-mail: jinzhuyu@cma.gov.cn).

MRMS radar dataset can be downloaded at https://mtarchive.geol.iastate.edu/.

Digital Object Identifier 10.1109/JSTARS.2024.3400407

explored the use of convolutional neural networks, such as fully convolutional neural networks [19], U-Net [20], UNet++ [21], GANs [21], and conditional GANs [22], [23] for multisource data fusion for data recovery [24]. By utilizing deep learning methods, it is possible to train neural networks to learn higher level semantic information, generate high-quality inpainted results, and achieve performances that significantly surpass current correction methods in radar data inpainting tasks [21].

However, deep neural methods continue to face certain challenges when applied to radar data-inpainting tasks. Owing to the disparities between traditional image data and pixel-level remote sensing data, such as satellite and radar data, which possess complex spatial features and high spatial resolution [25], [26], with each feature often having complex structures and surface roughness at a small scale–these features can be measured using quality evaluation metrics, such as peak signal-to-noise ratio (PSNR) [27] and structural similarity index (SSIM) [27]. Deep learning models often struggle to learn discrete features, which can lead to over-smoothing of inpainted regions [28], loss of significant discrete information, and higher false-positive rates. Researchers have acknowledged this issue and the proposed solutions aimed at mitigating over-smoothing [29]. This work suggests employing a conditional denoising diffusion probability model (CDDPM) as a means of mitigating over-smoothing. Denoising diffusion probabilistic models (DDPM) [30] are a popular, recently developed deep learning approach in the field of image generation. This is a generative model that exhibits outstanding performance in the field of image inpainting [31], [32] They employ deep neural networks to model the joint distribution of images with the aim of achieving denoising and image generation [33]. Inspired by DDPM [30] and noise conditional score networks [34], [35], we propose a novel approach to radar data inpainting based on conditional guidance. The basic idea is to perform a stepwise iterative restoration process, in which a specific conditional guides the transformation of a standard normal distribution into the desired empirical data distribution, similar to Langevin dynamics [35], [36]. We have made simple yet effective modifications to the original denoising target UNet by incorporating additional conditional information for learning. This allows the DDPM to predict noise through the UNet network and achieve data generation for missing regions through inverse denoising iterations. Our approach outperforms the aforementioned deep-learning methods in radar data inpainting, thus yielding more realistic and effective inpainted results.

Our main contributions are the proposal of a novel data-repair method (CDDPM) and the addition of pixel-level modification of the loss function. The CDDPM method uses known information as a condition to guide the generative model to repair areas of missing data, and improves the attention mechanism of the original model, greatly reducing the amount of calculations required. Compared with the existing deep-learning-based radar-data recovery methods, the proposed model produces more realistic and accurate results. The pixel-level modification of the loss function improves the accuracy of pixel-wise inpainting results.

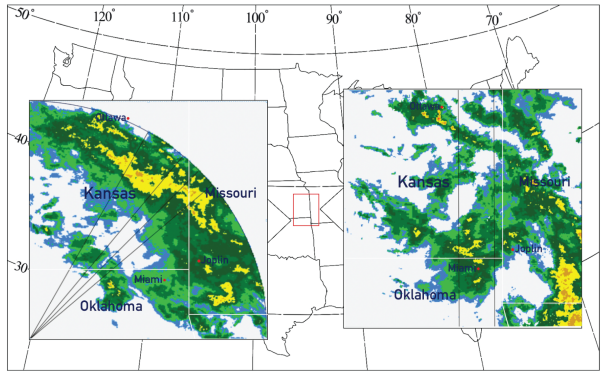

Fig. 1. Geographical location of the study area, which has a Bsk-Semiarid Steppe climate.

II. Methodology

A. Data

The analyses reported herein used the Seamless Hybrid Scan Reflectivity radar data provided by the multiradar/multisensor system (MRMS) [37], a product of the National Severe Storms Laboratory under the National Oceanic and Atmospheric Administration [38] in the United States. The data coverage includes selected regions in the southern United States and Canada, with a total area of 7000 km × 3500 km. As Fig. 1 illustrates, this article focused on meteorological features and missing regions in the central United States to evaluate the effectiveness with which it inpainted the missing regions. The region is adjacent to multiple climatic zones and experiences frequent occurrences of severe convective weather, which makes it conducive to capturing the characteristics of intense convective phenomena. The dataset's spatial resolution was 1 km, while its temporal resolution was set at 2 min.

We selected the MRMS data from March to August 2017–2021. We cropped a region of 256 × 256 in the central United States to simulate the effective scanning range of an individual radar [37], simulating scenarios with random PBB and short radar outage periods. The region is adjacent to several climatic zones and can thus take into account a range of climatic features. As Fig. 1 illustrates, to ensure a higher-quality dataset, we selected regions in which the reflectivity was greater than 10 dBZ [39] and covered 30 pixels or more with reflectivity greater than 35 dBZ [40]. This dataset includes a diverse range of high- and low-reflectivity information and was normalized to the range (0,1). To eliminate the effect of meaningless maxima and minima, we set radar reflectivity values >80 to 80, and reflectivity values <0 to 0. Then, we uniformly divided the values by 80 to normalize the data to (0, 1). We used data from 2017–2020 for model training and data from 2021 as the test dataset.

B. Method

1) Introduction: We considered the occluded data and the original data as forming dataset $\mathcal{D} = \{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^{N}$ composed of data pairs, where $\boldsymbol{x}_i$ represents a sample obtained by sampling from an unknown conditional data distribution $p(\boldsymbol{y} \mid \boldsymbol{x})$ where

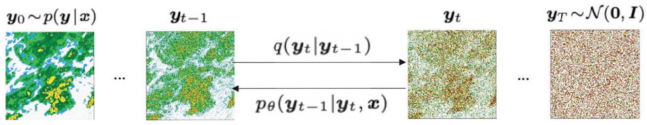

Fig. 2. Process of adding random Gaussian noise and removing Gaussian noise to the radar data region.

$\boldsymbol{x}$ is a set of $\boldsymbol{x}_i$. We mapped the occluded image $\boldsymbol{x}$ to the target inpainted image $\boldsymbol{y}$ through an inverse iterative process and approximate learning to fit the conditional data distribution $p(\boldsymbol{y} \mid \boldsymbol{x})$. To achieve this goal, we employed a DDPM [34] to generate the missing regions of the image by incorporating conditional guidance.

As Fig. 2 illustrates, the data-restoration process includes two stages: forward perturbation and inverse denoising. The forward-perturbation process, considered the training phase, treats the forward diffusion as a Markov chain process denoted as $q(\boldsymbol{y}_t \mid \boldsymbol{y}_{t-1})$. Starting from the original data y_0, Gaussian noise is gradually added in each step until it transforms into isotropic pure Gaussian noise. The model takes the masked data $\boldsymbol{x}$ and the result y_t from the previous denoising iteration as inputs and is trained to estimate the added noise at each step. The inverse process is the inference stage, during which the model is used to approximate the inverse of the forward diffusion process. The CDDPM generates the target data $\boldsymbol{y}$ through an inverse denoising iteration process with a time length of T. Starting from the original pure Gaussian noise data $\boldsymbol{y}_T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, data restoration is performed through continuous denoising iterations to generate missing regions. The goal is to obtain the target data y0 that fit the conditioned data distribution $\boldsymbol{y}_0 \sim p(\boldsymbol{y} \mid \boldsymbol{x})$.

In general, as Fig. 3 indicates, during the diffusion process, Gaussian noise is incrementally added to the original data, ultimately transforming it into isotropic Gaussian noise. In the model training phase, training is performed for each time step. At any time step t during the training process, missing data and information obtained by adding noise at each time step are concatenated. The current time step t's time label is incorporated as input to the model. The model was trained to learn the noise added at each time step during the diffusion process. In the inverse process, we utilized the trained model to obtain the noise for removal at each time step. This iterative denoising process is performed step by step, ultimately producing the desired restoration results. We shall provide a detailed derivation and specific algorithms for both the diffusion and inverse processes.

2) Gaussian Diffusion Process: Based on the Gaussian diffusion process of DDPM [34], [30] in the forward process, we first define a forward Markov diffusion process q. In each time step T of this process, we gradually add random Gaussian noise of the corresponding scale to the original data image y_0

$$q\left(\boldsymbol{y}_{1:T} \mid \boldsymbol{y}_0\right) = \prod_{t=1}^{T} q\left(\boldsymbol{y}_t \mid \boldsymbol{y}_{t-1}\right) \tag{1}$$

$$q\left(\boldsymbol{y}_t \mid \boldsymbol{y}_{t-1}\right) = \mathcal{N}\left(\boldsymbol{y}_t; \sqrt{1-\beta_t}\boldsymbol{y}_{t-1}, \beta_t \mathbf{I}\right). \tag{2}$$

The standard deviation of the Gaussian noise distribution is determined using a predefined β_t. The mean is determined based on a fixed value β_t and the current time step t of the data $\mathbf{y}_t$. It is important to note that as t varies, the final data distribution tends to converge to an isotropic Gaussian distribution. In addition, at any time step t, $\mathbf{y}_t$ can also be obtained through parameter renormalization, calculated directly from $\boldsymbol{y}_0$ and β_t, as shown in (3), where $\alpha_t = 1 - \beta_t, \bar{\alpha}_t = \prod_{i=1}^{T} \alpha_i$

$$q\left(\boldsymbol{y}_t \mid \boldsymbol{y}_0\right) = \mathcal{N}\left(\boldsymbol{y}_t; \sqrt{\bar{\alpha}_t}\boldsymbol{y}_0, \left(1-\bar{\alpha}_t\right)\mathbf{I}\right). \tag{3}$$

Subsequently, in the computation of the posterior distribution during the inverse diffusion process, given (y_0, y_t), through certain mathematical operations, it is possible to derive the posterior distribution of $\boldsymbol{y}_{t-1}$ along with its mean and variance, as indicated in (4), where $\bar{\alpha}_t = \prod_{i=1}^{T} \alpha_i$

$$\begin{aligned} q\left(\boldsymbol{y}_{t-1} \mid \boldsymbol{y}_0, \boldsymbol{y}_t\right) &= \mathcal{N}\left(\boldsymbol{y}_{t-1} \mid \boldsymbol{\mu}, \sigma^2 \boldsymbol{I}\right) \\ \boldsymbol{\mu} &= \frac{\sqrt{\bar{\alpha}_{t-1}}\left(1-\alpha_t\right)}{1-\bar{\alpha}_t}\boldsymbol{y}_0 + \frac{\sqrt{\alpha_t}\left(1-\bar{\alpha}_{t-1}\right)}{1-\bar{\alpha}_t}\boldsymbol{y}_t \\ \sigma^2 &= \frac{\left(1-\bar{\alpha}_{t-1}\right)\left(1-\alpha_t\right)}{1-\bar{\alpha}_t}. \end{aligned} \tag{4}$$

This posterior distribution can help compute the variational lower bound of the log-likelihood of the target data distribution. Next, we describe how the data distribution of the inverse diffusion process may be fitted using a neural network.

3) Inverse Diffusion Process: The inverse diffusion process, as shown in Fig. 4, incorporates additional side information in the form of the original data, x. We optimized a denoising neural network model based on UNet, which takes the image with occluded regions and the currently perturbed image as inputs. At any given time y_t, the perturbed image can be directly obtained by adding noise to the original data y_0, as expressed in (5) [30]

$$\boldsymbol{y}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{y}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{z} \quad \boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}). \tag{5}$$

The objective is to recover noise-free target completion data y_0. The definition of noisy target data y_t differs from the marginal distribution of noise data in various steps of the forward diffusion process. The denoising model $\epsilon_\theta(x, y_t, t)$ takes the original data x and the target data with added noise y_t as inputs. Through training, it predicts noise. By adjusting the scalar noise scale β_t, the model can perceive the level of noise. The objective function for training the model is given by (6), aiming to minimize the expected error between $\epsilon_\theta(x, y_t, t)$ and the added noise–that is, minimizing the expectation of d given x, y, α_t, squared l_2 norm with the target ϵ, allows the model to learn the estimation of noise

$$\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})}\mathbb{E}_{\epsilon,\bar{\alpha}_t}\left\|\boldsymbol{\epsilon}_\theta(\boldsymbol{x}, \underbrace{\sqrt{\bar{\alpha}_t}\boldsymbol{y}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{z}}_{\boldsymbol{y}_t}, t) - \boldsymbol{\epsilon}\right\|_2^2. \tag{6}$$

The model inference process refers to the inverse diffusion process, which is represented by (7). The denoising iteration process is defined as the conditional probability distribution $p_\theta(\boldsymbol{y}_{0:T} \mid \boldsymbol{x})$, achieved by iteratively denoising information at each time step t to obtain the desired probability distribution.

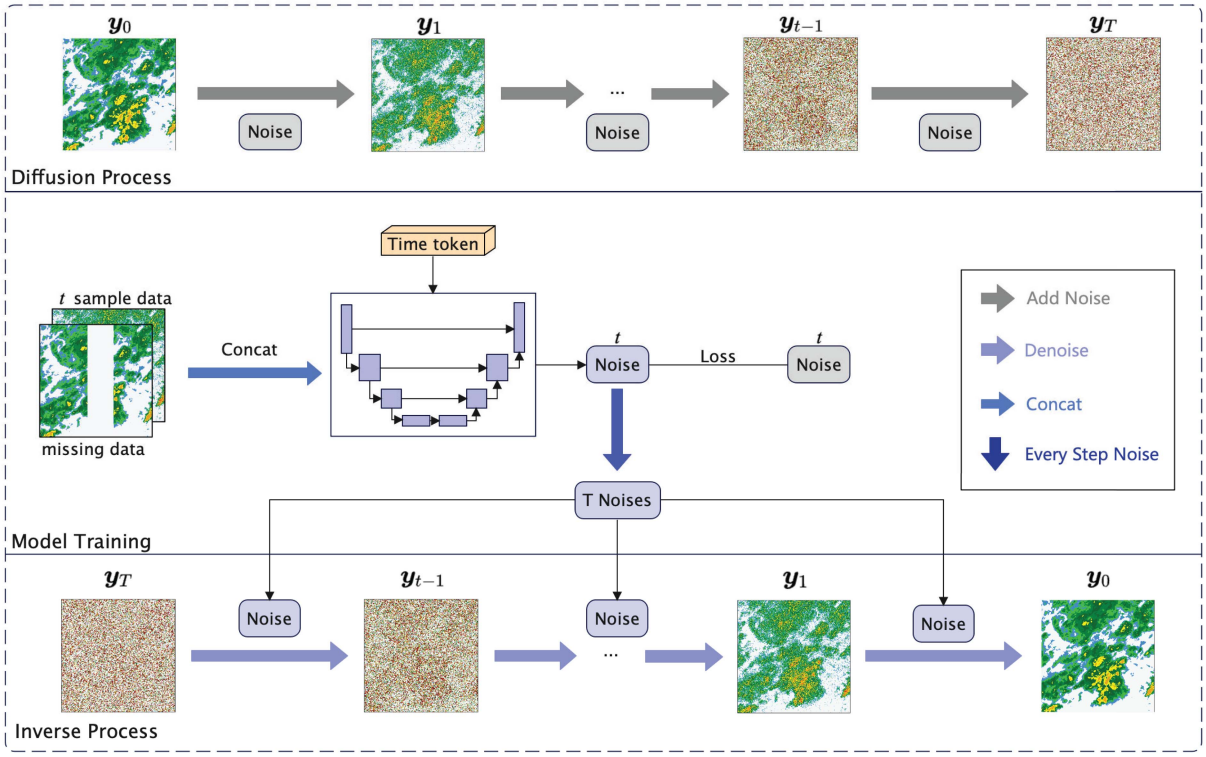

Fig. 3. Framework of the proposed inpainting model.

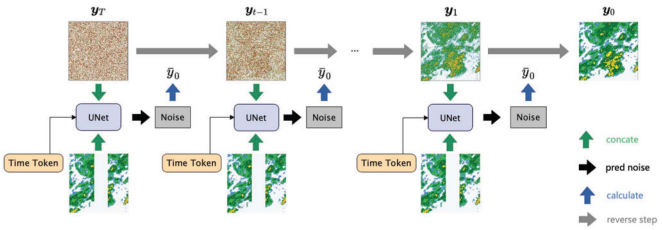

Fig. 4. Inverse diffusion process is progressively iterated to repair the data.

The denoising process at each time step is formulated in (9), where $p_\theta(\boldsymbol{y}_{t\text{-}1} \mid \boldsymbol{y}_t, \boldsymbol{x})$ is determined by the noise information $\mu_\theta(\boldsymbol{x}, \boldsymbol{y}_t)$ learned by the model, resulting in $y_{t\text{-}1}$, as per (12). This process is defined as an inverse Markov chain process, as shown in (8). It starts from a random Gaussian noise y_t; predicts y_0 through the noise in the model estimates; iteratively predicts $y_{t\text{-}1}$ in inverse, gradually removing noise; and ultimately generates the final result, satisfying the expected target conditional distribution

$$p_\theta\left(\boldsymbol{y}_{0:T} \mid \boldsymbol{x}\right) = p\left(\boldsymbol{y}_T\right) \prod_{t=1}^{T} p_\theta\left(\boldsymbol{y}_{t-1} \mid \boldsymbol{y}_t, \boldsymbol{x}\right) \tag{7}$$

$$p\left(\boldsymbol{y}_T\right) = \mathcal{N}\left(\boldsymbol{y}_T \mid \mathbf{0}, \boldsymbol{I}\right) \tag{8}$$

$$p_\theta\left(\boldsymbol{y}_{t-1} \mid \boldsymbol{y}_t, \boldsymbol{x}\right) = \mathcal{N}\left(\boldsymbol{y}_{t-1} \mid \mu_\theta\left(\boldsymbol{x}, \boldsymbol{y}_t\right), \sigma_t^2 \boldsymbol{I}\right). \tag{9}$$

We define the inference process as a conditional probability distribution, $p_\theta(\boldsymbol{y}_{t-1} \mid \boldsymbol{y}_t, \boldsymbol{x})$ assuming an isotropic Gaussian conditional distribution. The mean of the probability distribution, $\mu_\theta(\boldsymbol{x}, \boldsymbol{y}_t)$, is learned by the model. If we set the noise variance in the forward process to be as small as possible, the inverse diffusion process $p_\theta(\boldsymbol{y}_{t-1} \mid \boldsymbol{y}_t, \boldsymbol{x})$ may be approximated as a Gaussian distribution. Therefore, we can interpret the entire forward–backward process as being governed by a random Gaussian conditional distribution. It is important to maintain a sufficiently large noise scale such that y_t approximates the prior Gaussian distribution, allowing us to start the conditional sampling from random Gaussian white noise. Because our denoising model is trained to estimate noise, we rewrite (5) as a form of predicting y_0

$$\boldsymbol{y}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\boldsymbol{y}_t - \sqrt{1-\bar{\alpha}_t}\mu_\theta\left(\boldsymbol{x}, \boldsymbol{y}_t\right)\right). \tag{10}$$

By substituting and computing the mean of the conditional probability distribution, we obtain

$$\boldsymbol{\mu} = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{y}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\mu_\theta\left(\boldsymbol{x}, \boldsymbol{y}_t\right)\right). \tag{11}$$

Therefore, by parameterizing, we can obtain the iteratively generated results at each time step in the posterior conditional probability distribution of the inverse diffusion process learned by the model such as in (12), where $\boldsymbol{z}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ represents a specific quantity or characteristic of the generated results

$$\boldsymbol{y}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{y}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\mu_\theta\left(\boldsymbol{x}, \boldsymbol{y}_t\right)\right) + \sqrt{1-\alpha_t}\boldsymbol{z}_t. \tag{12}$$

In the end, the model consists of two stages: training and inference. In the training stage, as shown in Algorithm 1, given that x, y_0 follows the distribution $p(\boldsymbol{x}, \boldsymbol{y}_0)$, the model is trained

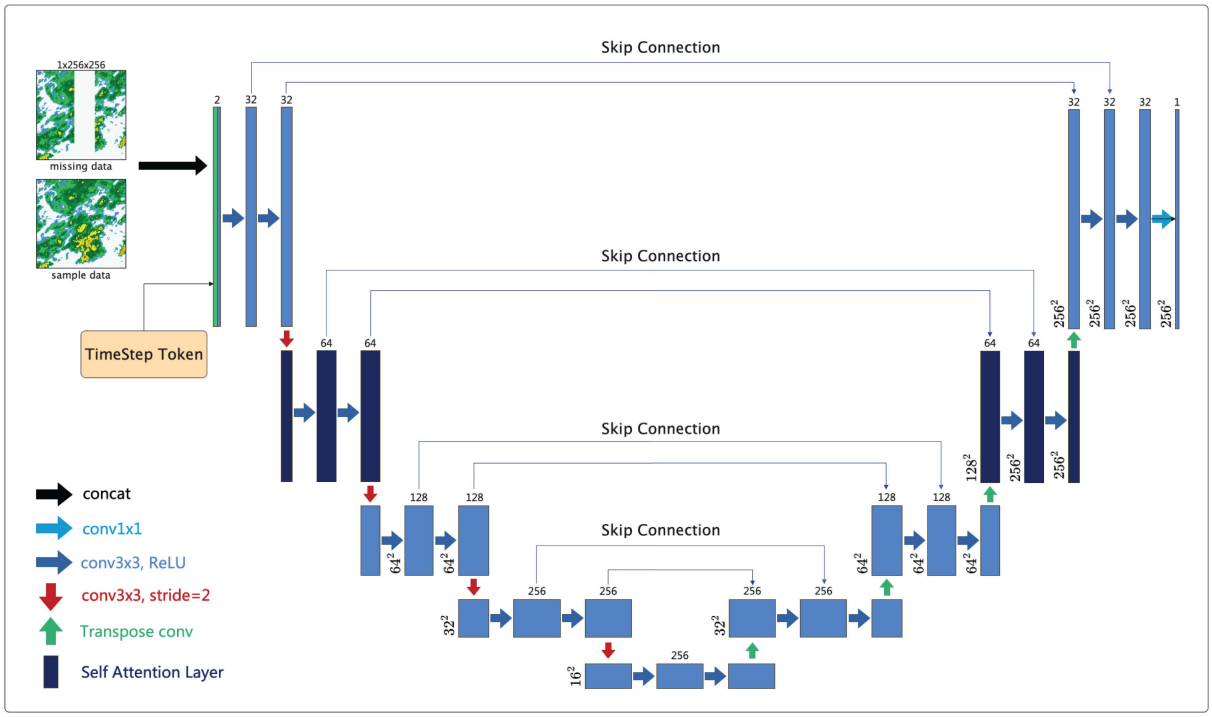

Fig. 5. Neural network architecture to learn the denoising process.

Algorithm 1: Training a Denoising Model ϵ_θ.

repeat
  $(\boldsymbol{x}, \boldsymbol{y}_0) \sim p(\boldsymbol{x}, \boldsymbol{y}_0)$
  $t \sim \text{Uniform}(\{1, \ldots, T\})$
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Take gradient descent step on
  $\nabla_\theta \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(x, y_t, t)\|^2$
  $+\nabla_\theta \|\boldsymbol{y}_0 - \frac{1}{\sqrt{\bar{\alpha}t}} \left(\boldsymbol{y}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(x, y_t, t)\right) \|^2$
until converged

Algorithm 2: Sample in T Iterative Inpaint Steps.

$\boldsymbol{y}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \ldots, 1$ **do**
  $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\boldsymbol{z} = \mathbf{0}$
  $\boldsymbol{y}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\boldsymbol{y}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(x, y_t, t)\right) + \sqrt{1-\alpha_t}\boldsymbol{z}$
end for
return $\boldsymbol{y}_0$

to learn the randomly added Gaussian noise from y_0 to y_t at any given time step t, initialized according to a uniform distribution. Through iterative learning, the model acquires knowledge of the randomly added Gaussian noise at all time steps T, allowing it to approximate the noise needed to estimate the inverse diffusion process.

In the sampling process of inverse diffusion, as shown in Algorithm 2, we randomly initialize a random Gaussian white noise $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ from the last time step. We use the noise predicted by our trained model to obtain the predicted y_0. Using y_0, along with the mean and standard deviation of the posterior conditional probability distribution, we predict the result of the previous time step, y_{t-1}. Through iterative steps, we ultimately generate a more diverse data distribution for the desired y_0.

4) Network Architecture: The neural network architecture designed to fit the denoising diffusion process is illustrated in Fig. 5. Taking a combination of real and occluded data and the time token of the current time step as input, the model consists of four layers. It effectively represents features by compressing the input into a latent space vector using multiple downsampling modules and employing 3×3 convolutions with a stride of 2. Subsequently, upsampling modules reconstruct the expected predicted noise data using deconvolution operations. Each residual module between the upsampling and downsampling modules features skip connections to facilitate information flow, enhancing the model's ability to capture both local and global features.

As depicted in Fig. 6, each sampling module includes two residual blocks, which are crucial for preserving important information during the encoding and decoding processes. Each residual block consists of two group normalization layers, a Swish activation function, dropout, and a 3×3 convolution operation, along with a residual connection. The residual connection enables the module's output to combine original and learned information, strengthening feature representation. Each set of residual modules contains a set of attention switches to decide whether to apply a self-attention operation in this layer. The model learns input information in different time steps, which are embedded in the model through a process that we term time projection.

This helps enhance global feature interactions, expanding the receptive field and ultimately bolstering the model's robustness. By leveraging self-attention mechanisms, the model gains the

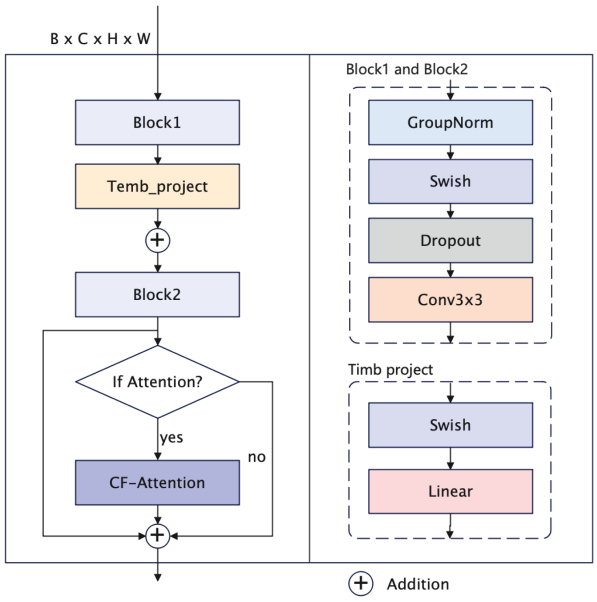

Fig. 6. Residual module.

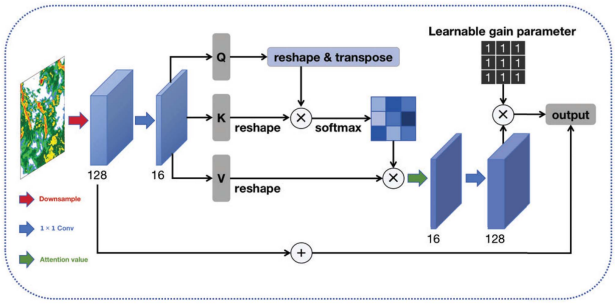

Fig. 7. CF-Attention: Self-attention mechanism based on channel fusion.

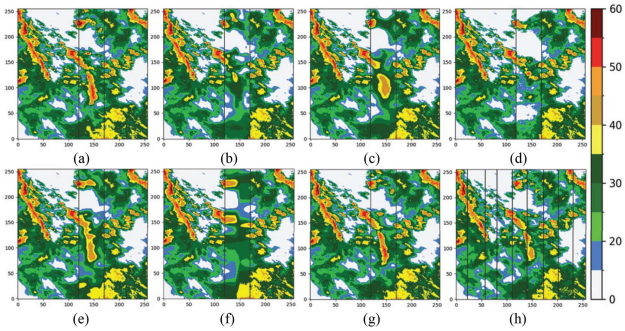

Fig. 8. Compare the effect of different methods to inpainting random short radar outage periods. (a) Observed data along with the restoration results of (b) UNet, (c) UNet++, (d) GAN, (e) ViT, (f) FFM, (g) CDDPM, and (h) extreme outage conditions.

ability to selectively focus on relevant features at different spatial positions, thereby promoting the interaction of contextual information.

To address the computational demands associated with the self-attention mechanism [41], a channel fusion-based approach is proposed in Fig. 7. In addition, to mitigate the significant computational overhead introduced by self-attention mechanisms, we propose the use of an adaptive channel fusion self-attention mechanism. For input features, we employed 1×1 convolution operations for channel fusion, followed by group normalization to reduce the feature dimensionality in the latent space during self-attention computations. After performing self-attention operations, the data dimensionality is restored through channel convolution. We initialized a set of learnable adaptive parameters to balance the results of self-attention. The computed features undergo a residual connection prior to output. Self-attention operations of this nature effectively reduce the computational burden traditionally associated with self-attention mechanisms. As given in Table I, we compared the computational and parameter costs of the model's CF-Attention module with traditional self-attention methods. For latent space variables with an input size of $64 \times 128 \times 128$, it can be observed that compared to traditional self-attention methods, the computational and parameter costs of the CF-attention module are significantly reduced. This substantial reduction contributes to significant decreases in computational expenses and training costs. This approach effectively alleviates the computational burden typically associated with conventional self-attention mechanisms. The restored data dimension is achieved through channel convolutions, ensuring the preservation of important information while mitigating computational complexity.

TABLE I
COMPARISON OF COMPUTATION AND PARAMETERS BETWEEN CF-ATTENTION AND ATTENTION

	FLOPs	Params
CF-Attention	33.554×10^6	2048
Attention	268.435×10^6	16 448

By incorporating these architectural components and techniques, the neural network aims to effectively learn and approximate the inverse diffusion process by predicting noise vectors [42], [43]. The trained model can then be utilized in the inference phase to generate samples that adhere to the desired target conditional distribution.

In the training preparation phase, for each iteration time step t, we chose a time step length of 500 steps, and adopted a linear increase strategy for the noise scale, with a maximum noise scale of 0.02. In terms of the choice of loss function, unlike conventional denoising loss functions, we introduced new constraint terms to form a comprehensive loss function. In addition to noise learning loss, we added loss terms for the real image and one-step prediction results. The specific loss function includes two parts: the noise loss and prediction loss terms. The noise loss term is used to measure the model's noise estimation capability, while the prediction loss term is used to evaluate the model's prediction accuracy for the real image and one-step prediction results. The predicted $\hat{y}_0$ was obtained using (10), and we minimized the L1 loss between the predicted and the ground truth $\hat{y}_0$. The specific formulas are as follows:

$$\mathcal{L}_{\text{model}} = \|z - \mu_\theta(\boldsymbol{x}, \boldsymbol{y}_t)\|^2 + \left\| y_0 - \frac{1}{\sqrt{\bar{\alpha}_t}} \left(\mathbf{y}_t - \sqrt{1-\bar{\alpha}_t} \mu_\theta(\boldsymbol{x}, \boldsymbol{y}_t) \right) \right\|^2. \quad (13)$$

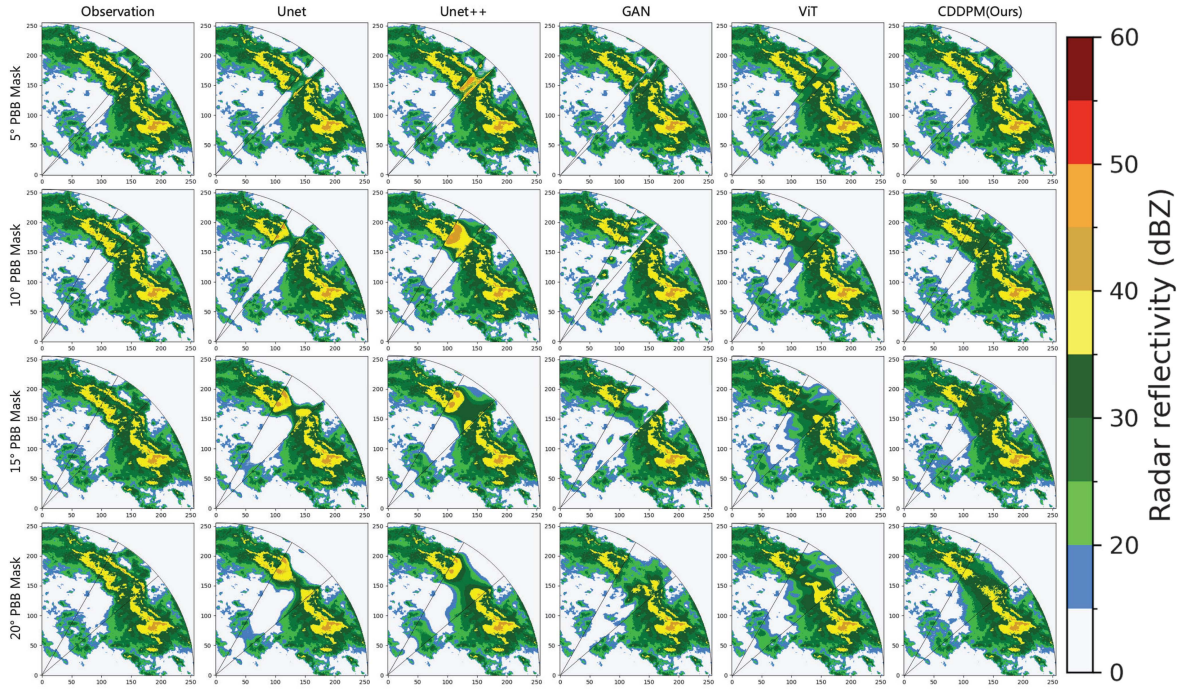

Fig. 9. Inpainting effects of different methods under PBB at four angles are compared.

During the training process, for each batch of data, we generated random time steps within the maximum time step range equal to the batch size. We directly trained the model using these time steps and corresponding data pairs. During the inference process, we experimented with a sampling approach using 50 spans of 10 time steps for DDIM [44] to generate the final completion results. The batch size was set to 4. We used the Adam optimizer with a fixed learning rate of 1e-4 for the completion model.

III. Results

A. Introduction to the Evaluation Indicators

To evaluate the model's overall data generation quality and assess its performance, we used several commonly used metrics in data completion tasks, including mean squared error (MSE), PSNR, and the SSIM. MSE is a metric used to measure regression models' performance. It effectively quantifies the pixel-level differences between the generated results and the ground truth data. The PSNR is based on the ratio of the peak signal intensity to the MSE. It measures the error sensitivity between corresponding labeled pixels, providing a quality evaluation of image sharpness and distortion levels. The specific formulas for these metrics are as follows:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{14}$$

$$\begin{aligned}\mathrm{PSNR} &= 10 \times \log_{10}\left(\frac{\mathrm{MAX}^2}{MSE}\right) \\ &= 20 \times \log_{10}\left(\frac{\mathrm{MAX}}{\sqrt{\mathrm{MSE}}}\right)\end{aligned} \tag{15}$$

where MAX represents the maximum possible pixel value in data, typically set at 255 for 8-bit images or 1 for normalized floating-point images. MSE quantifies the average squared difference between the pixel values of the original data and those of the reconstructed/compressed data.

In terms of measuring absolute errors, SSIM differs from MSE and PSNR. SSIM is a perceptual model that combines brightness, contrast, and structural information, taking into account the image's structural similarity. It better approximates the human perception of image quality. The specific formulas for these metrics are as follows:

$$\mathrm{SSIM} = \frac{(2\mu_p\mu_g + c_1)(2\delta_{pg} + c_2)}{(\mu_g^2 + \mu_p^2 + c_1)(\delta_p^2 + \delta_g^2 + c_2)} \tag{16}$$

where μ_p and μ_g represent the mean pixel values of the two images, respectively; δ_p and δ_g represent the pixel variances of the two images, respectively; δ_{pg} is the covariance of the pixels of the two images; and c_1 and c_2 are constants.

Due to the importance of high reflectivity radar data in meteorological research, particularly in the study of severe convection, and the sparsity of severe weather events, the distribution of direct echo data is highly imbalanced, particularly with a scarcity of high reflectivity data. Therefore, we take into account the completion performance of missing high reflectivity regions and utilize the critical success index (CSI), probability of detection (POD), and false alarm rate (FAR) [45] as evaluation metrics to assess completion effectiveness in high reflectivity regions. These metrics provide essential information regarding the consistency between the generated high reflectivity regions and the ground truth, detection accuracy, and the rate of false alarms.

The specific formulas for these metrics are as follows:

$$\mathrm{CSI} = \frac{\mathrm{TP}}{\mathrm{TP+FN+FP}} \tag{17}$$

$$\mathrm{POD} = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \tag{18}$$

$$\mathrm{FAR} = \frac{\mathrm{FP}}{\mathrm{FP+TP}} \tag{19}$$

where TP is true positive (prediction = 1 and truth = 1), FN is false negative (prediction = 0 and truth = 1), FP is false positive (prediction = 1 and truth = 0), and TN is true negative (prediction = 0 and truth = 0).

Subsequently, we conducted a comparative analysis of restoration performance metrics for various deep-learning methods, including U-Net [14], U-Net++ [46], GAN [47], and ViT [48], and traditional image-inpainting methods, such as the fast-marching method (FMM) [49] under two scenarios: PBB and short radar outage periods.

B. Comparative Experiments With Different Methods in Outage Case

We chose to assess performance under short radar outage periods by randomly obstructing specific regions in the test dataset. We randomly selected 95% of the data as training and the rest as testing. We also separately evaluated our method's restoration effectiveness in extreme blockage scenarios, utilizing a radar reflectivity threshold greater than 35 dBZ [50] to evaluate the performance metrics of CSI, POD, and FAR. This decision was motivated by the significant impact of extreme weather events, such as thunderstorms, tornadoes, and heavy precipitation, on various meteorological research fields. A radar reflectivity exceeding 35 dBZ is considered a crucial indicator of severe convective weather and plays a vital role in meteorological forecasting [50].

During the training process, we randomly selected initial column numbers within the range (100, 150) in a 256×256 data area, and then randomly added a number between 30 and 60 to the initial column number to determine the terminating obscured column number. Then, we obscured the region between the initial and terminating column numbers. We applied the same obscuring method during the testing phase. The input consisted of obscured information, and the output comprised the restored information

Existing radar data restoration methods, including UNet [20], UNet++ [21], and GAN [22], and so forth, continue to be beset by various challenges. Fig. 8 illustrates the (a) observed data along with the restoration results of (b) UNet, (c) UNet++, (d) GAN, (e) ViT, (f) FFM, and (g) CDDPM under outage occlusion while also showcasing the restoration performance of CDDPM under (h) extreme outage conditions.

The FFM method only partially extended the image according to edge information, resulting in unrealistic restoration results. By contrast, deep-learning-based methods effectively overcame this problem and produced more realistic repairs.

It should be noted that while UNet and ViT effectively recover high-intensity echo information, they exhibit over-smoothing features. These models tend to overlook discrete characteristics in the edge regions, resulting in a completion that appears less realistic. Meanwhile, GAN networks improve the realism of completed features in the occluded area but are notably inaccurate. Furthermore, due to the adversarial nature of GANs, their training tends to exhibit some degree of instability, which results in high training costs and makes it difficult for them to converge to an ideal state [51].

TABLE II
EVALUATION METRICS FOR THE RESTORATION PERFORMANCE UNDER OUTAGE CONDITIONS, ESPECIALLY THE HIGH ECHO REGION (>35 DBZ)

Models	MSE↓	PSNR↑	SSIM↑	CSI↑	POD↑	FAR↓
FMM	87.208	19.250	0.3464	0.210	0.383	0.599
UNet	32.273	23.993	0.5680	0.325	0.393	0.273
UNet++	33.043	23.681	0.5376	0.347	0.462	0.387
GAN	96.279	19.018	0.3542	0.219	0.436	0.624
ViT	35.197	23.721	0.4802	0.269	0.315	0.267
CDDPM-MSE	20.598	25.811	0.6436	0.414	0.502	0.301
CDDPM	**18.037**	**26.449**	**0.7030**	**0.454**	**0.574**	**0.252**
CDDPM-E	49.780	25.252	0.6639	0.419	0.519	0.309

Compared to these deep learning models, our CDDPM leverages sampled data to generate restoration in a highly effective manner. Moreover, compared with GANs, our DDPM-based method is easier to train [52]. It excels at restoring discrete features and accurately recovering missing high-intensity echo regions, demonstrating outstanding authenticity and precision. Furthermore, in extreme large-scale random outage scenarios, our approach demonstrates outstanding radar data restoration performance.

As Table II tabulates, we compared different methods' restoration performance in missing regions. Our approach evidently outperforms other deep learning methods in terms of data reconstruction metrics. Furthermore, compared with the other methods, our proposed technique exhibits significantly higher accuracy in restoring high reflectivity regions. This method's performance metrics in terms of repairing extreme data gaps (CDDPM-E) are also outstanding. To minimize pixel-level loss in the restored results to the greatest extent possible, we have added an image-to-image L2 loss-constraint term to the loss function. We conducted additional ablation experiments by retraining the model without the constraint term, to demonstrate the effectiveness of this modification. As given in the Table II (CDDPM-MSE), we obtained lower pixel-level loss without the constraint term than with the constraint term, which confirmed the effectiveness of our loss-function modification.

C. Comparative Experiments With Different Methods in Random PBB

For PBB scenarios, we performed tests to evaluate the restoration performance and image reconstruction metrics of different models at varying angles (5°, 10°, 15°, and 20°) of PBB obstruction. Owing to the missing regions' irregular shapes as a result of beam blocking, it is challenging to conduct tests exclusively on restoration areas. Therefore, to compare the performance differences in correction under different echo intensities, we examined the CSI, POD, and FAR metrics for the repaired complete data at the 10 and 35 dBZ thresholds.

TABLE III
INPAINTING EFFECT OF DIFFERENT METHODS IN 5°, 10°, 15°, AND 20° PBB

Methods	PBB-5°			PBB-10°			PBB-15°			PBB-20°		
	MSE	PSNR	SSIM	MSE	PSNR	SSIM	MSE	PSNR	SSIM	MSE	PSNR	SSIM
Unet	4.373	32.219	0.951	10.243	28.916	0.921	15.805	26.949	0.903	20.933	25.619	0.881
Unetpp	10.535	28.216	0.943	11.162	28.271	0.924	17.824	26.326	0.896	23.611	25.059	0.867
GAN	12.199	27.526	0.940	19.138	25.510	0.905	31.867	23.313	0.870	44.647	21.768	0.837
ViT	3.659	33.178	**0.961**	7.645	30.055	**0.926**	11.704	28.011	0.898	15.527	26.744	0.869
CDDPM(Ours)	**2.140**	**35.875**	0.957	**5.977**	**31.405**	0.925	**10.881**	**28.649**	**0.907**	**15.307**	**26.905**	**0.883**

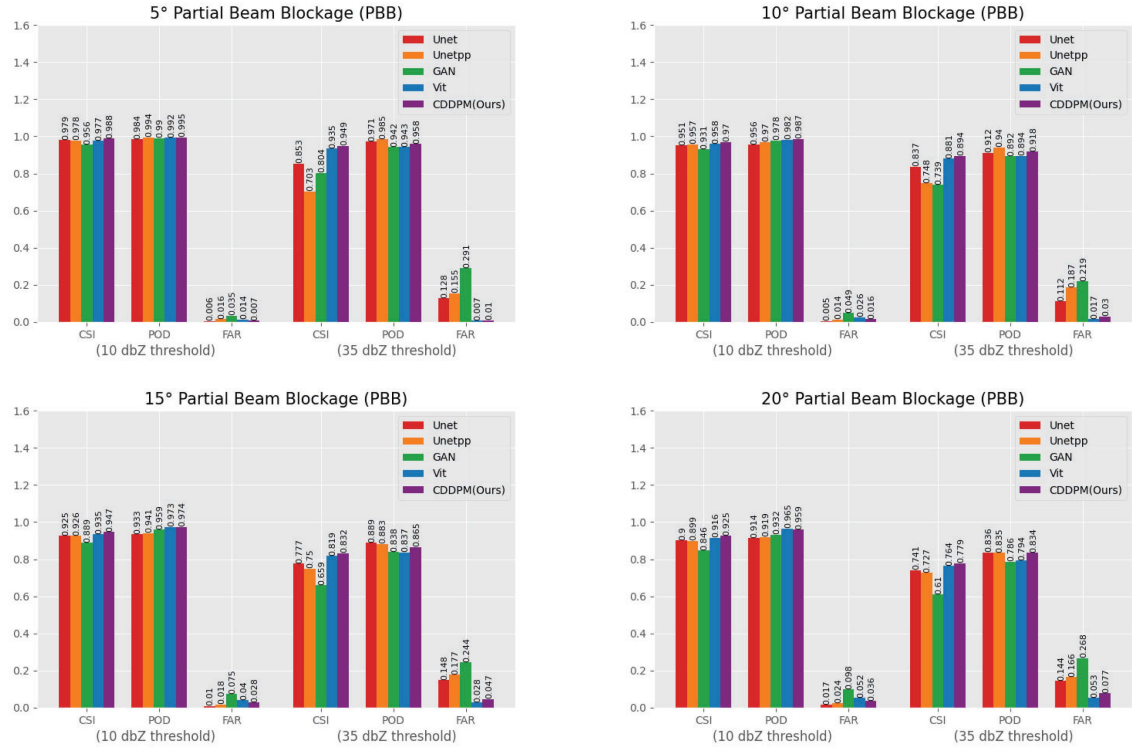

Fig. 10. Evaluation results of the beam blockage correction performance under various restoration methods, blockage sizes, and reflectance threshold levels.

Fig. 9 illustrates the comparative restoration results of our method against other models under different obstruction angles. It is evident that our approach consistently achieves more realistic and accurate restoration results under PBB at any angle. As the obstruction angle increases, all models' restoration performance tends to decline, with the results becoming smoother or less accurate. Our method effectively addresses both of these challenges, which makes it highly adaptable to various complex scenarios.

In Table III, we present the data reconstruction metrics for different obstruction angles. From the results, it is evident that our method excels in reconstructing fine details with greater realism. ViT also exhibits excellent performance in restoration under smaller angle obstructions but exhibits issues with excessive smoothing (as shown in Fig. 9), resulting in its low PSNR and SSIM. The evaluation of restoration quality metrics, such as PSNR and SSIM, still indicates subpar performance. As the obstruction angle increases, the other methods' performances show a notable decline, while our approach maintains exceptional performance metrics.

Fig. 10 displays the accuracy of reflectivity information restoration for thresholds greater than 10 dBZ and 35 dBZ at four different obstruction angles. While GAN manages to overcome the smoothing issue in the restoration results (as illustrated by Fig. 9), it suffers from significant inaccuracies. Compared with the other models, our approach consistently exhibits superior performance metrics.

IV. DISCUSSION

The proposed CDDPM demonstrates significant performance advantages in radar data completion tasks, effectively capturing complex data distributions and correlations. Compared with the other deep learning methods, our model exhibits remarkable performance improvements. Moreover, our research findings yield valuable insights for further exploration in the field of

condition-guided remote sensing data generation. DDPM has been empirically demonstrated on discrete remote sensing data, specifically radar data.

Future research directions could investigate the application of condition-guided diffusion models in various remote sensing tasks, including data fusion, data evolution, and accurate forecasting of strong convective meteorological events. In addition, efforts to enhance the model's robustness and generalization capabilities are warranted to address more intricate and realistic scenarios.

It is worth noting that our approach employed a simplistic MSE loss function without assigning specific weights to areas with high echo loss. The fixed learning rate configuration also leaves room for further model performance optimization. Furthermore, the validation process was restricted to a limited geographical area, which potentially limited the results' generalizability, considering regional variation in convective weather conditions. Future investigations should extend testing and validation to broader geographic regions encompassing diverse climate conditions to ascertain the proposed model's feasibility and practicality. Our research outcomes are relevant to a broad range of remote sensing applications, facilitating the repair of low-quality radar data and improving the accuracy of severe convective weather predictions.

Although our findings demonstrate the great potential of generative models for inpainting missing areas in radar data, such models may hallucinate unreal features, have serious ethical considerations, and may cause harm in the real world. Researchers must carefully consider these risks in implementing generative models.

V. Conclusion

Seamless fusion of radar imagery is crucial for data processing and quantitative applications in the field of meteorology. In this study, we have developed and implemented a CDDPM for radar data restoration in the presence of missing information. To address the computational challenges posed by self-attention mechanisms, we employed channel fusion in our model. We trained our restoration model using radar reflectivity data from the midwestern region of Kansas, USA, sourced from MRMS, and conducted testing on an independent dataset from a different year. We also compared our approach with various other deep learning methods. The key findings of this research may be summarized as follows.

1) Visually, our model effectively addresses the issues of excessive smoothing and inaccuracies commonly encountered when applying traditional deep learning methods to radar data. It also demonstrates outstanding restoration performance in cases of extreme radar outage and extensive beam blockage.
2) According to the evaluation results, our method shows excellent repair performance on radar data with large-scale missing regions.

This study's most significant finding is that the conditionally guided diffusion model is highly effective for radar reflectivity data with highly discrete data distributions, demonstrating effective restoration of details and accuracy in high-echo regions while simultaneously addressing the smoothing issue in the restoration results while addressing the issue of smoothness in the generated results. Both characteristics have highly significant implications for the remote sensing and meteorology fields.

References

[1] S. Shun Liu, "WSR-88D radar data processing at NCEP," *Weather Forecasting*, vol. 31, no. 6, pp. 2047–2055, 2016.

[2] P. Coulibaly and N. Evora, "Comparison of neural network methods for infilling missing daily weather records," *J. Hydrol.*, vol. 341, no. 1/2, pp. 27–41, 2007.

[3] J. Hubbert, M. Dixon, S. Ellis, and G. Meymaris, "Weather radar ground clutter. Part I: Identification, modeling, and simulation," *J. Atmospheric Ocean. Technol.*, vol. 26, no. 7, pp. 1165–1180, 2009.

[4] B. Isom et al., "Detailed observations of wind turbine clutter with scanning weather radars," *J. Atmospheric Ocean. Technol.*, vol. 26, no. 5, pp. 894–910, 2009.

[5] S. E. Giangrande and A. V. Ryzhkov, "Calibration of dual-polarization radar in the presence of partial beam blockage," *J. Atmospheric Ocean. Technol.*, vol. 22, no. 8, pp. 1156–1166, 2005.

[6] B. Hapke, "Coherent backscatter and the radar characteristics of outer planet satellites," *Icarus*, vol. 88, no. 2, pp. 407–417, 1990.

[7] S. PC and M. Maki, "Application of a modified digital elevation model method to correct radar reflectivity of x-band dual-polarization radars in mountainous regions," *Hydrological Res. Lett.*, vol. 8, no. 2, pp. 77–83, 2014.

[8] A. V. Ryzhkov, S. E. Giangrande, V. M. Melnikov, and T. J. Schuur, "Calibration issues of dual-polarization radar measurements," *J. Atmospheric Ocean. Technol.*, vol. 22, no. 8, pp. 1138–1155, 2005.

[9] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 29, no. 6, pp. 1153–1160, Dec. 1981.

[10] C. Langston, J. Zhang, and K. Howard, "Four-dimensional dynamic radar mosaic," *J. Atmospheric Ocean. Technol.*, vol. 24, no. 5, pp. 776–790, 2007.

[11] E. Ruzanski and V. Chandrasekar, "Weather radar data interpolation using a kernel-based Lagrangian nowcasting technique," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3073–3083, Jun. 2015.

[12] D. T. Larose and C. D. Larose, "K-nearest neighbor algorithm," Wiley Data and Cybersecurity, 2014.

[13] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, and Y. Akbari, "Image inpainting: A review," *Neural Process. Lett.*, vol. 51, pp. 2007–2028, 2020.

[14] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, 2015, pp. 234–241.

[15] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.

[16] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2536–2544.

[17] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–14, 2017.

[18] Y. Deng, S. Hui, S. Zhou, D. Meng, and J. Wang, "Learning contextual transformer network for image inpainting," in *Proc. 29th ACM Int. Conf. Multimedia*, 2021, pp. 2529–2538.

[19] H. Wu, Q. Liu, X. Liu, Y. Zhang, X. Xu, and M. Bilal, "A FCN approach to blockage correction in radars," in *Proc. IEEE Int. Conf. Dependable Autonomic Secure Comput., Int. Conf. Pervasive Intell. Comput., Int. Conf. Cloud Big Data Comput., Int. Conf. Cyber. Sci. Technol. Congr.*, 2021, pp. 482–487.

[20] P. Lepetit et al., "Using deep learning for restoration of precipitation echoes in radar data," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Feb. 19, 2021, Art. no. 5100914, doi: 10.1109/TGRS.2021.3052582.

[21] A. Geiss and J. C. Hardin, "Inpainting radar missing data regions with deep learning," *Atmospheric Meas. Techn.*, vol. 14, no. 12, pp. 7729–7747, 2021.

[22] S. Tan and H. Chen, "A conditional generative adversarial network for weather radar beam blockage correction," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, Jun. 14, 2023, Art. no. 4103014, doi: 10.1109/TGRS.2023.3286181.

[23] S. Tan, H. Chen, S. Yao, and V. Chandrasekar, "Weather radar beam blockage correction using deep learning," in *Proc. United States Nat. Committee URSI Nat. Radio Sci. Meeting*, 2023, pp. 296–297.

[24] L. Xie, Q. Zhao, J. Huo, and G. Cheng, "A ground penetrating radar data reconstruction method based on generation networks," in *Proc. IEEE Radar Conf.*, 2020, pp. 1–4.

[25] D. Atlas, R. Srivastava, and R. S. Sekhon, "Doppler radar characteristics of precipitation at vertical incidence," *Rev. Geophys.*, vol. 11, no. 1, pp. 1–35, 1973.
[26] K. Lengfeld et al., "Use of radar data for characterizing extreme precipitation at fine scales and short durations," *Environ. Res. Lett.*, vol. 15, no. 8, 2020, Art. no. 085003.
[27] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proc. 20th Int. Conf. Pattern Recognit.*, 2010, pp. 2366–2369.
[28] B. Petrovska, E. Zdravevski, P. Lameski, R. Corizzo, I. Štajduhar, and J. Lerga, "Deep learning for feature extraction in remote sensing: A case-study of aerial scene classification," *Sensors*, vol. 20, no. 14, 2020, Art. no. 3906.
[29] P. Wang, W. Zheng, T. Chen, and Z. Wang, "Anti-oversmoothing in deep vision transformers via the Fourier domain analysis: From theory to practice," 2022, *arXiv:2203.05962*.
[30] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 6840–6851.
[31] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool, "Repaint: Inpainting using denoising diffusion probabilistic models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 11461–11471.
[32] S. Xie, Z. Zhang, Z. Lin, T. Hinz, and K. Zhang, "Smartbrush: Text and shape guided object inpainting with diffusion model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 22428–22437.
[33] L. Yang et al., "Diffusion models: A comprehensive survey of methods and applications," 2022, *arXiv:2209.00796*.
[34] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2256–2265.
[35] Y. Song and S. Ermon, "Generative modeling by estimating gradients of the data distribution," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019.
[36] E. Paquet and H. L. Viktor, "Molecular dynamics, Monte Carlo simulations, and Langevin dynamics: A computational review," *BioMed Res. Int.*, vol. 2015, 2015, Art. no. 183918.
[37] J. Zhang et al., "Multi-radar multi-sensor (MRMS) quantitative precipitation estimation: Initial operating capabilities," *Bull. Amer. Meteorological Soc.*, vol. 97, no. 4, pp. 621–638, 2016.
[38] B. J. Etherton, M. S. Wandishin, L. A. Paulik, G. J. Layne, and M. A. Petty, "Assessment of the multi-radar/multi-sensor system (MRMS) and the corridor integrated weather system (CIWS)," 2014.
[39] D. F. Jorgensen, "Mesoscale and convective-scale characteristics of mature hurricanes. Part I: General observations by research aircraft," *J. Atmospheric Sci.*, vol. 41, no. 8, pp. 1268–1286, 1984.
[40] W. Zhang, H. Chen, L. Han, R. Zhang, and Y. Ge, "Pixel-CRN: A new machine learning approach for convective storm nowcasting," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, Apr. 17, 2023, Art. no. 4102212, doi: 10.1109/TGRS.2023.3267976.
[41] G. Brauwers and F. Frasincar, "A general survey on attention mechanisms in deep learning," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 4, pp. 3279–3298, Apr. 2023.
[42] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 9, pp. 10850–10869, Sep. 2023.
[43] C. Zhang, C. Zhang, M. Zhang, and I. S. Kweon, "Text-to-image diffusion model in generative AI: A survey," 2023, *arXiv:2303.07909*.
[44] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," 2020, *arXiv:2010.02502*.
[45] J. T. Schaefer, "The critical success index as an indicator of warning skill," *Weather Forecasting*, vol. 5, no. 4, pp. 570–575, 1990.
[46] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "U-net : A nested U-net architecture for medical image segmentation," in *Proc. 4th Int. Workshop Deep Learn. Med. Image Anal. Multimodal Learn. Clin. Decis. Support, 8th Int. Workshop Held Conjunction*, 2018, pp. 3–11.
[47] I. Goodfellow et al., "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020.
[48] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
[49] J. A. Sethian, "Fast marching methods," *SIAM Rev.*, vol. 41, no. 2, pp. 199–235, 1999.
[50] C. A. Doswell III, "Severe convective storms– An overview," *Severe Convective Storms*, Boston, MA: American Meteorological Society, pp. 1–26, 2001, doi: 10.1007/978-1-935704-06-5_1.
[51] Y. Zhao, C. Li, P. Yu, J. Gao, and C. Chen, "Feature quantization improves GAN training," 2020, *arXiv:2004.02088*.
[52] P. Dhariwal and A. Nichol, "Diffusion models beat GANs on image synthesis," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 8780–8794.

Wei Zhang (Member, IEEE) received the B.Sc. and M.Sc. degrees in applied mathematics from the Northeastern University of China, Shenyang, China, in 1998 and 2001, and the Ph.D. degree in computer science and technology from the University of Science and Technology of China, Hefei, China, in 2004.

From 2004 to 2006, he conducted postdoctoral research with the Joint Doctoral Workstation with the iflytek.com and the University of Science and Technology of China. From 2013 to 2014, he was an Academic Visitor with the School of Informatics, University of Edinburgh, Edinburgh U.K. Currently, he is an Associate Professor with the College of Information Science and Engineering, Ocean University of China, Qingdao, China. His research interests include artificial intelligence for remote sensing and meteorology.

Xinyu Zhang received the bachelor's degree in computer science and technology from Liaocheng University, Liaocheng, China in 2017. He is currently working toward the master's degree with the College of Computer Science and Technology, Ocean University of China, Qingdao, China.

His research interests include deep learning for remote sensing and meteorology.

Zhuyu Jin received the bachelor's degree in measurement and control technology and instrument from the North China University of Water Resources and Electric Power, Zhengzhou, China, in 2018, and the M.Sc. degree in physical oceanography from the National Marine Environmental Forecasting Center of China, Beijing, China.

He is currently a Researcher with the Qingdao Joint Institute of Marine Meteorology, Chinese Academy of Meteorological Sciences, Bijing. His research interests analysis of ocean wind field characteristics and research on wind speed correction technology based on CCMP

Youqi Wen received the bachelor's degree in communication engineering from the Central South University of Forestry and Technology, Changsha, China, in 2021. He is currently working toward the master's degree from the College of Computer Science and Technology, Ocean University of China, Qingdao, China.

His research interests include machine learning for remote sensing.

Jie Liu received the bachelor's degree in electronic information from the China University of Geosciences, Wuhan, China, in 2022. He is currently working toward the master's degree from the College of Computer Science and Technology, Ocean University of China, Qingdao, China. His research interests include deep learning for remote sensing.