Diffusion Model With Gradient Descent Module Guiding Reconstruction for Single-Pixel Imaging

Chen Huang, Qiurong Yan, *Member, IEEE*, Jinwei Yan, Yi Li, Xiaolong Luo, and Hui Wang

***Abstract*—Reconstructing high-quality images with few measurements has always been a primary goal for single-pixel imaging (SPI). Diffusion models have shown outstanding performance in image generation and have been effectively attempted in image reconstruction for ghost imaging. However, there is still a great deal of space for improvement in the quality of image reconstruction at low sampling rates. Inspired by the proximal gradient descent algorithm (PGD), we propose Diffusion Model with Gradient Descent Module Guiding Reconstruction for Single-Pixel Imaging. The gradient descent module in PGD is utilized for preliminary image reconstruction. The preliminary reconstruction serves as prior information to iteratively constrain the diffusion model, allowing it to generate target images consistent with the training data distribution. Additionally, the strong mapping ability of the diffusion model replaces the traditional proximal operator to accelerate convergence. Full connected sampling and convolutional sampling are proposed as alternative sampling methods to the traditional Gaussian random matrix sampling. Sampling and generation are optimized jointly to capture key image information and improve reconstruction accuracy. Simulations and experiments confirm that our proposed network can significantly improve the quality of image reconstruction at low measurement rates.**

***Index Terms*—Compressed sensing (CS), single pixel imaging (SPI), diffusion models(DMs), proximal gradient descent.**

I. Introduction

Single-pixel imaging (SPI) is a technology that uses point detectors to image objects. It has significant advantages in detection sensitivity, broadening spectral response range, and reducing imaging costs. With the development of compressed sensing (CS) theory, SPI based on CS has attracted much attention because of its elegant fusion of optics, mathematics, and optimization theory. Current solutions mainly focus on designing more efficient coding modes. This ensures that more information is obtained per measurement [1], [2], [3]. Additionally, new optimization algorithms have been developed to achieve better reconstruction results with fewer measurements [4], [5]. Nevertheless, the trade-off between sampling time and image quality is still a limitation of its practical application.

To overcome the limitations of SPI, deep learning [6] is widely used in SPI pattern generation and image reconstruction. Compared to traditional iterative reconstruction methods, deep learning algorithms can significantly improve both reconstruction speed and quality [5], [7], [8]. Methods can be categorized into three categories: The first category includes end-to-end reconstruction networks like Reconnet [9] and CSNet [10]. The second category comprises model unfolding networks, such as ISTA-Net [11] and GCDUN [12]. The third category involves generative models, such as generative adversarial network [13] and diffusion models [14].

With the development of generative models, generative models have been widely used in image generation, image denoising, image restoration, and other fields [15], [16], [17]. Different from learning direct mapping, generative models learn data distributions based on probability and statistical knowledge, and use these distributions to generate image samples. Generative models include Energy-Based Models (EBMs) [18], [19], Generative Adversarial Networks (GANs) [13], [20], normalizing flows (NFs) [21], [22], Variational Autoencoders (VAEs) [23], [24], and diffusion models [14], [25], [26]. Diffusion models effectively overcome the obstacles caused by the alignment of posterior distribution in VAEs, reduce the inherent instability of GANs against targets, simplify the complex training process relying on Markov chain Monte Carlo (MCMC) method in EBMs, and perform network constraints similar to NFs, showing superior performance [27]. Using the denoising score-matching target, diffusion models train the neural network to estimate the score function [28], provide a more stable training target than GANs, and are superior to VAEs, EBMs, and NFs in terms of generation quality [26], [29].

In 2022, Denoising Diffusion Restoration Models (DDRM) [30] was proposed as the first sampling-based inverse problem solver, effectively generating a series of high-quality, diverse, and effective solutions for general content images. In 2023, DiffIR [31] was introduced as a powerful, simple, and efficient reconstruction benchmark based on diffusion models. These studies show that diffusion models have achieved remarkable results in image synthesis [14], [26], [29], [32] and image restoration (IR) tasks (such as inpainting [33], [34] and super resolution [35]). Shuai Mao [17] first applied the diffusion

Manuscript received 9 June 2024; revised 19 July 2024; accepted 23 July 2024. Date of publication 29 July 2024; date of current version 6 August 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62165009 and Grant 61865010, and in part by the Cultivation Program for Academic and Technical Leaders in Major Disciplines of Jiangxi Province under Grant 20225BCJ22021. *(Corresponding author: Qiurong Yan.)*

Chen Huang, Qiurong Yan, Jinwei Yan, Yi Li, and Xiaolong Luo are with the School of Information Engineering, Nanchang University, Nanchang 330031, China (e-mail: chenhuang@email.ncu.edu.cn; yanqiurong@ncu.edu.cn; yjw@email.ncu.edu.cn; xgly@email.ncu.edu.cn; longuplive@163.com).

Hui Wang is with the College of Physics and Communication Electronics, Jiangxi Normal University, Nanchang 330022, China (e-mail: wanghuiyy35@163.com).

Digital Object Identifier 10.1109/JPHOT.2024.3434972

© 2024 The Authors. This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

model to ghost imaging and achieved amazing results. However, since the measured values are directly multiplied by the pseudo-inverse of the measurement matrix as the conditional input to the diffusion model, in the case of insufficient sampling, the limited sampling data may correspond to multiple possible target images. Therefore the reconstruction performance can be further improved.

When the measurement rate is less than 0.1, the results of traditional algorithms are usually affected by background noise or blurred lines, resulting in poor recognition. This is due to the lack of features. If we can add a generative part to the traditional model and use its imagination to extend the existing features under low measurement rate conditions, it is possible to break through the limitation of measurement rate on clarity [36], [37], [38]. Therefore, this study uses diffusion models (DMs), which are based on the noise diffusion process and Bayesian theory, to generate target images that correspond to the training data distribution. Compared to GANs, DMs can map random Gaussian noise to complex target distributions with high quality, avoiding mode collapse and training instability.

To enhance reconstruction performance at low measurement rates, we propose Diffusion Model with Gradient Descent Module Guiding Reconstruction for Single-Pixel Imaging. The gradient descent module in PGD is used for preliminary image reconstruction. The preliminary reconstruction provides prior information to guide the diffusion model. This enables the generation of target images that match the training data distribution. As a result, high-resolution reconstruction is achieved even at very low sampling rates. The powerful mapping ability of the diffusion model replaces the traditional proximal operator, accelerating convergence. Fully connected sampling and convolutional sampling are designed to replace traditional random Gaussian matrix sampling, making the sampling matrix learnable parameters. In this way, the sampling and generation are jointly optimized to obtain key image information and improve the image reconstruction quality. Moreover, by training the sampling matrix in binary form, our proposed network can be applied to SPI systems. Simulations and physical experiments verify the effectiveness of the proposed network. The contributions of this study are as follows:

- We propose a **D**M-based with **G**radient Descent Module Guiding **R**econstruction **N**etwork (DGRN) for SPI systems. We use the gradient descent module in PGD to perform preliminary image reconstruction from the sampled data. The preliminary reconstruction serves as a condition to direct the diffusion model's generation direction for image reconstruction.
- The learnable sampling matrix is input into the forward process to achieve the joint optimization of sampling and generation. This design reduces estimation error and enhances the robustness of the system.
- By training the sampling matrix in the network into binary, our proposed network can be applied to SPI systems, and has been verified through experiments. The experimental results show that the designed network can reconstruct the main characteristics of images even with few measurements.

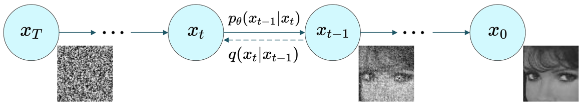

Fig. 1. DDPM, diffusion and reverse process.

II. Related Work

A. Diffusion Models

As shown in Fig. 1, diffusion models aim to learn the data distribution $\mathbf{p}(\mathbf{x})$ by denoising the normal distribution variables progressively. The forward diffusion process defines a Markov Chain where random noise is iteratively added to a given real image $\mathbf{x_0} \sim \mathbf{q}(\mathbf{x_0})$ until the distribution becomes an independent Gaussian distribution. On the contrary, the reverse diffusion process is to recover the original image from the Gaussian noise which also follows a Markov Chain process.

The forward process does not contain learnable parameters. Firstly, define the real data distribution $\mathbf{x_0} \sim \mathbf{q}(\mathbf{x_0})$ and Markov noise process. Then add Gaussian noise to the data, and add a total of $\mathbf{T}$ steps to generate a series of noisy samples $\mathbf{x_1} \sim \mathbf{x_T}$. The noise addition's mean and variance at each timestep are calculated by $\beta_\mathbf{t}$. The diffusion process can be formulated as follows:

$$\mathbf{q}(\mathbf{x_t}|\mathbf{x_{t-1}}) = \mathcal{N}(\mathbf{x_t}; \sqrt{\mathbf{1}-\beta_\mathbf{t}}\,\mathbf{x_{t-1}}, \beta_\mathbf{t}\mathbf{I}) \tag{1}$$

Ho et al. [14] note that we need not apply $\mathbf{q}(\mathbf{x_t}|\mathbf{x_{t-1}})$ repeatedly to sample from $\mathbf{x_t} \sim \mathbf{q}(\mathbf{x_t}|\mathbf{x_0})$. Instead, $\mathbf{q}(\mathbf{x_t}|\mathbf{x_0})$ can be expressed as a Gaussian distribution with $\alpha_\mathbf{t} = \mathbf{1} - \beta_\mathbf{t}$ and $\overline{\alpha}_\mathbf{t} = \prod_{\mathbf{s=0}}^{\mathbf{t}} \alpha_\mathbf{s}$.

$$\mathbf{q}(\mathbf{x_t}|\mathbf{x_0}) = \mathcal{N}(\mathbf{x_t}; \sqrt{\overline{\alpha}_\mathbf{t}}\,\mathbf{x_0}, (\mathbf{1}-\overline{\alpha}_\mathbf{t})\mathbf{I}) \tag{2}$$

The reverse process aims to infer the conditional probability $\mathbf{q}(\mathbf{x_{t-1}}|\mathbf{x_t})$ to gradually recover the original data $\mathbf{x_0}$ from the Gaussian noise $\mathbf{x_T}$. Sohl-Dickstein et al. [25] pointed out that $\mathbf{q}(\mathbf{x_{t-1}}|\mathbf{x_t})$ tends a diagonal Gaussian distribution when $\mathbf{T} \to \infty$ and correspondingly $\beta_\mathbf{t} \to \mathbf{0}$, so we can use the $\mathbf{p}_\theta(\mathbf{x_{t-1}}|\mathbf{x_t})$ obtained by deep network fitting to approximate $\mathbf{q}(\mathbf{x_{t-1}}|\mathbf{x_t})$. Therefore, in the reverse process, at each timestep $\mathbf{t}$, the deep network model estimates the predicted value $\tilde{\mathbf{z}}_\mathbf{t}$ of random noise $\mathbf{z}$ using $\mathbf{x_t}$ and $\mathbf{t}$. This estimation continues iteratively until $\mathbf{x_0}$ is obtained through calculation. Based on the Bayesian posterior probability, the backward process can be expressed as follows [14]:

$$\begin{aligned}
&\mathbf{p}_\theta(\mathbf{x_{t-1}}|\mathbf{x_t}) \sim \mathcal{N}(\mu, \sigma^\mathbf{2}\mathbf{I}) \\
&\sigma^\mathbf{2} = \frac{\mathbf{1}-\overline{\alpha}_\mathbf{t-1}}{\mathbf{1}-\overline{\alpha}_\mathbf{t}}(\mathbf{1}-\alpha_\mathbf{t}) \\
&\mu = \frac{\mathbf{1}}{\sqrt{\alpha_\mathbf{t}}}\left(\mathbf{x_t} - \frac{\mathbf{1}-\alpha_\mathbf{t}}{\sqrt{\mathbf{1}-\overline{\alpha}_\mathbf{t}}}\tilde{\mathbf{z}}_\mathbf{t}\right) \\
&\tilde{\mathbf{z}}_\mathbf{t} = \mathbf{Z}_\theta(\mathbf{x_t}, \mathbf{t})
\end{aligned} \tag{3}$$

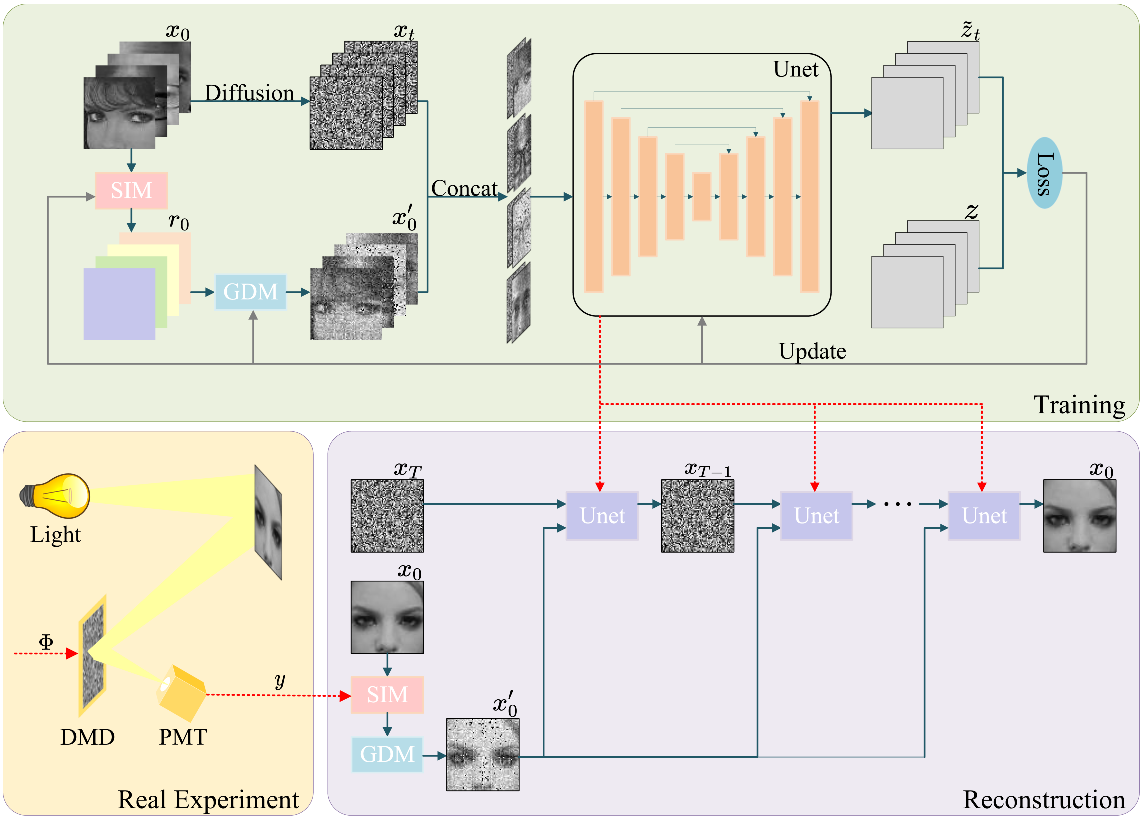

Fig. 2. The network structure diagram of DGRN. Specifically, DGRN consists of three modules: the Sampling and Initialization Module(SIM), the Gradient Descent Module(GDM), and the UNet-based deep network module. SIM and GDM together constitute an encoder-decoder network for low-quality preliminary reconstruction of the input image. The preliminary reconstructed image $\mathbf{x_0'}$ is concatenated with the noise image after a certain number of timesteps $\mathbf{x_t}$ in the channel dimension and then input into UNet for generation. The prediction noise $\tilde{\mathbf{z}}_\mathbf{t}$ is generated through training. Then, the loss of random Gaussian noise $\mathbf{z}$ and prediction noise $\tilde{\mathbf{z}}_\mathbf{t}$ is calculated, and the network parameters are updated via backpropagation.

B. *Proximal Gradient Descent Algorithm*

In SPI based on CS, we represent the compression measurement process as:

$$\mathbf{y} = \mathbf{\Phi x} + \epsilon \tag{4}$$

where $\mathbf{x} \in \mathbf{R^N}$ is original signal and $\mathbf{\Phi} \in \mathbf{R^{M*N}}$ is the measurement matrix($\mathbf{M} << \mathbf{N}$), $\mathbf{y} \in \mathbf{R^M}$ is the measured value, $\boldsymbol{\epsilon}$ represent noise. Reconstructing $\mathbf{x}$ by $\mathbf{y}$ is the solution to the underdetermined problem.

Based on the sparse prior, This recovery process can be expressed as the following energy function:

$$\mathbf{x} = \arg\min_{\mathbf{x}} \frac{\mathbf{1}}{\mathbf{2}} ||\mathbf{y} - \mathbf{\Phi x}||_\mathbf{2}^\mathbf{2} + \lambda \mathbf{J}(\mathbf{x}) \tag{5}$$

where $\boldsymbol{\lambda}$ is a hyper-parameter to weight the regularization term $\mathbf{J}(\mathbf{x})$.

Technically, PGD approximatively expresses (5) as an iterative convergence problem through the two subproblems: gradient descent (6) and proximal mapping (7):

$$\mathbf{r}^{(k)} = x^{(k-1)} - \rho \Phi^T \left(\Phi x^{(k-1)} - y\right) \tag{6}$$

$$\mathbf{x}^{(k)} = prox_{\lambda,J}(r^{(k)}) \tag{7}$$

PGD iteratively updates $\mathbf{r^k}$ and $\mathbf{x^k}$ until convergence. ISTA-Net [39] is a typical PGD-based algorithm in which the regulation term is defined as an $\mathbf{l_1}$ norm, $\mathbf{J}(\mathbf{x}) = ||\mathbf{x}||_\mathbf{1}$.

Inspired by [40], which collaboratively trained an UNet to serve as the proximal mapping in ADMM algorithm [41], in this study we apply the gradient descent module of PGD to guide the preliminary reconstruction of the image. The preliminary reconstruction module obtained by the gradient descent module is input into UNet. In each iteration, UNet can learn the characteristics of the data to update the parameters. Furthermore, it can perform constraint processing to replace the traditional proximal mapping operation, thereby improving the generalization ability and anti-overfitting ability of the model.

III. Proposed Network

We go into further detail about our proposed DGRN for SPI in this section. We will introduce the architecture and internal components of DGRN.

A. *Overview of DGRN*

The network structure is shown in Fig. 2, mainly composed of three modules: the Sampling and Initialization Module(SIM), the Gradient Descent Module(GDM), and the UNet-based deep

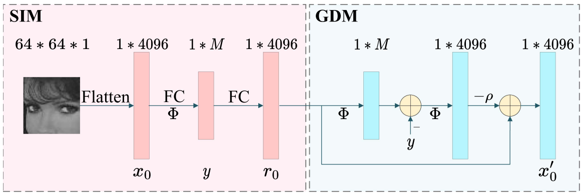

Fig. 3. Structure diagram of SIM and GDM based on fully connected layer.

network module. SIM and GDM together constitute an encoder-decoder network for low-quality preliminary reconstruction of the input image. The first layer of SIM simulates the compressed sampling process in SPI, and the layer's weight matrix serves as the measurement matrix. Unlike traditional random matrix, the measurement matrix is learnable. Inspired by diffusion models, the preliminary reconstructed image $\mathbf{x_0'}$ is concatenated with the noise image after a certain number of timesteps $\mathbf{x_t}$ in the channel dimension and then input into UNet for generation. The preliminary reconstructed image $\mathbf{x_0'}$ serves as a condition to guide the generation model learning the internal statistical distribution of the sample. The prediction noise $\mathbf{\tilde{z}_t}$ is generated through training. Then, the loss of random Gaussian noise $\mathbf{z}$ and prediction noise $\mathbf{\tilde{z}_t}$ is calculated, and the network parameters are updated via backpropagation.

The well-trained network can perform high-quality image reconstruction through the reverse diffusion process. The network predicts the noise $\mathbf{\tilde{z}_t}$ at each timestep $\mathbf{t}$ and uses these predictions to reduce the noise in the data. This process starts with random Gaussian noise as input $\mathbf{x_T}$. As time $\mathbf{t}$ regresses from $\mathbf{T}$ to $\mathbf{0}$, $\mathbf{x_{t-1}}$ is computed by (3), so as to gradually reconstruct the original image $\mathbf{x_0}$ from the learned prior distribution. During the iterative reconstruction, the initial reconstructed image $\mathbf{x_0'}$ serves as a data consistency term to limit the generation direction of the model.

In our numerical simulation experiments, both network training and testing are carried out in the process of simulating SPI using common photodetectors. In this process, we obtain the measured value input into the network by multiplying the measurement matrix with the gray matrix representing the light intensity. This method effectively simulates the working principle of common photodetectors in SPI systems.

B. SIM and GDM

In this section, we introduce the Sampling and Initialization Module(SIM) and the Gradient Descent Module(GDM) in detail. The encoding-decoding structure together forms a sampling and preliminary reconstruction network for low-quality image reconstruction of the input image. A fully connected sampling method and a convolutional sampling method were designed. Corresponding gradient descent modules were designed for each sampling method.

1) SIM and GDM Based on Fully Connected Layer: In this section, we introduce SIM and GDM based on the fully connected layer (Fig. 3). SIM consists of two fully connected layers. The original image size of the input is $\mathbf{64*64}$, and is flattened with the dimension of $\mathbf{x} \in \mathbf{R^{1*4096}}$. The first fully connected layer's output can be regarded as the measured value $\mathbf{y} \in \mathbf{R^{1*M}}$. The measured value $\mathbf{y}$ is mapped to the initialization $\mathbf{r_0}$ by the second fully connected layer. The relationship between the measurement rate($\mathbf{MR}$) and $\mathbf{M}$ can be described as follows:

$$\mathbf{MR} = \frac{\mathbf{M}}{\mathbf{64*64}} * \mathbf{100\%} \tag{8}$$

In our designed network, the weight matrix $\mathbf{\Phi}$ of sampling is used as a learnable parameter. Previous research indicates that using a fully connected layer for sampling not only provides adequate weight for image reconstruction but also decreases training time [42].

In the preliminary reconstruction of the image, the gradient descent module is employed. During the noise addition at each timestep of the forward diffusion, the network achieves rapid convergence and avoids local optima by repeatedly updating $\mathbf{x_0'}$ and $\mathbf{y}$ until convergence. The specific operation is shown in (9).

$$\mathbf{x_0'} = \mathbf{r_0} - \rho\mathbf{\Phi^T}(\mathbf{\Phi r_0} - \mathbf{y}) \tag{9}$$

2) SIM and GDM Based on Convolutional Layer: We refer to the subpixel convolution approach introduced by Shi et al. [43]. Fig. 4 shows the structure of the convolution sampling reconstruction network, and the size of the input image is $\mathbf{64*64*1}$. The first convolutional layer is composed of $\mathbf{M}$ convolution kernels. The size of each convolution kernel is $\mathbf{16*16}$, and the stride is $\mathbf{16}$, so the sampling matrix $\mathbf{\Phi}$ can be expressed as $\mathbf{R^{16*16*M}}$, resulting in measurements of $\mathbf{4*4*M}$. Then the sub-pixel interpolation method is used to generate a sampling result with dimensions of $\mathbf{4\sqrt{M}*4\sqrt{M}*1}$.

The up-sampling sub-network uses the measurement results from the down-sampling sub-network as its input. We use $\mathbf{256}$ kernels of size $\mathbf{\sqrt{M}*\sqrt{M}}$ with a stride of size $\mathbf{\sqrt{M}}$, producing an up-sampling result of dimensions $\mathbf{4*4*256}$. Then the sub-pixel interpolation method is used to generate a preliminary reconstructed image with a size of $\mathbf{64*64*1}$. The relationship between $\mathbf{MR}$ and $\mathbf{M}$ can be formulated as follows:

$$\mathbf{MR} = \frac{\mathbf{16*M}}{\mathbf{64*64}} * \mathbf{100\%} \tag{10}$$

Due to the use of convolution for sampling, the measured value is not a single-dimensional array, so it can not be directly calculated by (9). In (9), $\mathbf{\Phi x_0}$ can be regarded as a sampling process, $\mathbf{\Phi^T}(\mathbf{\Phi x_0} - \mathbf{y})$ can be regarded as a reverse sampling process. As shown in Fig. 4, in GDM, convolution and deconvolution are used. the process can be formulated as follows:

$$\mathbf{x_0'} = \mathbf{r_0} - \rho\mathbf{\Phi}\tilde{*}(\mathbf{\Phi} * \mathbf{r_0} - \mathbf{y}) \tag{11}$$

$*$ means convolution, and $\tilde{*}$ means deconvolution.

C. UNet-Based Deep Network Module

As shown in Fig. 5, the preliminary reconstructed image $\mathbf{x_0'}$ is concatenated with the noise image after a certain number of timesteps $\mathbf{x_t}$ in the channel dimension, then input into the UNet-based deep network module for generation and return the predicted noise $\mathbf{\tilde{z}_t}$. In terms of network architecture, we adopt the same UNet structure as DDPM [14]. First, the input

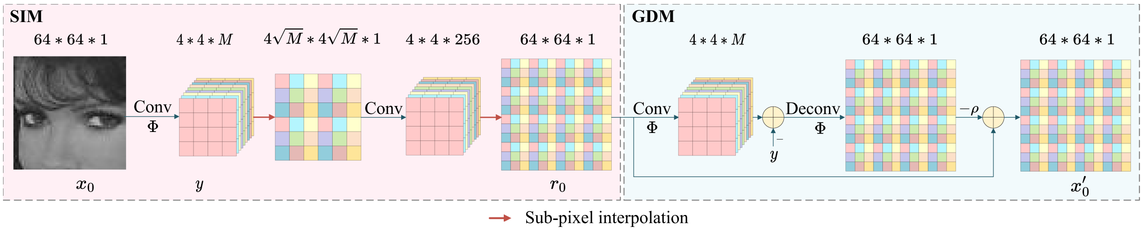

Fig. 4. Structure diagram of SIM and GDM based on convolutional layer.

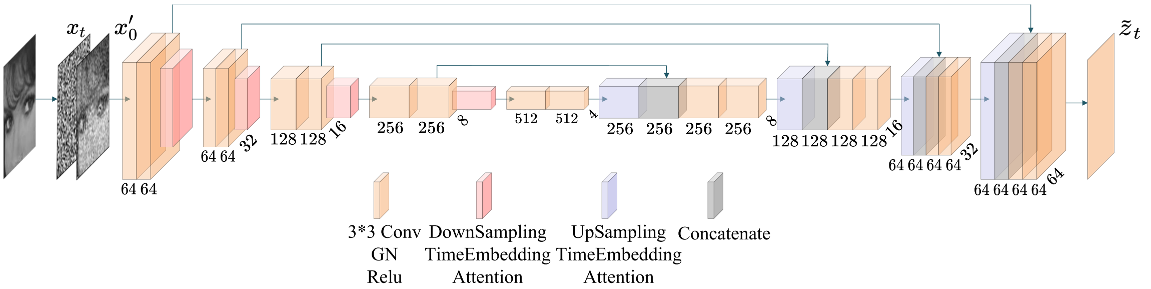

Fig. 5. The network structure diagram of UNet.

is down-sampled and then up-sampled. In addition, the depth of the network also has a certain impact on the reconstruction performance, which will be discussed in detail in Section IV-C3.

To introduce nonlinear factors, we use Group Normalization [44] and Attention [45] to enhance the ability of the network to process data. Since the network needs to process input images at any given timestep, similar to the position encoding in Transformer [45], we add additional timestep information to the network by time embedding.

These methods together constitute the architecture of the UNet-based deep network module, which can effectively process noise images $\mathbf{x_t}$ and generate accurate prediction noise $\tilde{\mathbf{z}}_\mathbf{t}$, improving the reconstruction effect and the overall performance of the network.

D. Network Training

In the training phase, the original image $\mathbf{x_0}$ is inputted. At each timestep from $\mathbf{0}$ to $\mathbf{T}$, the measured value $\mathbf{y}$ is obtained through convolutional or fully connected layer sampling. Subsequently, the preliminary reconstruction is performed to obtain $\mathbf{r_0}$ and the preliminary reconstruction $\mathbf{x'_0}$ using (11) or (9). Standard Gaussian noise $\mathbf{z}$ is then generated. $\mathbf{x_t}$ is calculated by (2) and concatenated with $\mathbf{x'_0}$ along the channel dimension as input to the UNet network to predict the noise $\tilde{\mathbf{z}}_\mathbf{t}$. Finally, the loss between the standard Gaussian noise $\mathbf{z}$ and the predicted noise $\tilde{\mathbf{z}}_\mathbf{t}$ is computed. The network parameters $\boldsymbol{\theta}$ are updated using gradient descent. These steps are repeated until the network converges. The whole process can be summarized as Algorithm 1.

The whole reconstruction process is summarized as Algorithm 2. After the network training is completed, sample a Gaussian noise as $\mathbf{x_T}$ from the standard Gaussian distribution. Then, according to different sampling methods, the measured value $\mathbf{y}$ is used to obtain $\mathbf{r_0}$ and the preliminary reconstructed image $\mathbf{x'_0}$. If $\mathbf{t} > \mathbf{1}$, a standard normal distribution noise sample $\mathbf{z}$ is generated. At each timestep, the noise $\tilde{\mathbf{z}}_\mathbf{t}$ is predicted by the UNet network, and $\mathbf{x_{t-1}}$ is calculated according to (3). Repeat the above process until $\mathbf{x_0}$ is reconstructed.

Algorithm 1: Training.

```
Input: The original image $\mathbf{x_0}$
repeat
  for $\mathbf{t = 0, 1, \ldots, T}$ do
    /*Reconstruct the image $\mathbf{x_0}$ with SIM and GDM*/
    if sampling by convolutional method then
      $\mathbf{y} \leftarrow \mathbf{F_{conv-sampling}(x_0)}$
      $\mathbf{r_0} \leftarrow \mathbf{F_{init}(y)}$
      $\mathbf{x'_0} = \mathbf{r_0} - \rho\boldsymbol{\Phi}\tilde{*}(\boldsymbol{\Phi} * \mathbf{r_0} - \mathbf{y})$
    else
      $\mathbf{y} \leftarrow \mathbf{F_{fc-sampling}(x_0)}$
      $\mathbf{r_0} \leftarrow \mathbf{F_{init}(y)}$
      $\mathbf{x'_0} = \mathbf{r_0} - \rho\boldsymbol{\Phi}^\mathbf{T}(\boldsymbol{\Phi}\mathbf{r_0} - \mathbf{y})$
    end if
    $\mathbf{z} \sim \mathbf{N(0, I)}$
    $\mathbf{x_t} = \sqrt{\bar{\alpha}_\mathbf{t}}\mathbf{x_0} + \sqrt{\mathbf{1} - \bar{\alpha}_\mathbf{t}}\mathbf{z}$
    the predicted noise $\tilde{\mathbf{z}}_\mathbf{t} \leftarrow \mathbf{F_{UNet}(x_t, x'_0)}$
    $\mathbf{Loss} \leftarrow \nabla_\theta ||\mathbf{z} - \tilde{\mathbf{z}}_\mathbf{t}||^\mathbf{2}$
    Update network parameters $\boldsymbol{\theta}$ using gradient descent
  end for
until network converges
```

IV. Experiments

In this section, we outline the implementation details of our designed network, and compare it with the existing excellent

Algorithm 2: Reconstruction.

```
Input: The measurement value $\mathbf{y}$
$\mathbf{x_T} \sim \mathbf{N}(\mathbf{0}, \mathbf{I})$
for $\mathbf{t} = \mathbf{T}, \ldots, \mathbf{1}$ do
    if sampling by convolutional method then
        $\mathbf{r_0} \leftarrow \mathbf{F_{init}}(\mathbf{y})$
        $\mathbf{x'_0} = \mathbf{r_0} - \rho\mathbf{\Phi}\tilde{*}(\mathbf{\Phi} * \mathbf{r_0} - \mathbf{y})$
    else
        $\mathbf{r_0} \leftarrow \mathbf{F_{init}}(\mathbf{y})$
        $\mathbf{x'_0} = \mathbf{r_0} - \rho\mathbf{\Phi^T}(\mathbf{\Phi}\mathbf{r_0} - \mathbf{y})$
    end if
    $\mathbf{z} \sim \mathbf{N}(\mathbf{0}, \mathbf{I})$ if $\mathbf{t} > \mathbf{1}$, else $\mathbf{z} = \mathbf{0}$
    the predicted noise $\tilde{\mathbf{z}}_\mathbf{t} \leftarrow \mathbf{F_{UNet}}(\mathbf{x_t}, \mathbf{x'_0})$
    $\mathbf{x_{t-1}} = \frac{\mathbf{1}}{\sqrt{\alpha_\mathbf{t}}}(\mathbf{x_t} - \frac{\mathbf{1}-\alpha_\mathbf{t}}{\sqrt{\mathbf{1}-\overline{\alpha}_\mathbf{t}}}\tilde{\mathbf{z}}_\mathbf{t}) + \sigma_\mathbf{t}\mathbf{z}$
end for
return The reconstruction image $\mathbf{x_0}$
```

methods. Additionally, we perform ablation experiments to analyze the contribution of each component.

A. Implementation Details

For network training, we randomly selected $\mathbf{15910}$ images from $\mathbf{202599}$ images in CelebA dataset, and cut them into $\mathbf{64 * 64}$ size as the training set. All networks are implemented in Pytorch. To make the convolution kernel size in convolutional sampling integer, our measurement rate is set to $\{\mathbf{25\%}, \mathbf{10\%}, \mathbf{6.25\%}, \mathbf{3.5\%}, \mathbf{1.6\%}\}$. As for the setting parameters, the batch size is $\mathbf{16}$, the epoch is $\mathbf{40}$, and the learning rate is $\mathbf{1 * 10^{-4}}$. We employ Adam optimizer [46] and Smooth L1 loss [47] to train the network. In GDM, as a learnable parameter, the step size $\boldsymbol{\rho}$ is initially set to $\mathbf{0.1}$ and is constantly updated in subsequent iterations. For DGRN, the total diffusion step $\mathbf{T}$ is set to $\mathbf{1000}$, and noise schedule $\beta_\mathbf{t}$ is set to increase linearly from $\mathbf{1e-4}$ to $\mathbf{2e-2}$ and then corrected by sigmoid function. $\beta_\mathbf{t}$ is a key parameter in the diffusion model to control the noise intensity added at each time step, and will be discussed in detail in Section IV-C3. In the test, we selected $\mathbf{16}$ pictures in CelebA dataset except $\mathbf{15910}$ for the training set as the test set. The reconstruction results are evaluated using two commonly used image assessment criteria: Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM).

B. Compare With State-of-the-Art Methods

In this section, we evaluate DGRN against the traditional optimization-based reconstruction algorithm TVAL3 [48] as well as four compressed sensing reconstruction algorithms based on deep learning: CS-Net [49], ISTA-Net+[11], MPIGAN [50] and DiffIR [31]. The PSNR and SSIM reconstruction performance in CelebA dataset is summarized in Table I. It can be seen from the results that our DGRN outperforms other competitive methods in terms of PSNR and SSIM in all cases. It is effectively proved that $\mathbf{x_0}$ and $\mathbf{x'_0}$ in diffusion models can guide the generation of images. The DGRN uses a diffusion model with generative features. Thanks to the "imaginative" ability of our method, DGRN can still generate clear images even at low measurement rates.

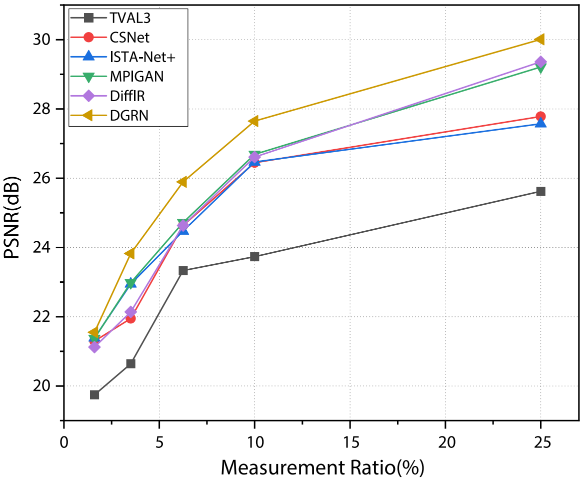

Fig. 6. Average PSNR on testing sets by different algorithms at different measurement rate.

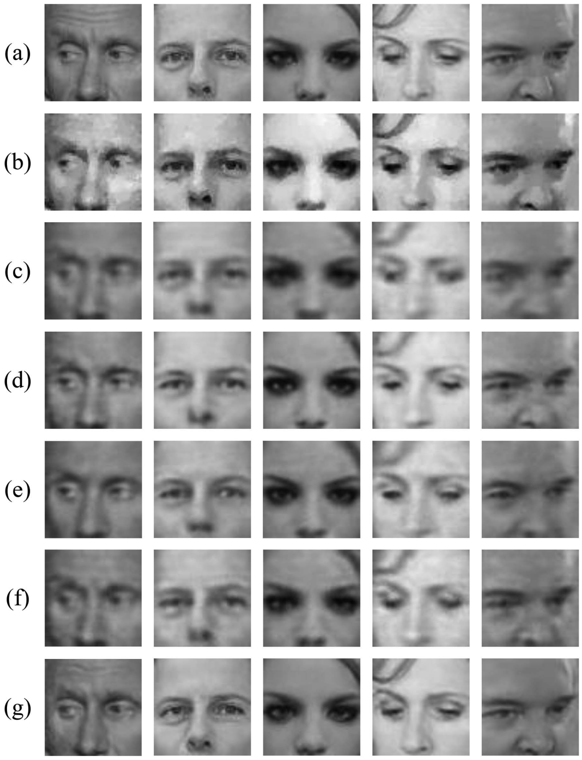

Fig. 7. Reconstruction results(MR = $\mathbf{0.1}$) by different algorithms. (a) Original images, (b) TVAL3, (c) CSNet, (d) ISTA-Net+, (e) MPIGAN, (f) DiffIR, (g) DGRN.

Fig. 6 shows the PSNR values for testing sets using different algorithms at various measurement rates. The samples generated by several models are shown in Fig. 7, indicating that the images generated by DGRN are more delicate and textured than other algorithms.

TABLE I
PSNR(dB) and SSIM of Different Algorithms Upon CelebA Dataset At Different Measurement Rate

Methods	MR=25%		MR=10%		MR=6.25%		MR=3.5%		MR=1.6%	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
TVAL3 [48]	25.620	0.8542	23.732	0.7277	23.334	0.6707	20.640	0.5882	19.745	0.5228
CSNet [49]	27.781	0.8591	26.445	0.8072	24.633	0.7814	21.948	0.6392	21.296	0.6080
ISTA-Net+ [11]	27.571	0.8654	26.468	0.8423	24.472	0.7823	22.944	0.7651	21.387	0.6274
MPIGAN [50]	29.211	0.8767	26.687	0.8150	24.717	0.7326	22.982	0.6487	21.375	0.5706
DiffIR [31]	29.359	0.8946	26.610	0.8186	24.634	0.7991	22.137	0.7067	21.122	0.6506
DGRN(Ours)	**30.009**	**0.9037**	**27.647**	**0.8672**	**25.892**	**0.8192**	**23.824**	**0.7739**	**21.551**	**0.7153**

TABLE II
PSNR(dB) and SSIM of Different Sampling Methods Upon CelebA Dataset At Different Measurement Rate

Sampling Methods	MR=25%		MR=10%		MR=6.25%		MR=3.5%		MR=1.6%	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
random gaussian matrix	27.288	0.8430	25.063	0.7872	25.407	0.7517	23.643	0.7294	20.886	0.6575
fully connected layer	28.222	0.8573	27.187	0.8294	25.883	0.7774	23.651	0.7510	21.094	0.6980
convolutional layer	**30.009**	**0.9037**	**27.647**	**0.8672**	**25.892**	**0.8192**	**23.824**	**0.7739**	**21.551**	**0.7153**

C. Ablation Studies

To gain further understanding, we used the same data to conduct ablation experiments from different perspectives at a measurement rate of 10%.

1) Impact of the Sampling Methods: We propose two sampling methods: fully connected layer sampling in Section III-B1 and convolutional sampling in Section III-B2 as part of the compressed sampling network. Fully connected sampling and convolutional sampling make the sampling matrix a learnable parameter, allowing them to be continuously optimized during the network training process, which can more effectively retain and extract important information in the image, overcome the shortcomings of traditional random matrix sampling, and significantly improve the reconstruction performance. We compare these two methods with random Gaussian matrix sampling while other network parameters are consistent. Table II shows the effects of three different sampling methods on image reconstruction performance. The results demonstrate that the use of learnable sampling matrix significantly improves the overall performance of the DGRN compared with the traditional Gaussian random matrix sampling method. By jointly optimizing sampling and generation, the network can capture image features more accurately, improve the reconstruction quality, and enhance the adaptability and robustness of the model.

The convolutional layer sampling achieves better results than the fully connected layer, especially at low sampling rates. As the network training progresses, convolutional layer sampling can continuously optimize the information extraction performance at low sampling rates. Additionally, convolutional layer sampling offers the benefits of parameter sharing and sparse connections, significantly reducing the number of weights in the sampling layer. At a measurement rate of $\mathbf{0.1}$, we calculate that the fully connected layer possesses $\mathbf{1677722}$ parameters as well as the parameters of the convolutional layer are $\mathbf{25600}$. The subsequent simulation experiments will use the network based on convolutional sampling.

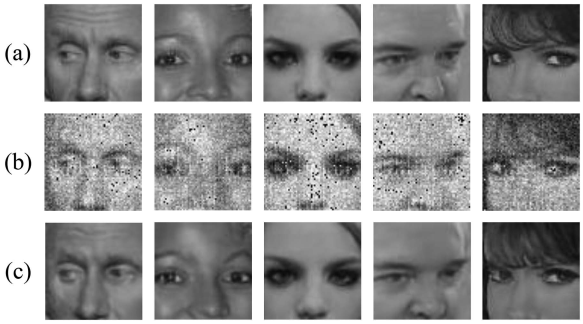

Fig. 8. Reconstruction results(MR $= \mathbf{0.1}$). (a) Original images, (b) Preliminary reconstruction, (c) reconstruction by DGRN.

2) Impact of the Gradient Descent Module: We use the gradient descent module in PGD in the preliminary reconstruction sub-network for low-quality image reconstruction. To explore the contribution of the gradient descent module to DGRN, we contrasted the performance of the network after removing the gradient descent module. Fig. 8 shows that the reconstructed images are generated under the guidance of the preliminary reconstructed image at a measurement rate of $\mathbf{0.1}$. Moreover, in Fig. 8(b) is the preliminary reconstructed image. The contour of the preliminary reconstructed image can be seen in the figure, which is consistent with the original image. It effectively proves that the preliminary reconstructed image with the gradient descent module can guide image generation in diffusion models. However, in the process of predicting noise, the preliminary reconstructed image $\mathbf{x_0'}$ is input into the UNet network together with the noise image $\mathbf{x_t}$. The network predicts the noise $\tilde{\mathbf{z}}_\mathbf{t}$ and

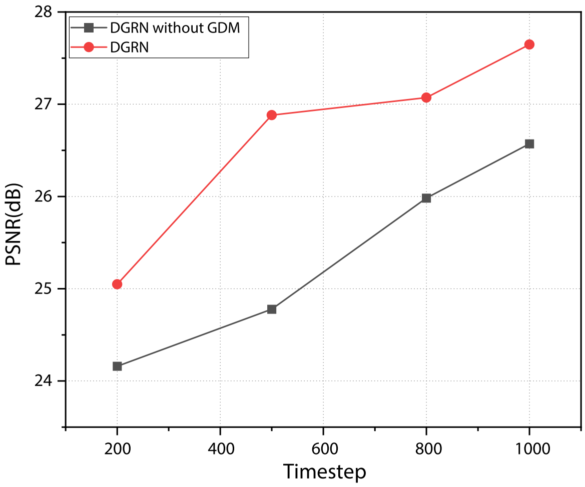

Fig. 9. Average PSNR on testing sets at different timesteps(MR=$\mathbf{0.1}$).

TABLE III
RESULTS OF ABLATION EXPERIMENTS IN THE CASE OF MEASUREMENT 10%

Experiment	Methods	PSNR(dB)	SSIM
	DGRN	**27.647**	**0.8672**
1	(a)	26.895	0.8306
2	(b)	25.178	0.8079
3	(c)	26.605	0.8326
4	(d)	25.167	0.7876
5	(e)	26.882	0.8472
6	(f)	26.305	0.8124

(a) DGRN with dim = (1,2,4), (b) DGRN with linear beta schedule, (c) DGRN with quadratic beta schedule, (d) DGRN with cosine beta schedule, (e) DGRN with timesteps = 500, (f) DGRN with timesteps = 2000.

trains by calculating the loss between the predicted noise $\tilde{\mathbf{z}}_\mathbf{t}$ and the random noise $\mathbf{z}$. Since this process does not directly calculate the loss between the original image $\mathbf{x_0}$ and the preliminary reconstructed image $\mathbf{x}'_\mathbf{0}$, the network may capture some noise characteristics, so some noise points will remain in the preliminary reconstructed image.

It can be seen from Fig. 9 that by integrating the gradient descent module into the reconstruction network, we can obtain higher-quality images with fewer iterations than without using the module. In addition, in the gradient descent module, the existence of the regularization term can make the model parameters smoother. This method of iteratively optimizing the parameters of the network helps to improve the anti-noise ability of the model and improve the quality of reconstruction.

3) Impact of the Network Parameters: To find the appropriate network parameters, we conducted ablation experiments from different perspectives. The effects of the dimension of UNet network layers, different variance schedule generation methods, and timesteps on the reconstruction performance were explored at a measurement rate of $\mathbf{10\%}$. In Table III, Experiment 1 is to modify the dimension of UNet network layers. When the dimension of layers is $\{\mathbf{1}, \mathbf{2}, \mathbf{4}\}$, the number of channels changed by the corresponding downsampling layer is $\{\mathbf{64}, \mathbf{128}, \mathbf{256}\}$. Experiment 2, Experiment 3 and Experiment 4 respectively used linear growth, cosine growth and quadratic function growth from $\mathbf{1e-4}$ to $\mathbf{2e-2}$ to generate variance schedule $\beta_\mathbf{t}$. The results show that the sigmoid growth method performs better in the quality of the generated image. Its smooth and nonlinear noise addition curve provides better noise distribution and a more stable training process, which helps the model to learn the distribution characteristics of data more effectively, to generate high-quality reconstructed images in the process of reverse denoising. In Experiment 5 and Experiment 6, the total timesteps $\mathbf{T}$ were set to $\mathbf{500}$ and $\mathbf{2000}$ respectively. The results show that the reconstruction effect is the best when the timestep is $\mathbf{1000}$. Although the step size of $\mathbf{2000}$ increases the computational complexity, it does not bring better reconstruction quality. This means that we can select the appropriate parameter configuration through experiments to find the best balance between reconstruction accuracy and computing resources. These experiments will guide further research and improvement of the proposed methods. We finally choose the dimension of UNet network layers $\{\mathbf{1}, \mathbf{2}, \mathbf{4}, \mathbf{8}\}$, sigmoid growth, and total timesteps of $\mathbf{1000}$ as the final setting of DGRN.

D. Application on Single Pixel Image

We have built a SPI system in the early stage [42]. To suit the hardware requirements, we developed a binary version of DGRN. When training the binary sampling matrix, the overall network architecture remains unchanged. Drawing inspiration from [51], we binarized our trainable measurement matrix using the sign function.

$$\mathrm{sign}(x) = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0 \end{cases} \tag{12}$$

However, the derivative of the sign function is almost zero everywhere, making the backpropagation unable to proceed smoothly. Thus in the backpropagation process, we use $\mathbf{Htanh}$ function to calculate the gradient.

When conducting actual experiments, the photodetector operates in counting mode, where each measurement is the photon count over equal time intervals. The photon counts over equal time intervals are proportional to the light intensity. We loaded the trained binary fully connected layer measurement matrix into the experimental device. The we normalized the measured values and input them into the trained network model, and successfully reconstructed the image. Fig. 10 illustrates the SPI reconstruction results $(\mathbf{64*64})$ of target Z and airplane compared with TVAL3 at five measurement rates$\{\mathbf{25\%}, \mathbf{10\%}, \mathbf{6.25\%}, \mathbf{3.5\%}, \mathbf{1.6\%}\}$. The target pattern is etched onto the mask plate, which must be a binary image. Only the patterned areas allow light to pass through. The results from our experiments effectively validate the results of our simulation experiments, and affirm the scientific soundness of our approach and the feasibility of its real-world application. In the future, we will strive to build imaging systems that support more complex

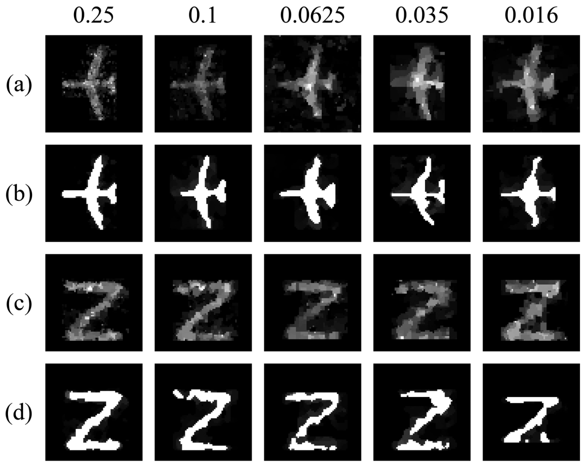

Fig. 10. Reconstruction results of sampled images in SPI system. (a) and (c) are the reconstruction results of TVAL3, (b) and (d) are the reconstruction results of DGRN.

scenes to further verify the applicability of our algorithms in complex scenes.

However, models with generative features will introduce a certain degree of distortion in reconstruction. Although the reconstruction result is clear at a low measurement rate, it is different from the original image, which involves the trade-off between clarity and authenticity. The traditional method has serious blurring and noise problems under extremely few measurements, which makes it difficult to effectively identify the target and cannot guarantee the authenticity of the target. Therefore, we believe that in very low measurement rate reconstruction, clarity should take precedence over authenticity. First, ensure the identifiability of the target, and then pursue higher authenticity. In the future, we will continue to study in this direction.

V. Conclusion

In this study, we propose a DM-based with Gradient Descent Module Guiding Reconstruction Network (DGRN) for SPI systems. To improve the performance of the single-pixel reconstruction network based on the diffusion model, we specifically designed the Sampling and Initialization Module(SIM) and the Gradient Descent Module(GDM). The outputs of SIM and GDM serve as conditions guiding the diffusion model. Experimental results show that our designed network outperforms existing compressed reconstruction networks. Ablation experiments demonstrate that the Gradient Descent Module effectively improves the quality of image reconstruction, enabling higher-quality images with fewer iterations. Additionally, it is proved that the learnable fully connected sampling and convolutional sampling overcome the defect of missing information in traditional random matrix sampling, and significantly improve the reconstruction performance especially at low sampling rates. By training the sampling matrix in the network into binary, our proposed network can be applied to SPI systems. The experimental results are better than TVAL3.

References

[1] Z. Zhang, X. Wang, G. Zheng, and J. Zhong, "Fast Fourier single-pixel imaging via binary illumination," *Sci. Rep.*, vol. 7, no. 1, 2017, Art. no. 12029.

[2] M.-J. Sun, L.-T. Meng, M. P. Edgar, M. J. Padgett, and N. Radwell, "A Russian dolls ordering of the hadamard basis for compressive single-pixel imaging," *Sci. Rep.*, vol. 7, no. 1, 2017, Art. no. 3464.

[3] Z.-H. Xu, W. Chen, J. Penuelas, M. Padgett, and M.-J. Sun, "1000 FPS computational ghost imaging using LED-based structured illumination," *Opt. Exp.*, vol. 26, no. 3, pp. 2427–2434, 2018.

[4] O. Katz, Y. Bromberg, and Y. Silberberg, "Compressive ghost imaging," *Appl. Phys. Lett.*, vol. 95, no. 13, 2009, Art. no. 131110.

[5] M. Lyu et al., "Deep-learning-based ghost imaging," *Sci. Rep.*, vol. 7, no. 1, 2017, Art. no. 17865.

[6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[7] S. Liu, X. Meng, Y. Yin, H. Wu, and W. Jiang, "Computational ghost imaging based on an untrained neural network," *Opt. Lasers Eng.*, vol. 147, 2021, Art. no. 106744.

[8] H. Wu et al., "Deep-learning denoising computational ghost imaging," *Opt. Lasers Eng.*, vol. 134, 2020, Art. no. 106183.

[9] K. Kulkarni, S. Lohit, P. Turaga, R. Kerviche, and A. Ashok, "ReconNet: Non-iterative reconstruction of images from compressively sensed measurements," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 449–458.

[10] W. Shi, F. Jiang, S. Zhang, and D. Zhao, "Deep networks for compressed image sensing," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2017, pp. 877–882.

[11] J. Zhang and B. Ghanem, "ISTA-net: Interpretable optimization-inspired deep network for image compressive sensing," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, 2018, pp. 1828–1837.

[12] T. Li, Q. Yan, Q. Zou, and Q. Dai, "Gates-controlled deep unfolding network for image compressed sensing," *IEEE Trans. Comput. Imag.*, vol. 10, pp. 103–114, 2024.

[13] I. Goodfellow et al., "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[14] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 6840–6851.

[15] X. Song et al., "High-resolution iterative reconstruction at extremely low sampling rate for Fourier single-pixel imaging via diffusion model," *Opt. Exp.*, vol. 32, no. 3, pp. 3138–3156, 2024.

[16] Q. Dai, Q. Yan, Q. Zou, Y. Li, and J. Yan, "Generative adversarial network with the discriminator using measurements as an auxiliary input for single-pixel imaging," *Opt. Commun.*, vol. 560, 2024, Art. no. 130485.

[17] S. Mao et al., "High-quality and high-diversity conditionally generative ghost imaging based on denoising diffusion probabilistic model," *Opt. Exp.*, vol. 31, no. 15, pp. 25104–25116, 2023.

[18] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang, "A tutorial on energy-based learning," *Predicting Structured Data*, vol. 1, 2006.

[19] J. Ngiam, Z. Chen, P. W. Koh, and A. Y. Ng, "Learning deep energy models," in *Proc. 28th Int. Conf. Mach. Learn.*, 2011, pp. 1105–1112.

[20] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE signal Process. Mag.*, vol. 35, no. 1, pp. 53–65, Jan. 2018.

[21] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1530–1538.

[22] I. Kobyzev, S. J. Prince, and M. A. Brubaker, "Normalizing flows: An introduction and review of current methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 3964–3979, Nov. 2021.

[23] D. P. Kingma et al., "An introduction to variational autoencoders," *Found. Trends Mach. Learn.*, vol. 12, no. 4, pp. 307–392, 2019.

[24] A. Oussidi and A. Elhassouny, "Deep generative models: Survey," in *Proc. IEEE Int. Conf. Intell. Syst. Comput. Vis.*, 2018, pp. 1–8.

[25] 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.

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

[27] H. Cao et al., "A survey on generative diffusion models," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 7, pp. 2814–2830, Jul. 2024.

[28] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," 2020, *arXiv:2010.02502*.

[29] P. Dhariwal and A. Nichol, “Diffusion models beat gans on image synthesis,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 8780–8794.

[30] B. Kawar, M. Elad, S. Ermon, and J. Song, “Denoising diffusion restoration models,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 23593–23606.

[31] B. Xia et al., “DiffiR: Efficient diffusion model for image restoration,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 13095–13105.

[32] J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans, “Cascaded diffusion models for high fidelity image generation,” *J. Mach. Learn. Res.*, vol. 23, no. 47, pp. 1–33, 2022.

[33] 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.

[34] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 10684–10695.

[35] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, “Image super-resolution via iterative refinement,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 4713–4726, Apr. 2023.

[36] C. Ledig et al., “Photo-realistic single image super-resolution using a generative adversarial network,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4681–4690.

[37] A. Bora, A. Jalal, E. Price, and A. G. Dimakis, “Compressed sensing using generative models,” in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 537–546.

[38] K. Hammernik et al., “Learning a variational network for reconstruction of accelerated MRI data,” *Magn. Reson. Med.*, vol. 79, no. 6, pp. 3055–3071, 2018.

[39] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[40] W. Dong, P. Wang, W. Yin, G. Shi, F. Wu, and X. Lu, “Denoising prior driven deep neural network for image restoration,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 10, pp. 2305–2318, Oct. 2019.

[41] S. Boyd et al., “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[42] Y. Guan, Q. Yan, S. Yang, B. Li, Q. Cao, and Z. Fang, “Single photon counting compressive imaging based on a sampling and reconstruction integrated deep network,” *Opt. Commun.*, vol. 459, 2020, Art. no. 124923.

[43] W. Shi et al., “Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1874–1883.

[44] Y. Wu and K. He, “Group normalization,” in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[45] A. Vaswani et al., “Attention is all you need,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.

[46] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, *arXiv:1412.6980*.

[47] R. Girshick, “Fast R-CNN,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.

[48] C. Li, *An Efficient Algorithm for Total Variation Regularization With Applications to the Single Pixel Camera and Compressive Sensing*. Houston, TX, USA: Rice Univ., 2010.

[49] W. Shi, F. Jiang, S. Liu, and D. Zhao, “Image compressed sensing using convolutional neural network,” *IEEE Trans. Image Process.*, vol. 29, pp. 375–388, 2020.

[50] S. Sun, Q. Yan, Y. Zheng, Z. Wei, J. Lin, and Y. Cai, “Single pixel imaging based on generative adversarial network optimized with multiple prior information,” *IEEE Photon. J.*, vol. 14, no. 4, Aug. 2022, Art. no. 8538110.

[51] J. Zhang, C. Zhao, and W. Gao, “Optimization-inspired compact deep compressive sensing,” *IEEE J. Sel. Topics Signal Process.*, vol. 14, no. 4, pp. 765–774, May 2020.