photonics

Article

A Compressed Reconstruction Network Combining Deep Image Prior and Autoencoding Priors for Single-Pixel Imaging

Jian Lin [1], Qiurong Yan [1,*], Shang Lu [2], Yongjian Zheng [1], Shida Sun [1] and Zhen Wei [1]

[1] School of Information Engineering, Nanchang University, Nanchang 330031, China; 6105119144@email.ncu.edu.cn (J.L.); 401030720018@email.ncu.edu.cn (Y.Z.); 6105119134@email.ncu.edu.cn (S.S.); 6105119141@email.ncu.edu.cn (Z.W.);

[2] School of Electronics and Information Engineering, Harbin Institute of Technology, Shenzhen 518055, China; 21s152102@stu.hit.edu.cn

* Correspondence: yanqiurong@ncu.edu.cn; Tel.: +86-0791-8396-9680

Abstract: Single-pixel imaging (SPI) is a promising imaging scheme based on compressive sensing. However, its application in high-resolution and real-time scenarios is a great challenge due to the long sampling and reconstruction required. The Deep Learning Compressed Network (DLCNet) can avoid the long-time iterative operation required by traditional reconstruction algorithms, and can achieve fast and high-quality reconstruction; hence, Deep-Learning-based SPI has attracted much attention. DLCNets learn prior distributions of real pictures from massive datasets, while the Deep Image Prior (DIP) uses a neural network's own structural prior to solve inverse problems without requiring a lot of training data. This paper proposes a compressed reconstruction network (DPAP) based on DIP for Single-pixel imaging. DPAP is designed as two learning stages, which enables DPAP to focus on statistical information of the image structure at different scales. In order to obtain prior information from the dataset, the measurement matrix is jointly optimized by a network and multiple autoencoders are trained as regularization terms to be added to the loss function. Extensive simulations and practical experiments demonstrate that the proposed network outperforms existing algorithms.

Keywords: single pixel imaging; compressed sensing theory; Deep Learning; deep image prior; autoencoder

Citation: Lin, J.; Yan, Q.; Lu, S.; Zheng, Y.; Sun, S.; Wei, Z. A Compressed Reconstruction Network Combining Deep Image Prior and Autoencoding Priors for Single-Pixel Imaging. *Photonics* **2022**, *9*, 343. https://doi.org/10.3390/photonics9050343

Received: 13 April 2022
Accepted: 11 May 2022
Published: 13 May 2022

1. Introduction

Single pixel imaging (SPI) is an imaging method based on compressive sensing and has become a research hotspot as a new imaging technology. Single-pixel imaging has two main advantages. One is that two-dimensional imaging can be achieved with a single-pixel detector without spatial resolution, so the cost is low, especially in special wavelengths such as infrared and terahertz. Second, the detector in the single-pixel system can collect the light intensity of multiple pixels at the same time, so that the signal-to-noise ratio is greatly improved. It has been widely used in medical imaging [1,2], radar [3–5], multispectral imaging [6,7], optical computing [8,9], optical encryption [10,11], etc.

Compared with multi-pixel imaging, single-pixel imaging is still very time-consuming, especially when performing high-resolution imaging, the sampling time and reconstruction time are very long, which limits its application in high-resolution and real-time scenarios. When the number of measurements is much smaller than the number of image pixels, it takes a lot of time for iterative operations to reconstruct the image by optimizing and solving the uncertainty problem. Traditional reconstruction algorithms include Orthogonal Matching Pursuit (OMP) [12], Gradient Projection for Sparse Reconstruction (GPSR) [13], Bayesian Compressive Sensing (BCS) [14], Total variation Augmented Lagrangian Alternating Direction Algorithm (TVAL3) [15], and so on. Deep Learning has achieved good results on computer vision tasks such as image classification, super-resolution, object detection, and restoration. Deep neural networks have been studied for the reconstruction of compressed measurement images. Compared with traditional iterative algorithms, Deep-Learning-based reconstruction methods effectively avoid huge computation and achieve fast and high reconstruction quality. Deep-Learning-based single-pixel imaging has attracted much attention. In 2017, Lyu et al. proposed a new computational ghost imaging (GI) framework based on Deep Learning [16]. In 2018, He et al. modified the commonly used convolutional neural network and then proposed a new ghost imaging method [17]. This method allows faster reconstruction of target images at low measurement rates. In the same year, Higham et al. achieved real-time, high-resolution video restoration using a deep convolutional autoencoder [18]. In 2019, Wang et al. developed a one-step end-to-end neural network that can directly use the measured bucket signals to recover target images [19]. In 2022, Wang et al. combined the physical model formed by GI images and deep neural networks to reconstruct far-field images at resolutions beyond the diffraction limit [20]. In 2020, Zhu et al. proposed a new ghost imaging scheme for dynamic decoding Deep-Learning framework, which greatly improves the sampling efficiency and the quality of image reconstruction [21]. We found that existing methods often require a large amount of training data to optimize the parameters of the network. However, in some fields, it is not easy to obtain enough data. The lack of a large amount of labeled data can easily lead to overfitting, resulting in poor reconstruction quality of the network. Second, the first layer weights of most networks are floating-point, limiting their application to single-pixel imaging. Therefore, we propose a compressed reconstruction network DPAP that combines Deep Image Prior (DIP) [22] and autoencoding priors. DIP uses a neural network's own structural prior to solve inverse problems without requiring a lot of training data. This makes up for the shortcomings of existing deep neural networks that rely on large datasets. Our main contributions are as follows:

1. We propose a compressed reconstruction network (DPAP) based on DIP for single-pixel imaging. DPAP is designed as two learning stages, which enables DPAP to focus on statistical information of the image structure at different scales. In order to obtain prior information from the dataset, the measurement matrix is jointly optimized by a network and multiple Autoencoders are trained as regularization terms to be added to the loss function.
2. We describe how DPAP optimizes network parameters with an optimized measurement matrix, enforcing network implicit priors. We also demonstrate by simulation that optimization of the measurement matrix can improve the network reconstruction accuracy.
3. Extensive simulations and practical experiments demonstrate that the proposed network outperforms existing algorithms. Using the binarized measurement matrix, our designed network can be directly used in single-pixel imaging systems, which we have verified by practical experiments.

2. Related Work and Background

2.1. Single Pixel Imaging System

Figure 1 shows the single pixel imaging system that we proposed previously. The parallel light source is provided by a LED (CreeQ5), parallel light tube collimators and attenuators (LOPF–25C–405). Under the illumination of parallel light, the target is imaged on the DMD (0.7XGA 12° DDR) by the imaging lens (OLBQ25.4–050). The DMD consists of 1024×768 micro-mirrors that can be rotated by ±12°, and the size of each micro-mirror is 13.68 um × 13.68 um. The binary measurement matrix is loaded on the DMD to module the image by FPGA (Altera DE2–115), and the micro-mirror whose corresponding element is 1 is flipped by +12°, and the micro-mirror whose corresponding element is 0 is flipped by −12°. In the +12° direction we place a lens (OLBQ25.4–050) to collect the light to the photon counter PMT (Hamamatsu H10682) to get the count value. If DMD becomes

N pixels through micro-mirror combination and A is a $M \times N$ binarized measurement matrix, the ith row of the measurement matrix A is loaded on the DMD, count value y_i can be obtained. After M modulations, the final result $y(y = [y_1, y_1, \dots, y_M])$ can be obtained. The whole process can be expressed as:

$$y = Ax + e \tag{1}$$

where e is the existing noise, and M/N is called the measurement rate (M is less than or equal to N). The image x is reconstructed from the measurement values y and the measurement matrix A.

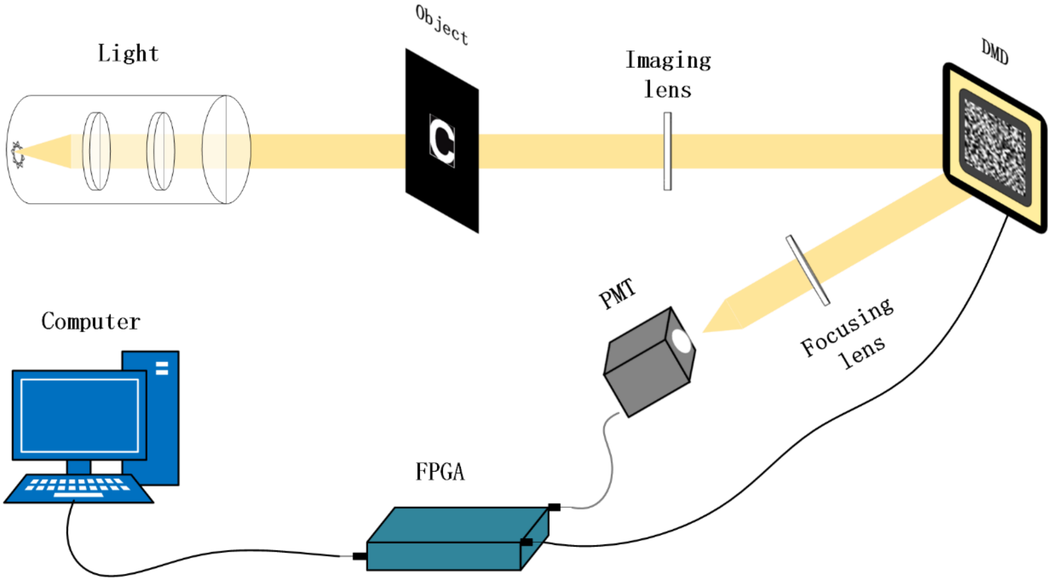

Figure 1. single pixel imaging system, DMD: Digital Micro-Mirror Device, PMT: photomultiplier Tube, FPGA: field programmable gate array.

2.2. Deep-Learning-based Compressed Sensing Reconstruction Network

Traditional compressive reconstruction algorithm: In single-pixel imaging, the number of measurements is much smaller than the number of image pixels, and the image must be reconstructed by solving the under-determined problem. Traditional compressive sensing reconstruction algorithms combine the prior knowledge of the scene to solve an under-determined problem. The prior knowledge includes sparsity prior, non-local low-rank regularization, and total variation regularization. There are three main methods to solve this under-determined problem: convex relaxation method, greedy matching pursuit method and Bayesian method. Algorithms such as OMP [12], GPSR [13], BCS [14], TVAL3 [15] mostly solve the reconstruction problem based on the assumption that images are sparse in the transform domain. GPSR [13] is a convex relaxation algorithm that converts a non-convex optimization problem based on l_0 norm to a convex optimization problem based on l_1 norm to solve. OMP [12] solves the minimum l_0 norm problem directly through a greedy algorithm. BCS [14] is a Bayesian method that transforms the reconstruction problem into a probabilistic solution problem by using the prior probability distribution of the signal. TVAL3 [15] combines the enhanced Lagrange function and the alternating minimization method based on the minimum total variance method. These traditional reconstruction methods are easy to understand and have theoretical guarantees. However, in these algorithms, even the fastest algorithms can hardly meet the requirement of real-time.

Deep-learning-based compressive reconstruction network: In recent years, compressed reconstruction networks based on Deep Learning have been widely used to solve image reconstruction problems. Instead of specifying prior knowledge, data-driven approaches have been explored to learn the signal characteristics implicitly. Mousavi et al.

stacked denoising autoencoders (DAEs) and then reversed the overall optimization [23], which improved compressed signal recovery performance and reduced the reconstruction time. Kulkarni et al. adopted the idea of blocking and proposed a ReconNet model based on image super-resolution reconstruction [24], which improved the accuracy of image compressed reconstruction. Yao et al. proposed a deep residual reconstruction network (DR2Net) [25], which introduced the ResNet structure on the basis of ReconNet to further improve the reconstruction quality of the compressed images. Inspired by generative adversarial network (GAN), Bora et al. proposed to use a pre-trained DCGAN for compressive reconstruction (CS-GM) [26]. Algorithms based on Deep Learning have higher running speed and reconstruction accuracy, but cannot explain the process of image reconstruction. Metzler et al. unroll the traditional iterative algorithm into a neural network, and use the training data to adjust the network parameters after unrolling. Through this hybrid approach, the corresponding prior knowledge can be learned from the training data while making the algorithm interpretable. LDAMP [27] unroll the DAMP [28] algorithm, and then using the algorithm parameters as the learned weights. It outperforms some advanced algorithms in both running time and accuracy.

2.3. Deep Image Prior

The deep image prior [22] is a method to solve the linear inverse problem using the neural network structure prior. DIP captures image statistical properties in an unsupervised way without any prior training.

$$\arg\min_{\theta}\|R_{\theta}(z)-x_0\|_2^2, \hat{x}=R_{\theta^*}(z) \tag{2}$$

where x_0 is the damaged image, $\hat{x}$ is the final recovered image and z is a fixed random vector. Different from the existing methods, DIP treats the neural network itself as a regularization tool. It implicitly exploits the regularization effect generated by the network in recovering damaged images. The optimized network can produce high-quality images. DIP does not rely on large datasets; hence, it has broad applications in areas where real data is difficult to obtain or data acquisition is expensive [29,30]. In recent years, some studies have proposed adding explicit priors to boost the DIP with better results [31,32].

2.4. Denoising Autoencoder Prior

Autoencoder prior has been shown to be incorporated into appropriate optimization methods to solve various inverse problems [33–35]. Bigdeli et al. defined the denoising autoencoder (DAE) prior [35], inspired by Alain [36]. Specifically, the DAE is trained using Equation 3, where D_{σ_η} denotes DAE and x is the input image.

$$L_{DAE}=E_{x,\eta}(\left\|x-D_{\sigma_\eta}(x+\eta)\right\|^2) \tag{3}$$

where the expectation is over all input images x and noise η with standard variance σ_η. The study by Alain et al. shows that the output of the optimal DAE is related to the true data density $p(l)$ [36].

$$D_{\sigma_\eta}(x)=\frac{\int g_{\sigma_\eta}(\eta)p(x-\eta)(x-\eta)d\eta}{\int g_{\sigma_\eta}(\eta)p(x-\eta)d\eta} \tag{4}$$

where g_{σ_η} is a Gaussian kernel with standard deviation σ_η. More importantly, the autoencoder error is proportional to the log-likelihood gradient of the smoothing density when the noise has a Gaussian distribution [36].

$$D_{\sigma_\eta}(x)-x=\sigma_\eta^2\nabla\log\,[g_{\sigma_\eta}*p](x) \tag{5}$$

The DAE error vanishes exactly at the stationary points of the true data distribution smoothed by the mean shift kernel [36]. Therefore, the DAE error can reasonably be used as a prior to solve various problems. Inspired by the above work, we use Equation 6 as a prior for our image reconstruction.

$$L_{DAE} = \|x - D(x + \eta)\|_2^2 \tag{6}$$

A novel prior based on the autoencoding network was added to our network and achieved more desirable results. DPAP is a plug-and-play model that can be plugged into any effective DAE. Furthermore, different priors can be provided by different denoisers (BM3D [37], FFDNet [38], etc.) [39].

3. Proposed Network

3.1. Network Architecture

Inspired by deep image prior, we proposed a compressive reconstruction network CSDIP, shown in Figure 2. It consists of a preliminary reconstruction network $F_f(\cdot)$ and a DIP reconstruction network $F_r(\cdot)$. The original image x is multiplied by a random measurement matrix A to obtain the input measurement value y. $F_f(\cdot)$ maps the measurement value y to a preliminary reconstructed image x^r with the same dimensions as the original image. x^r is input to $F_r(\cdot)$ to obtain prediction of the original image $\hat{x}$. $\hat{x}$ is multiplied by this measurement matrix A to obtain prediction of measurement value $\hat{y}$. We adjust the parameters of the network by minimizing the difference between y and $\hat{y}$. Update the network weights by gradient descent, so that the smaller the difference between y and $\hat{y}$, the closer the network output $\hat{x}$ is to x. Different from the existing compressive reconstruction network, CSDIP set the loss function in the measurement domain and only uses a single image to optimize network weights.

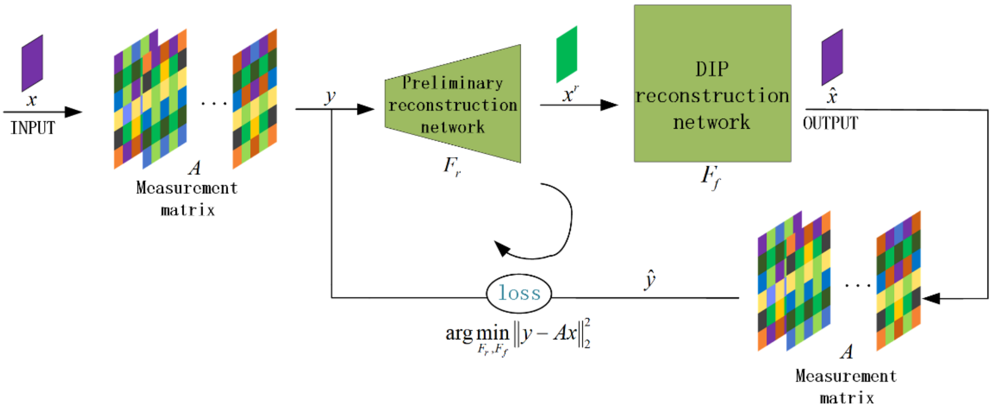

Figure 2. Structure of the proposed CSDIP.

On the basis of CSDIP, we propose DAPA as shown in Figure 3, which mainly makes two improvements. First, we designed a network that includes two fully connected layers to optimize the measurement matrix A by data learning. The weights of the first fully connected layer is used as measurement matrix A. The optimized measurement matrix A contains the prior information of the image data distribution and provides more underlying information of the image. This can speed up the reconstruction process and make the network more robust. Second, we add the DAE prior as a regularization term to the loss function. Three copies of $\hat{x}$ is used as three-channel fed into two DAEs with different levels of noise training. Here the DAEs are all trained in advance. The outputs of the two DAEs are averaged, respectively, to obtain single-channel images $\hat{x}_1$ and $\hat{x}_2$. We take the difference between the output of DPAP $\hat{x}$ and $\hat{x}_1$,$\hat{x}_2$ as the display regularization term.

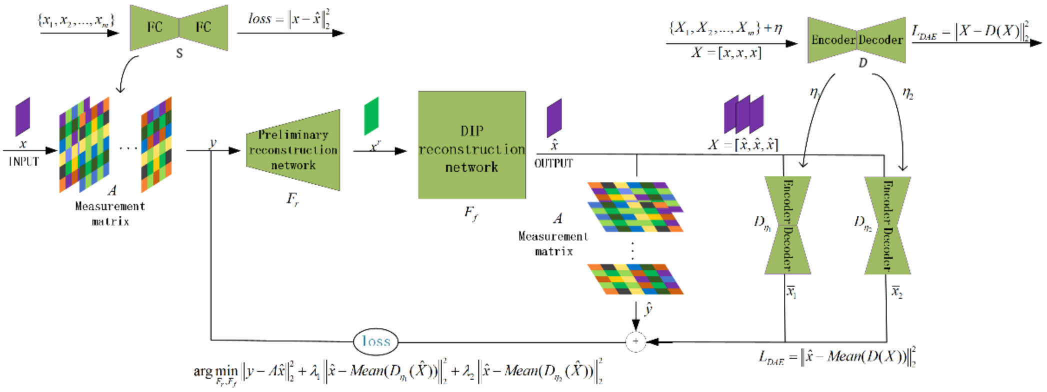
$\{x_1, x_2, ..., x_m\}$
FC
FC
S
$loss = |x - \hat{x}|_2^2$
$\{X_1, X_2, ..., X_m\} + \eta$
$X = [x, x, x]$
Encoder
Decoder
D
$L_{DAE} = |X - D(X)|_2^2$
η_1
η_2
x
INPUT
A
Measurement matrix
y
Preliminary reconstruction network
F_r
x^r
DIP reconstruction network
F_f
$\hat{x}$
OUTPUT
$X = [\hat{x}, \hat{x}, \hat{x}]$
A
Measurement matrix
D_{η_1}
Encoder Decoder
D_{η_2}
Encoder Decoder
$\bar{x}_1$
$\bar{x}_2$
$\hat{y}$
loss
$\arg\min_{F_r, F_f} \|y - A\hat{x}\|_2^2 + \lambda_1 \|\hat{x} - Mean(D_{\eta_1}(\hat{X}))\|_2^2 + \lambda_2 \|\hat{x} - Mean(D_{\eta_2}(\hat{X}))\|_2^2$
$L_{DAE} = \|\hat{x} - Mean(D(X))\|_2^2$

Figure 3. Structure of the proposed DPAP, where $Mean(\cdot)$ means to average the three-channel image.

Preliminary reconstruction network: The fully connected layer in the network is used to restore the low-dimensional measurements to the size of original image. This ensures that the subsequent reconstruction process is carried out smoothly.

DIP reconstruction network: The structure of the DIP reconstruction network is shown in Figure 4. The DIP reconstruction network consists of two U_net [40] sub-networks, that is $F_{r1}(\cdot)$ (first stage) and $F_{r2}(\cdot)$ (second stage).

The initial features are extracted by a 3*3 convolution and sent to subsequent decoders and encoders. Max pooling and deconvolution layers are used for upsampling and downsampling. The channels of features are doubled when downsampling, and the max pooling layer is shared by both sub-networks. $F_{r1}(\cdot)$ and $F_{r2}(\cdot)$ use different convolution modules to learn the features, where $F_{r1}(\cdot)$ uses the Edsr module [41] and $F_{r2}(\cdot)$ uses the HIN module [42]. In the first stage, we process the output of $F_{r1}(\cdot)$ through the supervised attention module (SAM) [43] to obtain an enhanced attention feature. This feature will be used as the residual feature required in the next stage. In the second stage, we first process the initially reconstructed image using 3×3 convolution (+ReLU), and then add the generated feature with the enhanced attention feature to obtain the multiscale feature. This multiscale feature is processed by two 3×3 convolutions (+ReLU) to obtain the $F_{r2}(\cdot)$ input. Finally, a 1×1 convolution is used in $F_{r2}(\cdot)$ to get the output of the DIP reconstruction network. The cross-stage feature fusion (CSFF) module [43] is used to fuse the two-stage features, which enriches the multiscale features of the second stage. The two sub-networks of the DIP reconstruction network can focus on information at different scales. This enables the network to focus on finer features, thus increasing the gain of the network.

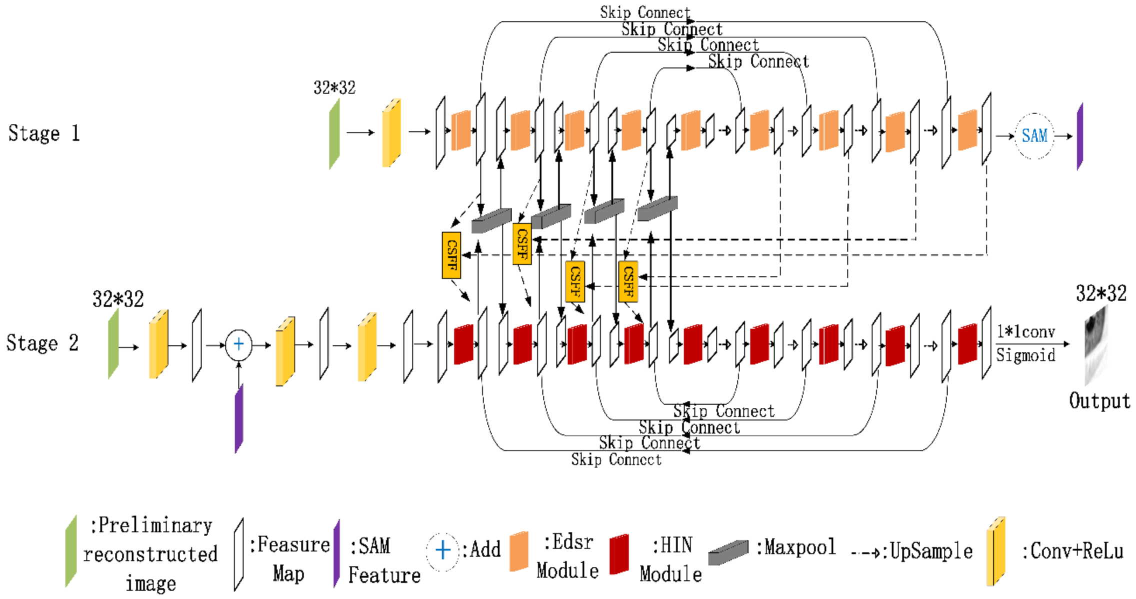

Figure 4. Structure of the DIP reconstruction network.

The structure of Edsr module and HIN module is shown in Figure 5. The Edsr module consists of two 3×3 convolutions, Dropout layers, and Selu functions. It preserves the residual structure of ResNet. Residual scaling is used because it greatly stabilizes and optimizes the training process. We use a Dropout layer after the convolution of both HIN module and Edsr module, which is also to stabilize the optimization process. The HIN module divides the input features extracted by the first convolution into two parts by channel, and subsequently performs a normalization operation on one part alone, leaving the other part unprocessed, and finally merges the two parts. The merged intermediate features are processed by 3×3 convolution to obtain the residual features. The branch feature is obtained after a 1×1 convolution of the input feature. HIN module output by add the residual features with branching features. Dropout layers are not added to the first and last two modules used in $F_{r1}(\cdot)$ and $F_{r2}(\cdot)$.

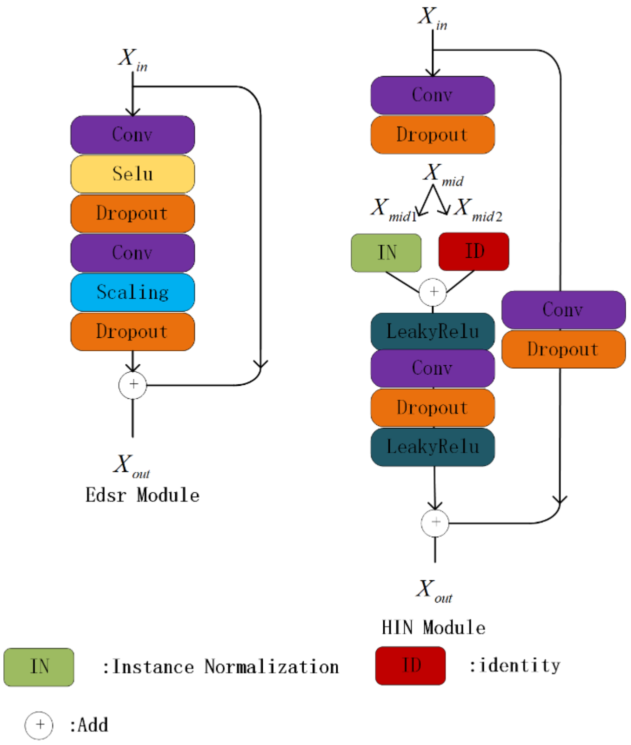

Figure 5. Structure of Edsr Module and HIN Module.

Denoising autoencoder: The structure of the DAE that we used is given in Figure 6. DAEs use noisy data to learn how to efficiently recover the original input. The DAE consists of an encoder and a decoder that generates the reconfiguration. Max pooling and deconvolution layers are used for upsampling and downsampling. The convolutional layer before upsampling is followed by its rectified linear units (ReLU). The size of the convolution kernel is 3×3. In the encoder, the number of channels is 64, 64 and 32, respectively. In the decoder, the number of channels is 32, 32 and 64, respectively. Both the input and output of the DAE are 3 channels. This under-complete learning can force the DAE to learn the most salient features in the training data, helping the network to restore the most useful features of the input signal.

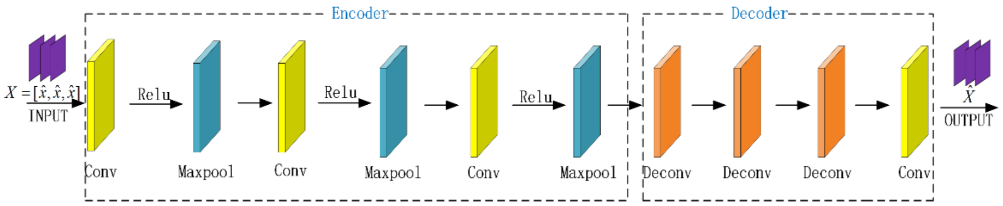

Figure 6. The structure of the denoising autoencoder.

3.2. Loss Function/Regularization Term Design

The input of DPAP is a single $32 \times 32 \times 1$ image. We take the difference between the two measurements as the loss function fidelity term. ϕ_θ is a set of weights and biases for the initial reconstruction network and the DIP reconstruction network during optimization.

$$y = Ax \tag{7}$$

$$\arg\min_{\theta} \|A\phi_\theta(y) - y\|_2^2 \quad \hat{x} = \phi_{\theta^*}(y) \tag{8}$$

The DAE error is used as a display regularity term to accelerate DPAP to produce good quality images.

$$L_{DAE} = \|\hat{x} - D(\hat{x})\|_2^2 \tag{9}$$

To make our DAE more efficient, we first make 3 copies of the single-channel image, and then concatenate them in the last dimension to get a three-channel image. We train the DAE using three-channel images.

$$X = [x, x, x] \tag{10}$$

Benefiting from the three-channel strategy, more similar detailed features are learned by the DAE. We average the output of the DAE, still using the difference between the single-channel images as the regularization term. Noise level is a very important parameter. Too high or too low noise level will make the network unable to converge to the global optimal result. The network wants to trade more noise for more texture details when the noise level is set lower, and it prefers to create smoother results when the noise level is set higher [44]. If we only add a single noise during training, although more detailed features such as edge texture can be preserved, the generalization ability of the network is poor. Therefore, two DAEs trained with different noise levels work in parallel in DPAP, which helps to enhance the generalization performance of the network and pay attention to different levels of priors. The DAE prior can be expressed as:

$$L_{DAE} = \left\|\hat{x} - Mean(D_{\eta_1}(\hat{X}))\right\|_2^2 + \left\|\hat{x} - Mean(D_{\eta_2}(\hat{X}))\right\|_2^2 \tag{11}$$

We add a regularization factor before the regularization term to control the balance between the regularization term and the fidelity term. The loss function of DPAP can be expressed as:

$$\begin{aligned} \hat{x} = \arg\min_{\theta,\hat{x}} \|A\phi_\theta(y) - y\|_2^2 + \lambda_1 \left\|\hat{x} - Mean(D_{\eta_1}(\hat{X}))\right\|_2^2 \\ + \lambda_2 \left\|\hat{x} - Mean(D_{\eta_2}(\hat{X}))\right\|_2^2 \end{aligned} \tag{12}$$

3.3. Training Method

A typical Deep-learning method is to learn a feasible mapping function from a large number of data pairs(y_i, x_i).

$$\theta^* = \arg\min_{\theta} \|\phi_\theta(y_i) - x_i)\|_2^2 \tag{13}$$

where ϕ_θ is obtained by random initialization. However, this case is only suitable when a large amount of data is available and the test images are similar to the training set. Our scheme can compute a feasible solution by minimizing the measurement domain loss function without relying on the object data. We adjust the parameters of the network by minimizing the loss function in the measurement domain and update the weights by gradient descent. The smaller the difference between y and $\hat{y}$, the closer the network output $\hat{x}$ is to x. It can be expressed as:

$$\begin{aligned} \theta^* = \arg\min_{\theta} \|A\phi_\theta(y) - \mathrm{y})\|_2^2 + \lambda_1 \left\|\hat{x} - Mean(D_{\eta_1}(\hat{X}))\right\|_2^2 \\ + \lambda_2 \left\|\hat{x} - Mean(D_{\eta_2}(\hat{X}))\right\|_2^2, \hat{x} = \phi_{\theta^*}(y) \end{aligned} \tag{14}$$

The network structure itself can capture rich low-level image statistical priors for image reconstruction [45]. The mapping function ϕ_{θ^*} obtained after the optimization of the network parameters can be directly used to restore the image $\hat{x}(\hat{x} = \phi_{\theta^*}(y))$. We use the Adam optimizer to iterate 20,000 times and update the reconstructed image every 100 iterations. We use the same set of 91 images used in [46] to generate the training data for the DAEs and measurement matrices. Subsequently, 32 × 32 pixel blocks are extracted from these images with a stride of 14. This process results in a total of 21,760 pixel blocks, which are used as input to the network. The images generated by DPAP are of poor quality in the early stages of iteration. A very high error value will be obtained if poor quality images are used as input to the pre-trained DAE. Therefore, if the DAE error is introduced into the network as a regularization term at the early stage of training, the overall loss of the network will be difficult to converge because the regularization term is too high, which is obviously a result we do not want to see. We iterate the network before adding the DAE until DAE error is similar to the measurement error, which can effectively avoid this problem. The regularization factor remains constant during iterations. The detailed training method has been given in Algorithm1.

Algorithm 1 DPAP algorithm

```
Autoencoder training:
    Initialize encoder weight $D_e$ and decoder weight $D_d$.
    For number of training iterations undertake the following:
        Input batches of data $(x_i)_i^N$,
        For all i, Copy the input image $(x_i)_i^N$ as three-channels:
                    $(X_i)_i^N = (x_i, x_i, x_i)_i^N$
        Generate Gaussian random noise:
                    $[dim_1, dim_2] = the\ size\ of\ x_i$
                    $\eta_i = [dim_1, dim_2, 3]$
        Add Gaussian noise: $(X_{i\eta})_i^N = (X_i)_i^N + \eta_i$,
        Compute coded value: $Y_i = D_e X_{i\eta}$
        Decode the coded value: $\hat{X} = D_d Y_i$
        Updata the $D_e$ and $D_d$ to minimize the reconstruction error:
                    $\arg\min_{D_e,D_d} \left\|X_i - D_e D_d X_{i\eta}\right\|_2^2$
    end for
    Get $D_e$ and $D_d$.
Measurement matrix training:
    Initializes the weights of the two fully connected layers: $W_1, W_2$.
    For number of training iterations undertake the following:
        Input batches of data $(x_i)_i^N$,
        For all i, compute the fully connected layer rebuild value: $\hat{x} = W_1 W_2 x_i$
        Updata the $W_1\ and\ W_2$ to minimize the reconstruction error:
                    $\arg\min_{W_1,W_2} \|x_i - W_1 W_2 x_i\|_2^2$
    end for
    Get $W_1$ and $W_2$.
DPAP testing:
    Initialize the weight of preliminary reconstruction network $F_f(\cdot)$ and DIP reconstruction network $F_r(\cdot)$.
    Restore the fully connected layer weight $W_1$ as the measurement matrix $A$.
    Restore the Autoencoder weight: $D_{e1}$,$D_{d1}$ and $D_{e2}$,$D_{d2}$.
    For number of training iterations undertake the following:
        Input: just one image $x$
        Compute measurement value: $y = Ax$
        Compute the value of preliminary reconstruction network and DIP reconstruction network:
                    $x^r = F_f(y)$
                    $\hat{x} = F_r(x^r)$
        Compute the measurement value of the reconstruction:
                    $\hat{y} = A\hat{x}$
        Copy the input image $\hat{x}$ as three-channels: $\hat{X} = [\hat{x}, \hat{x}, \hat{x}]$
        Compute Autoencoder error:
            $L_{DAE} = \lambda_1\left\|\hat{x} - Mean(D_{e1}D_{d1}\hat{X})\right\|_2^2 + \lambda_2\left\|\hat{x} - Mean(D_{e2}D_{d2}\hat{X})\right\|_2^2$
        Compute measurement error:
            $L_{measurement} = \|y - \hat{y}\|_2^2$
        If the Autoencoder error and measurement error are on the same order of magnitude:
            Updata the preliminary reconstruction network and DIP reconstruction network to minimize reconstruction error:
                $\arg\min_{F_f,F_r}\|y - \hat{y}\|_2^2 + \lambda_1\left\|\hat{x} - Mean(D_{e1}D_{d1}\hat{X})\right\|_2^2 + \lambda_2\left\|\hat{x} - Mean(D_{e2}D_{d2}\hat{X})\right\|_2^2$
        else:
            Updata the preliminary reconstruction network and DIP reconstruction network to minimize measurement error:
                $\arg\min_{F_f,F_r}\|y - \hat{y}\|_2^2$
    end for
    Return $\hat{x}$.
```

The measurement matrix needs to be binarized when our network is applied to a SPI system. We binarize the first fully connected layer weights with the sign function.

$$A^{\mathrm{b}} = \mathrm{sign}(A)\begin{cases} +1, A \geq 0 \\ -1, \text{otherwise} \end{cases} \tag{17}$$

where A is a floating-point weight, and A^b is a binary weight. However, the derivative of the sign function is almost 0 everywhere, which makes the backpropagation process unable to proceed smoothly. Inspired by the binarized neural network [47], we use the tanh function to replace the sign function during backpropagation to rewrite the gradient.

$$\mathrm{Htanh}(x) = \mathrm{Clip}(x, -1{,}1) = \max[x - 1, \min(1, x)] \tag{18}$$

when the network optimization is complete, the weight matrix of the first fully connected layer is the binary measurement matrix we need.

4. Results and Discussion

In this section, a series of simulation experiments will verify the reconstruction performance of our network and use the peak signal-to-noise ratio (PSNR) to evaluate the quality of reconstruction.

4.1. DIP Reconstruction Network Performance Verification Experiment

We compare the reconstruction effect of DPAP using different DIP reconstruction networks. We trained U_Net, Edsr [41], Wdsr [48] and our DIP reconstruction network (ODRN) using Algorithm 1, separately. To ensure fairness and simulate a real environment, we did not use DAEs and added uniform random noise with a noise level of 0.1. Table 1 and Figure 7 show the imaging results of these four networks. The reconstruction quality of our DIP reconstruction network is much better than U_Net, Edsr and Wdsr. This also proves that the stage learning network has advantages in single image reconstruction. The first stage of DPAP learns the residual features accurately and the second stage adds details quickly, which ensures that DPAP reconstructs images with high quality.

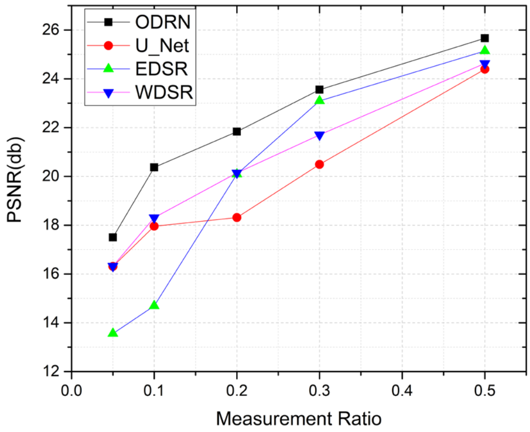

Figure 7. PSNR values of DPAP imaging results when using different DIP reconstruction networks.

Table 1. PSNR of reconstruction results at different measurement rates when DPAP uses different DIP reconstruction networks.

Image Name	Methods	MR = 0.05	MR = 0.1	MR = 0.2	MR = 0.3	MR = 0.5
eye	U_net	17.577	22.079	17.344	23.273	26.354
	Edsr	13.494	15.140	21.459	26.885	**28.414**
	Wdsr	18.615	19.830	20.896	23.190	26.333
	ODRN	**21.087**	**23.990**	**27.102**	**27.425**	28.314
butterfly	U_net	**16.494**	17.177	18.141	18.344	**27.669**
	Edsr	12.816	10.344	16.789	18.926	21.488
	Wdsr	14.979	16.872	17.060	19.690	23.491
	ODRN	15.898	**18.808**	**19.211**	**21.296**	23.943
parrot	U_net	19.078	17.562	19.401	21.050	24.863
	Edsr	13.530	14.666	19.204	**25.923**	26.244
	Wdsr	17.833	18.967	21.253	21.360	22.482
	ODRN	**20.552**	**23.373**	**24.736**	25.083	**27.107**
babara	U_net	12.127	14.513	17.617	19.673	21.152
	Edsr	12.276	14.862	**21.703**	20.159	25.351
	Wdsr	12.490	14.910	19.635	**22.939**	**25.935**
	ODRN	**12.957**	**16.011**	17.709	22.087	24.623
hat	U_net	16.259	18.456	19.077	20.132	21.933
	Edsr	15.635	18.436	21.311	**23.585**	24.206
	Wdsr	**17.695**	**20.970**	**21.876**	21.340	**24.914**
	ODRN	16.982	19.671	20.410	21.882	24.336
Mean PSNR	U_net	16.307	17.958	18.316	20.494	24.394
	Edsr	13.559	14.690	20.093	23.096	25.141
	Wdsr	16.322	18.310	20.144	21.705	24.631
	ODRN	**17.495**	**20.370**	**21.834**	**23.555**	**25.664**

The combination of DIP and the sampling process reduces the need for high-quality noise-free images. As shown in Figure 8, our DIP reconstruction network can recover sharper detailed textures and boundary contours than U_Net, Edsr and Wdsr in the case of noise pollution. Figure 9 shows some images generated by DPAP during testing.

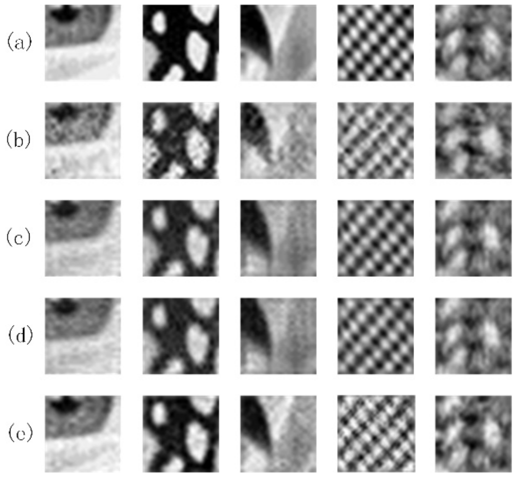

Figure 8. Imaging results of DAPA using different DIP reconstruction networks at 0.2 measurement rate, from top to bottom are (**a**) original image; (**b**) reconstructed by U_Net; (**c**) reconstructed by Edsr; (**d**) reconstructed by Wdsr; (**e**) reconstructed by our DIP reconstruction network.

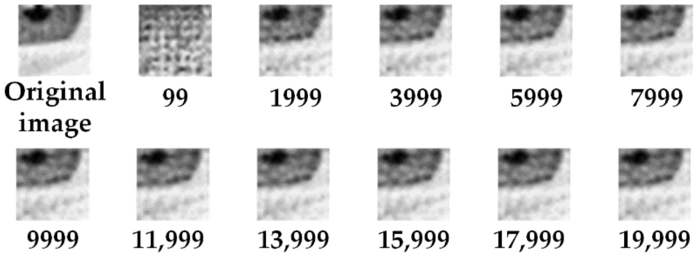

Figure 9. Some images generated by DPAP with a measurement rate of 0.1 at different number of iterations.

4.2. DPAP Performance Evaluation after Optimizing Measurement Matrix

To verify that optimizing the measurement matrix resulted in higher gains for the network, we also selected the random Gaussian matrix as the measurement matrix for comparison. For each measurement rate, a random Gaussian matrix with a standard variance of 0.01 was generated as the measurement matrix. The experimental results are shown in Table 2 and Figure 10. Our handcrafted measurement matrix performs much better than the Gaussian random matrix. Especially at 0.2 measurement rate, our handcrafted measurement matrix outperforms the random Gaussian matrix by 5.7 dB in PSNR. Optimizing the measurement matrix by data learning is a very effective method. Compared to random Gaussian matrices, our handcrafted measurement matrices impose stricter constraints on DPAP, forcing DPAP to perform the implicit prior.

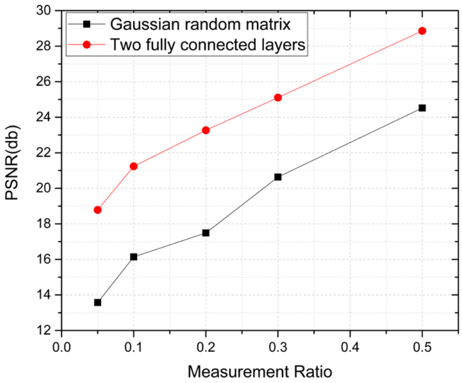

Figure 10. PSNR values of imaging results when DPAP uses different measurement matrices.

Table 2. PSNR values of DPAP imaging results using different measurement matrices at different measurement rates.

Image Name	Methods	MR = 0.05	MR = 0.1	MR = 0.2	MR = 0.3	MR = 0.5
eye	Gaussian	12.688	20.863	23.179	24.167	28.140
	FC	**23.307**	**25.736**	**27.499**	**29.216**	**32.837**
butterfly	Gaussian	14.841	10.935	17.025	18.745	24.315
	FC	**17.942**	**20.243**	**23.064**	**22.581**	**26.530**
parrot	Gaussian	14.377	17.605	18.836	22.790	26.103
	FC	**21.650**	**23.779**	**25.715**	**29.424**	**32.727**
babara	Gaussian	12.079	14.637	12.737	20.124	22.504
	FC	**12.825**	**16.057**	**17.227**	**21.985**	**26.837**
hat	Gaussian	13.905	16.697	15.671	17.342	21.545
	FC	**18.206**	**20.383**	**22.820**	**22.066**	**25.365**
Mean PSNR	Gaussian	13.574	16.142	17.489	20.635	24.522
	FC	**18.786**	**21.240**	**23.265**	**25.106**	**28.859**

FC: a measurement matrix trained with two fully connected layers; Gaussian: a random Gaussian matrix as a measurement matrix.

4.3. An Exploratory Experiment on Denoising Autoencoder Priors

Different form most algorithms that rely on Alternating Direction Method Multipliers (ADMM) [49], we avoid the complicated derivation of the denoising function in back-propagation. We compared DPAP without DAE, DPAP with a single-channel DAE and DPAP with a three-channel DAE. Single-channel DAE means the DAE is trained with single-channel images, and three-channel DAE means the DAE is trained with three-channel images. Four regularization coefficients 0.0011, 0.0012, 0.0013, 0.0014 are selected according to the actual situation. DAE errors are added as regularization terms when DPAP iterates 3000 times. This is to prevent the network from converging because the regularization term is too high. Figure 11 and Table 3 show the best results for DPAP among these four regularization coefficients.

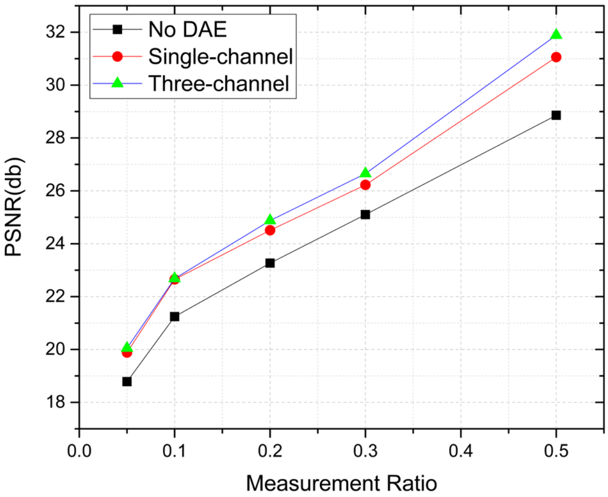

Figure 11. PSNR values of test pictures at different measurement rates when DPAP uses different channel denoising autoencoders.

Table 3. PSNR of 5 test images when DPAP uses different channel denoising autoencoders at different measurement rates.

Image Name	Methods	MR = 0.05	MR = 0.1	MR = 0.2	MR = 0.3	MR = 0.5
eye	No DAE	23.307	25.736	27.499	29.216	32.837
	Single-channel	24.571	27.357	29.372	**31.450**	34.720
	Three-channel	**25.183**	**27.789**	**29.748**	30.682	**34.793**
butterfly	No DAE	17.942	20.243	23.064	22.581	26.530
	Single-channel	18.734	20.282	22.832	23.578	27.772
	Three-channel	**19.051**	**20.393**	**24.141**	**24.534**	**28.605**
parrot	No DAE	21.650	23.779	25.715	29.424	32.727
	Single-channel	23.820	**27.377**	**28.256**	30.104	36.140
	Three-channel	**24.389**	27.158	27.300	**30.633**	**37.639**
babara	No DAE	12.825	16.057	17.227	21.985	26.837
	Single-channel	**13.344**	16.614	18.837	22.401	28.920
	Three-channel	12.661	**16.938**	**19.944**	**24.441**	**29.377**
hat	No DAE	18.206	20.383	22.820	22.066	25.365
	Single-channel	18.963	**21.620**	23.251	23.604	28.653
	Three-channel	**18.983**	21.142	**23.266**	**26.290**	**29.026**
Mean PSNR	No DAE	18.786	21.240	23.265	25.106	28.859
	Single-channel	19.886	22.650	24.510	26.228	31.058
	Three-channel	**20.053**	**22.684**	**24.880**	**26.653**	**31.888**

The performance of the network has been greatly improved after using DAE. Figure 12 shows images reconstructed by DPAP using DAEs trained with different channels images. The underlying data density distribution learned by the DAE helps DPAP to recover more fine textures. The DAE with three-channel image training achieves better performance than single-channel image, and the texture details of the reconstructed images are more completely preserved. This also preliminarily confirms that the DAE error can be used as a very effective prior to solve the problem of image reconstruction, and the high-dimensional training data can further enrich the prior information.

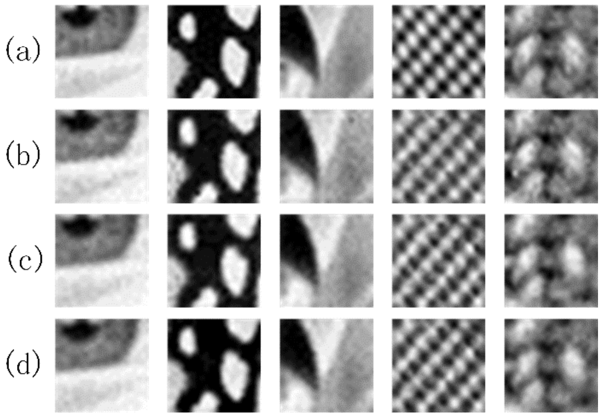

Figure 12. DPAP imaging results using different channel denoising autoencoders at a measurement rate of 0.2, from top to bottom are (**a**) original images; (**b**) reconstructions without denoising autoencoders; (**c**) a single-channel denoising autoencoder; (**d**) a three-channel denoising autoencoder.

4.4. Comparison of DPAP with Other Existing Networks

We also compare our reconstruction algorithm with the TVAL3 algorithm and two Deep-Learning-based algorithms, ReconNet, DR2Net. To ensure fairness, we add the same measurement matrix in front of ReconNet and DR2Net. The training data of the DAE is used to train ReconNet and DR2Net, with a total of 21,760 training samples of size 32*32.

ReconNet and DR2Net use the Adam optimizer for 2000 iterations, while the autoencoding network and the network for optimization of the measurement matrix iterates only 500 times. The main network of DPAP does not need to be trained, and the parameters of the network are optimized through an iterative strategy. Figure 13 and Table 4 give the PSNR of the five reconstructed images. Even compared to existing algorithms trained with large amounts of data, our algorithm still has significant advantages. At all measurement rates, DPAP demonstrates its powerful prior modeling capabilities. The reconstruction results are shown in Figure 14. In simulation, our network achieves better loss-reduction levels (experiments show that this level is maintained around 1e-5 order of magnitude) and shorter convergence times than other networks under the same conditions. This also confirms the feasibility of our proposed scheme. Figure 15 shows the reconstruction error curves of DPAP, ReconNet and DR2Net at a sampling rate of 0.05. We ensure that ReconNet and DR2Net are already in a state of convergence.

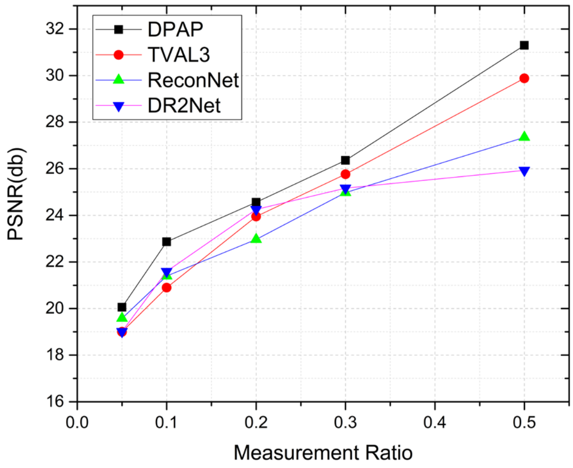

Figure 13. PSNR of test images with different algorithms at different measurement rates.

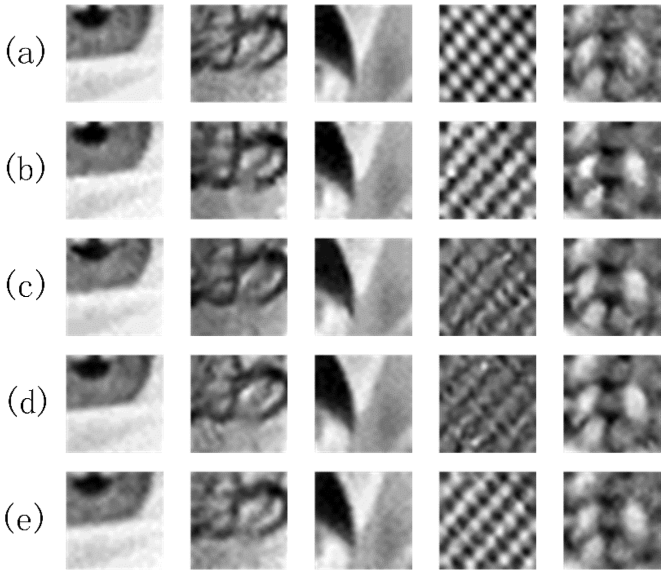

Figure 14. Reconstructed images of 5 test images under different algorithms at a measurement rate of 0.2, from top to bottom are (**a**) original images; (**b**) reconstructed by TVAL3; (**c**) reconstructed by ReconNet; (**d**) reconstructed by DR2Net; (**e**) reconstructed by DPAP.

Table 4. Reconstructed PSNR of 5 test images using different networks at different measurement rates.

Image Name	Methods	MR = 0.05	MR = 0.1	MR = 0.2	MR = 0.3	MR = 0.5
eye	TVAL3	24.214	25.419	29.208	31.255	**34.822**
	ReconNet	24.046	22.534	28.857	28.556	31.078
	DR2Net	21.999	**27.876**	29.522	30.799	30.404
	DPAP	**25.183**	27.789	**29.748**	**31.682**	34.793
hair	TVAL3	17.404	21.115	22.268	**24.137**	**26.881**
	ReconNet	19.018	20.897	19.443	23.745	25.126
	DR2Net	18.533	**21.479**	22.367	18.735	24.301
	DPAP	**19.043**	21.305	**22.555**	23.049	25.651
parrot	TVAL3	23.284	23.030	27.597	29.275	34.395
	ReconNet	22.736	26.245	**27.647**	16.051	18.995
	DR2Net	22.591	22.188	31.648	**33.288**	31.704
	DPAP	**24.389**	**27.158**	27.300	30.633	**37.639**
babara	TVAL3	12.585	16.159	18.865	21.274	26.366
	ReconNet	12.737	16.261	16.501	18.955	24.574
	DR2Net	**13.012**	15.798	14.681	19.360	19.726
	DPAP	12.661	**16.938**	**19.944**	**23.777**	**29.377**
hat	TVAL3	17.512	18.754	21.815	22.878	**26.960**
	ReconNet	**19.362**	21.045	22.839	**24.133**	25.811
	DR2Net	18.913	20.605	23.048	23.654	23.530
	DPAP	18.983	**21.142**	**23.266**	23.638	29.026
Mean PSNR	TVAL3	19.000	20.895	23.951	25.764	29.885
	ReconNet	19.580	21.397	22.967	24.977	27.346
	DR2Net	19.014	21.593	24.253	25.167	25.933
	DPAP	**20.052**	**22.866**	**24.563**	**26.356**	**31.297**

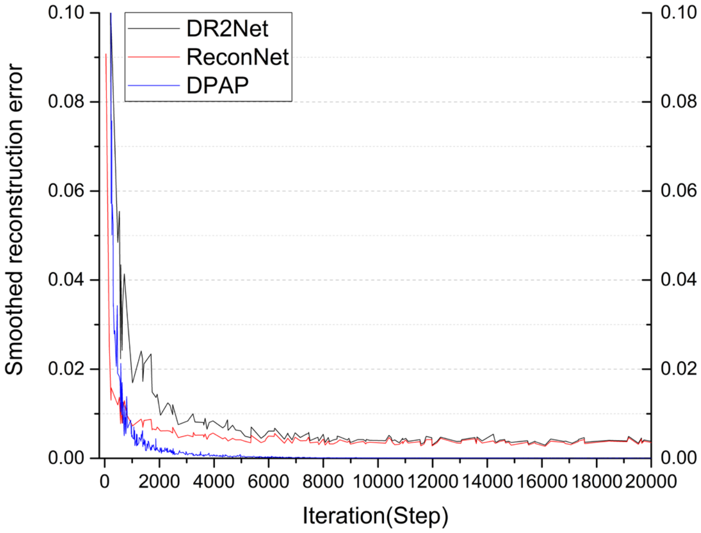

Figure 15. Loss curves of ReconNet, DR2Net and DPAP at a measurement rate of 0.05.

4.5. Validation of DPAP on A Single Pixel Imaging System

We also applied our scheme to a single pixel imaging system. In the actual experiment, we binarized the measurement matrix and loaded it into the DMD, and then used the measurement value of the single pixel imaging system for reconstruction. The pictures "flower" and "horse" were reconstructed at measurement rates of 0.1, 0.2, 0.3 and 0.5, respectively. The reconstruction results are shown in Figure 16. The reconstruction effect of DPAP was not significantly better than that of TVAL3. We think this may be caused by the following two reasons. First, when DPAP is applied to a single pixel imaging system, the measurement matrix needs to be binarized. The binarization operation reduces the accuracy of the measurement matrix, which leads to a decrease in the DPAP reconstruction effect. However, we can see in Figure 13 that DPAP outperforms TVAL3 when using the floating-point matrix. Secondly, the texture of the image we selected is relatively simple and TVAL3 is more suitable for reconstructing this sort of picture.

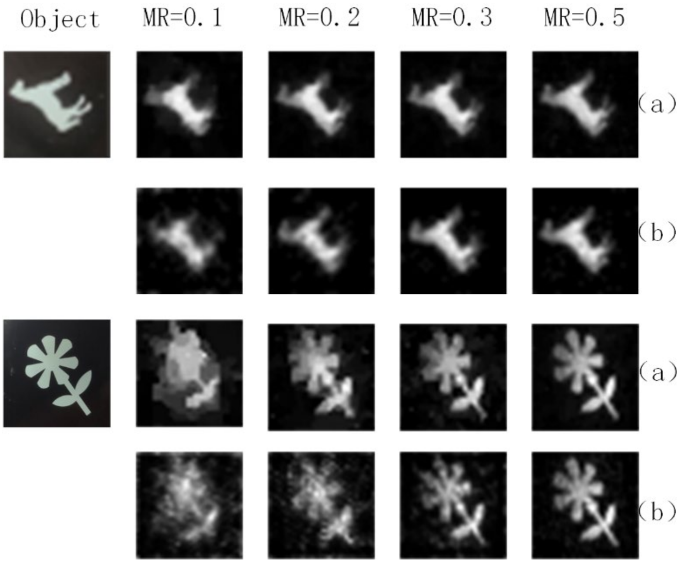

Figure 16. Reconstruction results of the proposed network and the conventional network on a single pixel imaging system, (**a**) the reconstruction algorithm is TVAL3; (**b**) the reconstruction algorithm is DPAP.

5. Conclusions

This paper proposes a compressed reconstruction network based on Deep Image Prior for single pixel imaging. A series of experiments show that the stage-learned network DPAP, which combines autoencoding priors and DIP, can make full use of the structural priors of the network. Compared with traditional TVAL3, ReconNet, and DR2Net, DPAP has better reconstruction effects and faster convergence speed. The higher the measurement rate, the more obvious the advantage of DPAP. DPAP also does not depend on large data sets, so it has broad application prospects in the field of medical imaging where real data inside the human body cannot be obtained and some areas with strong confidentiality. In addition, as long as the measurement matrix is binarized, it can be loaded on the DMD for actual imaging, which enables our network to be directly applied to the single pixel imaging system.

Author Contributions: Conceptualization, Q.Y. and J.L.; methodology, J.L. and S.L.; validation, J.L., Y.Z. and Z.W.; writing, J.L., Q.Y. and S.S.; data curation, J.L., S.L. and S.S.; supervision, Q.Y.; funding acquisition, Q.Y. All authors have read and agreed to the published version of the manuscript.

Funding: National Natural Science Foundation of China (62165009); National Natural Science Foundation of China (61865010).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Studer, V.; Bobin, J.; Chahid, M.; Mousavi, H.S.; Candes, E.; Dahan, M. Compressive fluorescence microscopy for biological and hyperspectral imaging. *Proc. Natl. Acad. Sci. USA* **2012**, *109*, E1679–E1687. https://doi.org/10.1073/pnas.1119511109.
2. Delogu, P.; Brombal, L.; Di Trapani, V.; Donato, S.; Bottigli, U.; Dreossi, D.; Golosio, B.; Oliva, P.; Rigon, L.; Longo, R. Optimization of the equalization procedure for a single-photon counting CdTe detector used for CT. *J. Instrum.* **2017**, *12*, C11014.
3. Yu, Y.; Liu, B.; Chen, Z. Improving the performance of pseudo-random single-photon counting ranging lidar. *Sensors* **2019**, *19*, 3620.
4. Wu, C.; Xing, W.; Feng, Z.; Xia, L. Moving target tracking in marine aerosol environment with single photon lidar system. *Opt. Lasers Eng.* **2020**, *127*, 105967.
5. Zhou, H.; He, Y.-H.; Lü, C.-L.; You, L.-X.; Li, Z.-H.; Wu, G.; Zhang, W.-J.; Zhang, L.; Liu, X.-Y.; Yang, X.-Y. Photon-counting chirped amplitude modulation lidar system using superconducting nanowire single-photon detector at 1550-nm wavelength. *Chin. Phys. B* **2018**, *27*, 018501.
6. Liu, Y.; Shi, J.; Zeng, G. Single-photon-counting polarization ghost imaging. *Appl. Opt.* **2016**, *55*, 10347–10351.
7. Liu, X.-F.; Yu, W.-K.; Yao, X.-R.; Dai, B.; Li, L.-Z.; Wang, C.; Zhai, G.-J. Measurement dimensions compressed spectral imaging with a single point detector. *Opt. Commun.* **2016**, *365*, 173–179.
8. Jiao, S.; Feng, J.; Gao, Y.; Lei, T.; Xie, Z.; Yuan, X. Optical machine learning with incoherent light and a single-pixel detector. *Opt. Lett.* **2019**, *44*, 5186–5189.
9. Zuo, Y.; Li, B.; Zhao, Y.; Jiang, Y.; Chen, Y.-C.; Chen, P.; Jo, G.-B.; Liu, J.; Du, S. All-optical neural network with nonlinear activation functions. *Optica* **2019**, *6*, 1132–1137.
10. Zheng, P.; Dai, Q.; Li, Z.; Ye, Z.; Xiong, J.; Liu, H.-C.; Zheng, G.; Zhang, S. Metasurface-based key for computational imaging encryption. *Sci. Adv.* **2021**, *7*, eabg0363.
11. Jiao, S.; Feng, J.; Gao, Y.; Lei, T.; Yuan, X. Visual cryptography in single-pixel imaging. *Opt. Express* **2020**, *28*, 7301–7313.
12. Tropp, J.A.; Gilbert, A.C. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Inf. Theory* **2007**, *53*, 4655–4666.
13. Figueiredo, M.A.; Nowak, R.D.; Wright, S.J. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE J. Sel. Top. Signal Processing* **2007**, *1*, 586–597.
14. Ji, S.; Xue, Y.; Carin, L. Bayesian compressive sensing. *IEEE Trans. Signal Processing* **2008**, *56*, 2346–2356.
15. Li, C. An Efficient Algorithm for Total Variation Regularization with Applications to the Single Pixel Camera and Compressive Sensing. Ph.D. Thesis, Rice University, Houston, TX, USA, 2010.
16. Lyu, M.; Wang, W.; Wang, H.; Wang, H.; Li, G.; Chen, N.; Situ, G. Deep-learning-based ghost imaging. *Sci. Rep.* **2017**, *7*, 1–6.
17. He, Y.; Wang, G.; Dong, G.; Zhu, S.; Chen, H.; Zhang, A.; Xu, Z. Ghost imaging based on deep learning. *Sci. Rep.* **2018**, *8*, 1–7.
18. Higham, C.F.; Murray-Smith, R.; Padgett, M.J.; Edgar, M.P. Deep learning for real-time single-pixel video. *Sci. Rep.* **2018**, *8*, 1–9.
19. Wang, F.; Wang, H.; Wang, H.; Li, G.; Situ, G. Learning from simulation: An end-to-end deep-learning approach for computational ghost imaging. *Opt. Express* **2019**, *27*, 25560–25572.
20. Wang, F.; Wang, C.; Chen, M.; Gong, W.; Zhang, Y.; Han, S.; Situ, G. Far-field super-resolution ghost imaging with a deep neural network constraint. *Light Sci. Appl.* **2022**, *11*, 1–11.
21. Zhu, R.; Yu, H.; Tan, Z.; Lu, R.; Han, S.; Huang, Z.; Wang, J. Ghost imaging based on Y-net: A dynamic coding and decoding approach. *Opt. Express* **2020**, *28*, 17556–17569.
22. Ulyanov, D.; Vedaldi, A.; Lempitsky, V. Deep image prior. In Proceedings of the IEEE conference on computer vision and pattern recognition, Salt Lake, UT, USA, 18–23 June 2018; pp. 9446–9454.
23. Mousavi, A.; Patel, A.B.; Baraniuk, R.G. A deep learning approach to structured signal recovery. In Proceedings of the 2015 53rd annual allerton conference on communication, control, and computing (Allerton), Monticello, IL, USA, 29 September–2 October 2015; pp. 1336–1343.
24. Kulkarni, K.; Lohit, S.; Turaga, P.; Kerviche, R.; Ashok, A. Reconnet: Non-iterative reconstruction of images from compressively sensed measurements. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 449–458.
25. Yao, H.; Dai, F.; Zhang, S.; Zhang, Y.; Tian, Q.; Xu, C. Dr2-net: Deep residual reconstruction network for image compressive sensing. *Neurocomputing* **2019**, *359*, 483–493.
26. Bora, A.; Jalal, A.; Price, E.; Dimakis, A.G. Compressed sensing using generative models. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 537–546.
27. Metzler, C.; Mousavi, A.; Baraniuk, R. Learned D-AMP: Principled neural network based compressive image recovery. *Adv. Neural Inf. Processing Syst.* **2017**, *30*, 1772–1783.
28. Metzler, C.A.; Maleki, A.; Baraniuk, R.G. From denoising to compressed sensing. *IEEE Trans. Inf. Theory* **2016**, *62*, 5117–5144.
29. Yoo, J.; Jin, K.H.; Gupta, H.; Yerly, J.; Stuber, M.; Unser, M. Time-dependent deep image prior for dynamic MRI. *IEEE Trans. Med. Imaging* **2021**, *40*, 3337–3348.
30. Gong, K.; Catana, C.; Qi, J.; Li, Q. PET image reconstruction using deep image prior. *IEEE Trans. Med. Imaging* **2018**, *38*, 1655–1665.
31. Mataev, G.; Milanfar, P.; Elad, M. DeepRED: Deep image prior powered by RED. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Korea, 27–28 October 2019.

32. Van Veen, D.; Jalal, A.; Soltanolkotabi, M.; Price, E.; Vishwanath, S.; Dimakis, A.G. Compressed sensing with deep image prior and learned regularization. *arXiv* **2018**, arXiv:1806.06438.
33. Tezcan, K.C.; Baumgartner, C.F.; Luechinger, R.; Pruessmann, K.P.; Konukoglu, E. MR image reconstruction using deep density priors. *IEEE Trans. Med. Imaging* **2018**, *38*, 1633–1642.
34. Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; Zhang, L. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Trans. Image Processing* **2017**, *26*, 3142–3155.
35. Bigdeli, S.A.; Zwicker, M. Image restoration using autoencoding priors. *arXiv* **2017**, arXiv:1703.09964.
36. Alain, G.; Bengio, Y. What regularized auto-encoders learn from the data-generating distribution. *J. Mach. Learn. Res.* **2014**, *15*, 3563–3593.
37. Dabov, K.; Foi, A.; Katkovnik, V.; Egiazarian, K. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Processing* **2007**, *16*, 2080–2095.
38. Zhang, K.; Zuo, W.; Zhang, L. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Trans. Image Processing* **2018**, *27*, 4608–4622.
39. Shi, B.; Lian, Q.; Chang, H. Deep prior-based sparse representation model for diffraction imaging: A plug-and-play method. *Signal Processing* **2020**, *168*, 107350.
40. Saxe, A.M.; Koh, P.W.; Chen, Z.; Bhand, M.; Suresh, B.; Ng, A.Y. On random weights and unsupervised feature learning. In Proceedings of the ICML, Bellevue, WA, USA, 28 June–2 July 2011.
41. Lim, B.; Son, S.; Kim, H.; Nah, S.; Mu Lee, K. Enhanced deep residual networks for single image super-resolution. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 136–144.
42. Chen, L.; Lu, X.; Zhang, J.; Chu, X.; Chen, C. HINet: Half instance normalization network for image restoration. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 182–192.
43. Zamir, S.W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F.S.; Yang, M.-H.; Shao, L. Multi-stage progressive image restoration. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14821–14831.
44. Liu, Q.; Yang, Q.; Cheng, H.; Wang, S.; Zhang, M.; Liang, D. Highly undersampled magnetic resonance imaging reconstruction using autoencoding priors. *Magn. Reson. Med.* **2020**, *83*, 322–336.
45. Wang, F.; Bian, Y.; Wang, H.; Lyu, M.; Pedrini, G.; Osten, W.; Barbastathis, G.; Situ, G. Phase imaging with an untrained neural network. *Light Sci. Appl.* **2020**, *9*, 1–7.
46. Dong, C.; Loy, C.C.; He, K.; Tang, X. Learning a deep convolutional network for image super-resolution. In Proceedings of the European conference on computer vision, Zurich, Switzerland, 6–12 September 2014; pp. 184–199.
47. Courbariaux, M.; Hubara, I.; Soudry, D.; El-Yaniv, R.; Bengio, Y. Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1. *arXiv* **2016**, arXiv:1602.02830.
48. Yu, J.; Fan, Y.; Yang, J.; Xu, N.; Wang, Z.; Wang, X.; Huang, T. Wide activation for efficient and accurate image super-resolution. *arXiv* **2018**, arXiv:1808.08718.
49. Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends® Mach. Learn.* **2011**, *3*, 1–122.