EnGe-CSNet: A Trainable Image Compressed Sensing Model Based on Variational Encoder and Generative Networks

Bowen Zheng 1, Jianping Zhang 2, Guiling Sun 1,* and Xiangnan Ren 1

1 College of Electronic Information and Optical Engineering, Nankai University, Tianjin 300350, China; zhengbwen@mail.nankai.edu.cn (B.Z.); 1611101@mail.nankai.edu.cn (X.R.)
2 Electrical Engineering and Computer Science, Northwestern University, Evanston, IL 60208, USA; JianpingZhang2018@u.northwestern.edu
* Correspondence: sungl@nankai.edu.cn; Tel.: +86-135-1245-3718

Abstract: The present study primarily investigates the topic of image reconstruction at high compression rates. As proven from compressed sensing theory, an appropriate algorithm is capable of reconstructing natural images from a few measurements since they are sparse in several transform domains (e.g., Discrete Cosine Transform and Wavelet Transform). To enhance the quality of reconstructed images in specific applications, this paper builds a trainable deep compressed sensing model, termed as EnGe-CSNet, by combining Convolution Generative Adversarial Networks and a Variational Autoencoder. Given the significant structural similarity between a certain type of natural images collected with image sensors, deep convolutional networks are pre-trained on images that are set to learn the low dimensional manifolds of high dimensional images. The generative network is employed as the prior information, and it is used to reconstruct images from compressed measurements. As revealed from the experimental results, the proposed model exhibits a better performance than competitive algorithms at high compression rates. Furthermore, as indicated by several reconstructed samples of noisy images, the model here is robust to pattern noise. The present study is critical to facilitating the application of image compressed sensing.

Keywords: compressed sensing; image reconstruction; neural networks

1. Introduction

Compressed Sensing (CS) [1] integrates the sampling and compression of information acquisition and significantly downregulates the sampling rate of the measurement system. The typical problem of CS refers to the reconstruction of sparse signal $\hat{x}$ from measurements $y \in \mathbb{R}^m$, $y = \Phi x$, (1)

where $\Phi \in \mathbb{R}^{m \times n} (m \ll n)$ denotes a general Gaussian random matrix with far fewer rows than columns, which is termed the measurement matrix. $x$ indicates any generalized sparse signal (e.g., sound, image or sensor data with spatial correlation). The compressed measurement $y$ is transmitted by the energy-limited system (e.g., Wireless Sensor Network). It carries nearly identical information to $x$, whereas it exhibits a much smaller size. Equation (1) denotes a system of under-determined equations. A unique solution is unlikely to be found unless $x$ exhibits special structures. Fortunately, most natural world images are sparse under some transformation, for example, Discrete Cosine Transform (DCT) [2] or Wavelet Transform (WT) [3]. For this reason, the signal required may be the sparsest solution for this system.

It refers to a Non-deterministic Polynomial hard (NP-hard) problem to seek the sparsest solution to an under-determined system of equations. Exact algorithms are considered to be incapable of solving NP-hard problems. As indicated in existing works, if
the matrix $\Phi$ satisfies some conditions, for example, Restricted Isometry Property (RIP) [4,5] or the related Restricted Eigenvalue Condition (REC) [6], the unique solution can be identified by using the convex optimization algorithm. Conventional algorithms to address the reconstruction problems consist of greedy algorithms [7], convex relaxation [8], and Bayesian framework [9]. The compression ratio and image quality of the mentioned methods on images recovering remain unsatisfactory. Deep Convolution Generative Adversarial Networks (DCGAN) [10] and Variational Autoencoder (VAE) [11] exhibit prominent performance on low dimensional representation of images. As indicated by Bora’s work [12], applying deep generative networks to image reconstruction as the prior information is suggested to have a positive effect.

As inspired by Larsen’s work [13], the present study proposes a deep CS model termed EnGe-CSNet to improve the quality of reconstructed images in Compressed Sensing applications. EnGe-CSNet integrates variational encoder and generative network. As impacted by the high structural similarity between images in the applications (e.g., crop monitoring and face detection), EnGe-CSNet are pre-trained on the image set to learn the general features. The learned features act as prior information when the algorithm reconstructs images using compressed measurements. To be specific, EnGe-CSNet is pre-trained on Arabidopsis thaliana seedlings images [14] or Large-scale CelebFaces Attributes (CelebA) dataset [15] to determine a mapping $G$ between low dimensional representation $z \in \mathbb{R}^k (k \leq m)$ and image $x \in \mathbb{R}^n$. When EnGe-CSNet reconstructs unknown images, the image $\hat{x}$ generated by $G$ tends to approach the original image $x$ by optimizing $z$. Compared with the existing approaches, the model proposed can conduct more effective reconstruction and exhibits a higher ability to resist noise.

The major contributions of the present study are summarized below:

- The present study builds novel convolutional generative networks termed as EnGe-CSNet that applies more to compressed sensing applications. EnGe-CSNet more effectively extracts the general features of target images in specific applications by integrating the advantages exhibited by VAE and DCGAN.
- The present study designs a novel deep CS framework to up-regulate the compression rate and improve the reconstruction quality in CS applications (e.g., crop monitoring and face detection). The model proposed employs pre-trained generative networks as prior information to overall exploit the structural similarity of images collected by sensors.
- The study verifies that the image reconstructed algorithms based on generative networks exhibit a strong anti-noise ability.

The present study is organized as follows: In Section 1, the research background, significance, contributions, article structure, and the fundamental idea of the proposed model are introduced. In Section 2, the relevant works on CS based on neural networks are outlined. In Section 3, the architecture and specific design of the proposed algorithm are presented. In Section 4, the experiment is elucidated, and the proposed model is compared with existing methods. In Section 5, the study is summarized, and the prospects for subsequent works are proposed.

2. Related Work

2.1. Compressed Sensing

Compressed Sensing theory aims to overcome the limitation of the Nyquist theorem and reconstruct the original signal from a few measurements. This theory requires original signal $x$ to be sparse, that is, only a few elements in vector $x$ are nonzero. The reconstruction problem is expressed as

$$\min_x \|x\|_0 \quad \text{s.t.} \quad y = \Phi x. \tag{2}$$

Candés [5] proved that sparse signals could be reconstructed with high probability if the measurement matrix $\Phi$ satisfy RIP:
\((1 - \delta)\|x\|_2^2 \leq \|\Phi x\|_2^2 \leq (1 + \delta)\|x\|_2^2\), \hspace{1cm} (3)

where \(\delta > 0\) is a small constant; \(\| \cdot \|_2\) is L2-norm. \(x_i\) denotes \(i\)th element of vector \(x\).

\[\|x\|_2 := \sqrt{\sum_{i=1}^{n} (x_i)^2}. \hspace{1cm} (4)\]

RIP ensures the difference between the measurements of two separate vectors. Most nature signals are not obviously sparse, whereas they are sparse under some transformation bases. Moreover, it is NP-hard to minimize L0-norm. Thus, in practice, Equation (2) is transformed to an L1-norm minimization problem

\[\min_s \|s\|_1 \hspace{1cm} \text{s.t.} \hspace{1cm} y = \Phi x, x = \Psi s, \hspace{1cm} (5)\]

where \(\Psi\) denotes the sparse basis; \(s\) expresses the sparse representation coefficient. As \(\Psi\) is known, we can reconstruct the original signal \(x\) after \(s\) is obtained.

Equation (5) has been extensively studied in depth, and the solution can be obtained quickly and accurately by using numerous algorithms [16]. Besides, some previous works have designed useful sensors for image CS. Zhang [17] presented a low power all-CMOS implementation of temporal compressive sensing with pixel-wise coded exposure, which can reconstruct 100 fps videos from coded images sampled at 5 fps. Another important application of image CS is to mine effective information from compressed measurements. Kwan [18] proposed a real-time framework for processing compressive measurements directly without image reconstruction. This study adopted a pixel-wise coded exposure (PCE) to condenses multiple frames into a single frame. This real-time system is applied to object detection and classification with the help of YOLO.

2.2. Image Reconstruction Based on Neural Networks

The conventional CS reconstruction algorithms requires the signal \(x\) to be \(k\)-sparse in a known basis, and these algorithms are effective in simple signal reconstruction. However, a suitable transform basis for a particular kind of image is hard to find. Bora [12] proposed to solve the problem of CS using generative models (dubbed CSGM). A more general Set Restricted Eigenvalue Condition (S-REC) is proposed in [12]. \(X \subseteq \mathbb{R}^n\) denotes the set of all possible images required. For parameters \(\gamma > 0\) and \(\delta \geq 0\), \(\Phi\) satisfies the S-REC if \(\forall x_1, x_2 \in X\),

\[\|\Phi (x_1 - x_2)\|_2^2 \geq \gamma\| (x_1 - x_2)\|_2^2 - \delta. \hspace{1cm} (6)\]

Random Gaussian measurement matrices meet the S-REC with a high probability for neural networks (e.g., VAE and DCGAN). CSGM employs the pre-trained generative network of DCGAN to map low dimensional latent space to image space. For this reason, the sparsity of signals is no longer required.

Some strategies have been proposed recently to improve the performance of CS based on Neural Networks. In [19], a framework jointly training a generator and the optimization for reconstruction via meta-learning was proposed. Shi [20] built an image CS framework by adopting convolutional neural networks (dubbed CSNet) that consisted of a sampling network and a reconstruction network. In [21], a deep learning architecture (dubbed ADMM-CSNet) was proposed for magnetic resonance images reconstruction. In [22], Deep neural network-based CS was also adopted on Magnetic Resonance Imaging (MRI) reconstruction.

3. Proposed Method

3.1. Image Reconstruction Using Generative Networks

Conventional image reconstruction methods require the sparse representation \(s\) of \(x\) to be established in advance as an initial condition. However, the most suitable sparse
matrix $\Psi$ is hard to find for a certain type of image. The sparsity essentially originates from the redundancy of information carried by natural images. The color and outline in a meaningful image should be regular. Besides, two photos in one type should be similar (e.g., two different face images contain numerous similar features). As assisted by convolutional neural networks, this study directly excavates the structural similarity of images instead of finding the sparse basis.

Unlike the conventional methods to reconstruct the sparse representation $s$ of image $x$, compressed sensing based on generative networks aims to find the low dimensional representation $z$. This present study builds the mapping $G$ from low dimensional latent space to image space in advance for a particular type of image in a specific application. The original image $x$ is restored after the low dimensional representation $z$ is determined. Since the dimension of hidden vector $z$ is lower than measurements $y$, the goal of CS is altered to find the solution of a system of over-determined equations, instead of under-determined equations. The present study solves the mentioned problem with the least square method. The objective of the reconstructed model is

$$\min_z \| \Phi G(z) - y \|^2_2.$$  

(7)

It can be solved by using gradient descent methods. After the optimal $z$ is found through several iterations, reconstructed image $\hat{x}$ is generated by

$$\hat{x} = G(z).$$  

(8)

The optimizer adopted here to solve Equation (7) is Adam [23]. It is helpful to add penalty terms while solving practical problems. As mentioned in [12], $\|z\|^2_2$ is an appropriate regularizer since both VAE and GAN typically impose an isotropic Gaussian prior on $z$. The penalty $\| \Phi G(z) - y \|_1$ can also improve the performance because it tends to make the error vector $e = \Phi G(z) - y$ more sparse (i.e., more element in $e$ is equal to zero). Thus, the objective function employed for minimization is expressed as:

$$\text{loss}(z) = \| \Phi G(z) - y \|^2_2 + \alpha \| z \|^2_2 + \beta \| \Phi G(z) - y \|_1,$$  

(9)

where $\alpha$ measures the importance of prior $\| z \|^2_2$ and $\beta$ denotes the ratio of L1-norm penalty to measurement error $\| \Phi G(z) - y \|^2_2$.

It is easy to fall into a locally optimal solution since gradient descent is a greedy algorithm. This study attempts to seek the global optimal solution by multiple restarts. Algorithm 1 expresses the pseudo code of the proposed model. The maximum restart times are set to $r_{\text{max}}$. A separate search is conducted after each restart. First, $z$ is initialized to a random point in space $\mathbb{R}^k$. Next, $i_{\text{max}}$ iterations are taken to optimize $z$. In the respective iteration, the loss is calculated by using Equation (8) and $z$ is updated by Adam with the learning rate $lr$. Next, the square of Euclidean distance $d = \| \Phi G(z) - y \|^2_2$ is employed to estimate the quality of the reconstructed image. If $d$ falls below a threshold $t$, it is considered that the correct image is found, and the algorithm will terminate. Otherwise, $z$ will be reinitialized randomly and the next search will be initiated.

Figure 1 illustrates the process of image acquisition in crop monitoring based on deep CS. The image $x$ of a plant seedling is compressively sampled with a wireless sensor node. The dimension of the measurements $y$ is significantly lower than the original image, so the amount of data transmitted decreases noticeably. Subsequently, a small number of measurements are sent to the base station via energy-limited wireless networks. When images are being reconstructed, the pre-trained generative network $G$ from EnGe-CSNet acts as a prior. The reconstructed image $\hat{x}$ is obtained after several iterations.
Algorithm 1 Proposed Approach

**Input:** \( \Phi, y, \alpha, \beta \) \( t \), learning rate \( lr \), maximum restart steps \( r_{\text{max}} \), Iterations \( i_{\text{max}} \)

for \( r = 0 \) to \( r_{\text{max}} \) do

Randomly initialize \( z \in \mathbb{R}^k \)

for \( i = 0 \) to \( i_{\text{max}} \) do

\[
\text{loss}(z) = \| \Phi \mathcal{G}(z) - y \|_2^2 + \alpha \| z \|_2^2 + \beta \| \Phi \mathcal{G}(z) - y \|_1
\]

Optimize \( \hat{z} \leftarrow z - lr \frac{\partial}{\partial z} \text{loss}(z) \)

end for

\[
d = \| \Phi \mathcal{G}(z) - y \|_2^2
\]

if \( d < t \) then

Break

end if

end for

**Output:** \( \hat{x} = \mathcal{G}(z) \).

---

**Figure 1.** The process of image acquisition based on deep CS. The image sensor compressed samples the original image and transmits the measurements to the base station. The image is reconstructed using the pre-trained generative network \( \mathcal{G} \) as prior information.

**3.2. Generative Adversarial Networks with Variational Encoder**

The framework of EnGe-CSNet proposed here is presented in Figure 2. \( Z \subseteq \mathbb{R}^k \) is assumed as the set of all possible low dimension latent vectors. \( p(X) \) is set as the distribution of images, and \( p(Z) \) is set as the distribution of latent vectors. As mentioned in [11], Generator, termed as called Decoder in VAE, is trained to learn a model \( p(X|Z) \) that maximizes \( p(X) \),

\[
p(X) = \sum_Z p(X|Z)p(Z).
\]

\( p(Z) \) is considered the standard Gaussian prior distribution in DCGAN. In deep CS, more stress is placed on the vectors that more likely generate natural images. The Encoder is trained to learn a posterior probability distribution \( q(X|Z) \).

**Figure 2.** The framework of EnGe-CSNet. The Encoder is used to find the low dimension latent vector \( z \) of image \( x \). The Generator is used to reconstruct images. The Discriminator is used to distinguish real and erroneous images.

The problem that VAE tends to produce blurred images is solved by Discriminator (\( D \)). It comprises three convolutional layers and two fully connected layers. Discriminator
attempts to distinguish the real from the erroneous images. When Discriminator is being trained, the target score of real image \( x \) is 1, and that of the image \( x_\eta \) generated with vector \( \eta \) sampled from standard Gaussian distribution is 0. As inspired by [10], cross-entropy acts as the loss function of Discriminator:

\[
L_D = -\log(D(x)) - \log(1 - D(x_\eta)). \tag{11}
\]

Generator \((G)\) covers a fully connected layer and four deconvolutional layers. It is updated twice in each iteration. In the DCGAN process, image \( x_\eta \) is generated using vector \( \eta \) sampled from standard Gaussian distribution to fool Discriminator. When Generator is being trained, the target score of \( x_\eta \) is 1. Thus, Equation (12) is derived as the loss function of the Generator in the DCGAN process:

\[
L_{G,GAN} = -\log(D(x_\eta)). \tag{12}
\]

In the VAE process, \( G \) generates image \( x_z \) with vector \( \eta \) sampled from \( q(X|Z) \). It aims to make the Euclidean distance between \( x_z \) and \( x \) to be minimum. Thus, Equation (13) is derived as the loss function of the Generator trained in VAE process:

\[
L_{G,VAE} = \|x_z - x\|_2^2 + \|D(x_z) - D(x)\|_2^2. \tag{13}
\]

Encoder \((E)\) comprises three convolutional layers and a fully connected layer. It finally outputs two vectors, i.e., a mean vector \( \mu \) and a variance vector \( \sigma \). \( z \) is randomly sampled from the normal distribution \( N(\mu, \sigma^2) \). As derived in [24], latent loss of VAE is

\[
L_L = -\sum_{i=1}^{k} \frac{1}{2} \left[ 1 + \log(\sigma_i^2) - \sigma_i^2 - \mu_i^2 \right]. \tag{14}
\]

where \( \mu_i \) denotes the \( i \)th element of vector \( \mu \); \( \sigma_i \) expresses the \( i \)th element of vector \( \sigma \). Encoder aims to find the latent vector \( z \) that makes \( G(z) \) approach \( x \) and fool the Discriminator. Thus, Equation (15) is derived as the loss function of the Encoder:

\[
L_E = \|G(z) - x\|_2^2 + \|D(G(z)) - D(x)\|_2^2 + \|z\|_2^2 + L_L. \tag{15}
\]

Regularizer \( \|z\|_2^2 \) is introduced since Gaussian representation is selected for the latent prior \( p(z) \) and the approximate posterior \( q(z|x) \).

4. Experiments and Discussion

4.1. Dataset and Training Details

The performance of the proposed model is tested with plant seedling and face images. The Aberystwyth leaf evaluation dataset contains four sets of 20 Arabidopsis Thaliana plants that have been grown in trays. Images of the respective tray are taken in a 15-min timelapse sequence. Interested readers can obtain this dataset from the following link: https://zenodo.org/record/168158#, accessed on 18 January 2021. CelebA is a large-scale face dataset with over 200,000 images involved. It can be obtained from this link: http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html, accessed on 25 January 2021. For both datasets, we randomly sample 19,200 images for training, and 1000 for evaluation. In this study, each sample is cut from a large-scale image and scaled to an RGB image of size 64 \times 64, giving \( n = 64 \times 64 \times 3 \) inputs. The value of the respective input is scaled to \([-1, 1]\).

The Encoder of EnGe-CSNet consists of three latent convolutional layers and one full connection output layer with a kernel size of 5 \times 5, strides of 2. In the latent layers, \( \text{relu} \) acts as the activation function. The dimension of latent vector \( z \) is set to \( k = 100 \). The Generator comprises one full connection layer and four transposed convolutional layers. Besides, all layers adopt the hyperbolic tangent function as activation. The discriminator consists of three convolutional layers and two full connection layers. All layers except for the output
employ leakyrelu activation. Adam is employed to update all parameters of EnGe-CSNet
with batch size 32 and learning rate 0.0001.

The optimizer of the reconstruction model refers to Adam with learning rate \( lr = 0.1 \).
This study searches up to \( r_{\text{max}} = 5 \) times for the optimal low dimensional representation
of each image. After the respective search step with \( i_{\text{max}} = 500 \) iterations, the search will be
terminated if \( d < 50 \), as the optimal solution is considered to be found.

The experiments’ hardware environment includes Intel(R) Xeon(R) W-2145 @3.70 GHz
CPU, 64.0 GB DDR4 Memory, and NVIDIA Quadro RTX4000 GPU. The software environ-
ment is Windows 10 64-bit operating system, Python as the programming language, and
Tensorflow2.0 as a library.

4.2. Comparisons with State-of-the-Art Approaches

4.2.1. Experimental Setup

For baselines, the proposed model is compared with Lasso [4], TVAL3 [25], NL-
RCS [26], DAMP [27], GAPTV [28] and CSGM [12]. The first five are optimization-based
methods. The implementation codes of these algorithms are downloaded from authors’
websites. Lasso is applied to reconstruct images in WT domain because we found it perform
better than in DCT domain. To reconstruct RGB images, we use CBM3D as the denoiser
of DAMP. The RGB versions of TVAL3, NLRCS and GAPTV do not exist currently, so we
reconstruct the three color channels independently. Other parameters for these methods,
including the number of iterations, are set to the default values suggested by authors
without any changes. CSGM is a reconstructed method based on generative models. The
architecture of the generator in the CSGM is identical to EnGe-CSNet. According to [12], we
optimize CSGM using Adam with a learning rate of 0.1 and do two random restarts with
500 update steps per restart and pick the reconstruction with the best measurement error.

The performance of the mentioned methods are tested at different compression rates
\( (cr = n / m, \text{that is, the ratio of the dimension of images } x \text{ and measurements } y) \). The regu-
larization parameters in Equation (8) are set to \( \alpha = 0.1 \) and \( \beta = 0.1 \) in the experiments since
these parameters are suggested to lead to the best performance according to experience.
The parameters’ influence is discussed in Section 4.3. For all methods, the measurement
matrix \( \Phi \) is a random Gaussian matrix with each entry sampled i.i.d from \( N(0, 1) \).

The Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) are adopted
for quantitative comparison. The PSNR is calculated by Equation (16).

\[
PSNR(x, \hat{x}) = 10 \log_{10}(\frac{255^2}{MSE(x, \hat{x})}),
\]

where \( MSE(x, \hat{x}) \) denotes the Mean Squared Error:

\[
MSE(x, \hat{x}) = \frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{x}_i)^2.
\]

The SSIM is calculated by Equation (18).

\[
SSIM(x, \hat{x}) = \frac{(\mu_x \mu_{\hat{x}} + c_1)(2\sigma_{x\hat{x}} + c_2)}{\mu_x^2 + \mu_{\hat{x}}^2 + c_1(\sigma_x^2 + \sigma_{\hat{x}}^2 + c_2)},
\]

where \( \mu_x \) expresses the mean of \( x \), \( \mu_\hat{x} \) is the mean of \( \hat{x} \), \( \sigma_x^2 \) is the variance of \( x \), \( \sigma_{x\hat{x}} \) is the
variance of \( \hat{x} \), \( \sigma_{x\hat{x}} \) is covariance between \( x \) and \( \hat{x} \), \( c_1 \) and \( c_2 \) are constants that avoid dividing
by zero.

4.2.2. Results and Discussion

Some visual examples reconstructed by various methods at the compression rate
\( cr = 20 \) are illustrated in Figures 3 and 4. Figure 3 gives four plant seedling images and
Figure 4 shows four face images. Lasso’s reconstructed results are illegible. The results of
TVAL3 are composed of many small color blocks, which makes the image unnatural. The
reconstructed images of NLRCS and DAMP are excessively smooth and lose considerable useful information. As indicated from the face reconstruction results of GAPTV, there are numerous jagged edges in the image. CSGM is capable of reconstructing the outline of plants with high probability, whereas many details are lost. By comparison, the images reconstructed by the proposed model are much explicit than others with the compression rate $cr = 20$. The edges of objects in images are easy to distinguish, which is more valuable information in applications. In terms of visual effect, the present study’s approach achieves better results.

![Ground Truth](image1.png) ![Lasso](image2.png) ![TVAL3](image3.png) ![NLRCS](image4.png)

**Figure 3.** Four examples of plant seedling images reconstructed by various methods. The compression rate is set to $cr = 20$.

![GAPTV](image5.png) ![DAMP](image6.png) ![CSGM](image7.png) ![Ours](image8.png)

**Figure 4.** Four examples of face images reconstructed by various methods. The compression rate is set to $cr = 20$.

Table 1 lists the quantitative evaluations of different approaches. The average PSNR and SSIM at different compression rates on two test datasets are shown. The optimal results are marked in bold font. According to the quantitative results, the proposed model is significantly superior to six compared CS methods on plant seedling images. With a low compression rate of $cr = 10$, the proposed model achieves the result similar to those achieved by NLRCS and DAMP. The PSNR&SSIM of three methods are 30.31&0.8110,
The performance of the model here complies with the existing methods when there are sufficient measurements. However, the advantages gradually appear with the increase of compression rate. At the compression rate of 20, the PSNR&SSIM of NLRCs and DAMP decrease to 27.30&0.7014, 26.68&0.6780. But our method’s performance is almost maintained, the PSNR&SSIM are 30.01&0.8020. The performance of the proposed method on the face dataset is slightly worse than that on the plant seedling images, whereas it remains better than comparison methods at the high compression rate. DAMP achieves the best results with the PSNR&SSIM of 29.05&0.8506 at a compression rate of 10. Nevertheless its performance declines rapidly with the increase of compression rate. At $cr = 50$, DAMP completely collapses, and the proposed model has PSNR&SSIM of 25.01&0.7492, which is significantly better than other methods. To be specific, when $cr = 10$, the performance of the proposed model is highly consistent with CSGM’s on the face dataset because neither of them has sufficient information to reconstruct the high-quality images. Yet both outperform other algorithms, which indicates that the compressed sensing method based on generative networks performs better at the high compression rate.

### Table 1. Average PSNR and SSIM comparisons of different image CS approaches. Bold indicates the best result.

| Images | Methods | $cr = 10$ | $cr = 20$ | $cr = 50$ | $cr = 80$ | $cr = 100$ |
|--------|---------|-----------|-----------|-----------|-----------|------------|
|        | PSNR    | SSIM      | PSNR      | SSIM      | PSNR      | SSIM       | PSNR      | SSIM     | PSNR      | SSIM      |
| Plants | Lasso   | 22.91     | 0.4851    | 20.97     | 0.3573    | 18.88      | 0.2431     | 17.93     | 0.2087    | 17.67      | 0.2049    |
|        | TVAL3   | 18.61     | 0.5670    | 17.58     | 0.4620    | 15.59      | 0.3491     | 14.80     | 0.3072    | 14.41      | 0.2904    |
|        | NLRCs   | 30.19     | 0.8044    | 27.32     | 0.7014    | 23.33      | 0.5162     | 21.48     | 0.4355    | 21.02      | 0.4156    |
|        | GAPTV   | 28.41     | 0.7911    | 26.32     | 0.6737    | 24.32      | 0.5379     | 23.29     | 0.4773    | 22.75      | 0.4554    |
|        | DAMP    | 30.22     | 0.8098    | 26.68     | 0.6780    | 10.28      | 0.1108     | 10.18     | 0.1528    | 10.34      | 0.1561    |
|        | CSGM    | 25.25     | 0.5724    | 24.89     | 0.5543    | 23.93      | 0.5120     | 23.17     | 0.4760    | 22.92      | 0.4669    |
|        | Ours    | 30.31     | 0.8110    | 30.01     | 0.8020    | 28.86      | 0.7618     | 26.66     | 0.6582    | 25.06      | 0.5798    |
| Faces  | Lasso   | 17.20     | 0.3802    | 15.69     | 0.2764    | 14.08      | 0.1765     | 13.28     | 0.1436    | 13.06      | 0.1328    |
|        | TVAL3   | 17.79     | 0.6410    | 14.87     | 0.4768    | 12.02      | 0.3218     | 10.27     | 0.2548    | 9.72       | 0.2269    |
|        | NLRCs   | 28.85     | 0.8458    | 24.06     | 0.7307    | 19.04      | 0.4705     | 17.53     | 0.3751    | 17.02      | 0.3458    |
|        | GAPTV   | 25.44     | 0.8208    | 23.01     | 0.7179    | 20.37      | 0.5355     | 19.26     | 0.4504    | 18.55      | 0.4077    |
|        | DAMP    | 29.05     | 0.8506    | 24.32     | 0.7519    | 6.55       | 0.0825     | 6.60      | 0.0967    | 6.57       | 0.0890    |
|        | CSGM    | 24.63     | 0.7315    | 24.34     | 0.7194    | 23.33      | 0.6796     | 22.38     | 0.6377    | 21.86      | 0.6055    |
|        | Ours    | 28.16     | 0.8324    | 26.67     | 0.8070    | 25.01      | 0.7492     | 23.05     | 0.6605    | 21.94      | 0.6008    |

The intuitive comparison presented in Figure 5 combines PSNR and SSIM. At the compression rate of 20, the SSIM of the proposed model reaches over 0.8 on both datasets, which is noticeably better than others. Moreover, Figures 3 and 4 show that the proposed model retains more significant details. In brief, it is a more suitable method for specific applications of compressed sensing image reconstruction. With sufficient datasets, the proposed method is capable of obtaining high-quality images with meager sampling rates.

Table 2 shows the average runtimes of comparison methods on plant seedling images. The runtime is not the focus of our work because our method work on GPU while the first six baselines are implemented in MATLAB or scikit-learn and only utilize CPU. Our method runs faster than CSGM when the compression rate is low (e.g., $cr = 10$ and $cr = 50$) because it reconstructs high-quality images with fewer restarts. It is hard to seek the optimal solution when the compression rate is high (e.g., $cr = 100$), so the proposed method costs more time than CSGM.
Table 2. The average runtime (seconds) for each method with the variety of compression rates. Bold indicates the best result.

| Methods | Devices | $cr = 10$ | $cr = 50$ | $cr = 100$ |
|---------|---------|-----------|-----------|------------|
| Lasso   | CPU     | 5.0481    | 0.2887    | 0.0443     |
| TVAL3   | CPU     | 0.9515    | 1.0463    | 1.0862     |
| NLRCs   | CPU     | 98.1406   | 97.9562   | 97.6856    |
| GAPTV   | CPU     | 2.5941    | 2.5623    | 2.5415     |
| DAMP    | CPU     | 5.8094    | 3.6383    | 3.4367     |
| CSGM    | GPU     | 0.2541    | 0.2486    | 0.2417     |
| Ours    | GPU     | **0.1564**| **0.2355**| **0.4935**|

Figure 5. Comparison of PSNR and SSIM at the compression rate of 20. The results of various methods on plant seedling images (left) and the CalebA dataset (right) are shown.

4.3. The Influence of Hyper-Parameters

Figure 6 shows the curves of the MSE of the reconstructed images using different hyper-parameters in Equation (9) with the number of iterations. In this subsection, the effect of $\alpha$ and $\beta$ on the performance of our algorithm is mainly evaluated. The reconstructed model is considered over-fitting if the MSE decreases rapidly in the early stage of iteration but is finally maintained at a large value. It is considered under-fitting if the MSE decreases slowly and can not reach the minimum error. In Figure 6 (left), $\beta$ is set to 0.1 and $\alpha$ is set to 0, 0.1, 0.5 respectively. It is shown that $\alpha = 0.1$ gives the best performance. The proposed method is over-fitting when $\alpha = 0$ and it is under-fitting when $\alpha = 0.5$. In Figure 6 (right), $\alpha$ is set to 0.1 and $\beta$ is set to 0, 0.1, 0.5, respectively. It is suggested that the three combinations exhibit the identical convergence rate at the early stage of iteration, but $\beta = 0.1$ finally reaches the minimum test error. The proposed method is over-fitting when $\beta = 0.5$ and it is under-fitting when $\beta = 0$. Hence, the recommended parameter combination here is $\alpha = 0.1, \beta = 0.1$.

Figure 6. The variety of average MSE test error with the increasing of epochs using 1000 plant seedling images. The influences of $\alpha$ (left) and $\beta$ (right) in Equation (9) are presented.
4.4. Anti-Noise Performance

The images captured by sensors are commonly accompanied by noise. Several possible causes are presented as follows:
1. The scene is insufficiently bright, or the brightness is non-uniform when the image sensor is operating.
2. The temperature of image sensors is excessively high due to their long working time.
3. Noise affecting the sensor’s circuit components may have an impact on the output image captured.

Accordingly, the image acquisition system should have strong anti-noise ability. The anti-noise performance is tested with Gaussian noise and salt-and-pepper noise. The image collected by the sensor is assumed below:

\[ x^* = noisy(x). \]  

(19)

The algorithm aims to reconstructed image \( \hat{x} \) from measurements \( y = \Phi x^* \). The anti-noise performance is assessed by comparing the similarity between \( \hat{x} \) and \( x \). Figure 7 presents some reconstructed samples of various methods. Gaussian noise is added to seedling and face images.

\[ noisy(x) = x + \sigma n, \]  

(20)

where \( \sigma \) denotes the ratio of noise, \( n \in \mathbb{R}^n \) is the Gaussian random noise, \( n_i \sim N(0, 1) \). As revealed from the visual comparison, the reconstructed image’s quality of the proposed approach is significantly higher than others. To be specific, the images reconstructed by TVAL3 and NLRCS are extremely blurry, which is even worse than noisy images. GAPTV is capable of recovering the target’s outline in general, whereas many color blocks cause the whole image lack details. DAMP achieves good results on several images, such as the female face, while other images are illegible. CSGM’s face reconstructed samples are unnatural. Though the outline is consistent with the ground truth, the details (e.g., eyes and mouth) are different. Besides, CSGM completely collapses while reconstructing the first plant seedling image. As opposed to the mentioned, the proposed model is more accurate in reconstructing image details and the edge of the target is clearer.

![Figure 7. Some reconstruction samples of plant seedling and face images adding Gaussian noise with \( cr = 10 \) and \( \sigma = 0.2 \). The ground truth, noisy images, and images reconstructed by various methods are illustrated.](image-url)
Table 3 shows the quantitative comparison of various methods with $cr = 10$ and $\sigma = 0.2$. The best results are marked in bold font. The proposed method’s PSNR and SSIM are higher than others on both plant seedling and face datasets. As revealed from the evaluation, GAPTV also achieves good results, whereas the samples in Figure 7 reveal that its visual effect is poor.

| Images   | TVAL3 | NLRCS | GAPTV | DAMP | CSGM | Ours   |
|----------|-------|-------|-------|------|------|--------|
|          | PSNR  | SSIM  | PSNR  | SSIM | PSNR | SSIM   | PSNR  | SSIM  |
| Plants   | 12.01 | 0.3028| 20.52 | 0.4090| 25.67| 0.6235 | 23.96 | 0.5552|
| Faces    | 14.16 | 0.4536| 20.15 | 0.5247| 23.70| 0.7176 | 23.08 | 0.6851|

In another experiment, salt-and-pepper (i.e., pixel loss) noise is added to images. The noisy image is defined as:

$$noisy(x)_i = \begin{cases} 
-1 & \text{if } p_i < \theta/2 \\
1 & \text{if } p_i > 1 - \theta/2 \\
x_i & \text{else}
\end{cases}$$

where $\theta$ denotes the ratio of pixels lost, and $p_i$ abides by the uniform distribution between 0 and 1. Figure 8 presents some reconstructed samples and Table 4 gives the quantitative evaluation with $cr = 10$ and $\theta = 0.02$. The results on images with salt-and-pepper noise provide the identical conclusion as that of images with Gaussian noise. In brief, the proposed model’s anti-noise performance is significantly better than that of other methods, and the model has strong robustness.

| Images   | TVAL3 | NLRCS | GAPTV | DAMP | CSGM | Ours   |
|----------|-------|-------|-------|------|------|--------|
|          | PSNR  | SSIM  | PSNR  | SSIM | PSNR | SSIM   | PSNR  | SSIM  |
| Plants   | 12.66 | 0.3015| 20.19 | 0.3914| 25.25| 0.6110 | 23.72 | 0.5472|
| Faces    | 14.27 | 0.4465| 19.90 | 0.5176| 23.61| 0.7125 | 22.88 | 0.6771|

Figure 8. Some reconstruction samples of plant seedling and face images adding salt-and-pepper noise with $cr = 10$ and $\theta = 0.02$. The ground truth, noisy images, and images reconstructed by various methods are illustrated.

Table 3. Average PSNR and SSIM comparisons using images with Gaussian noise. Bold indicates the best result.

Table 4. Average PSNR and SSIM comparisons using images with salt-and-pepper noise. Bold indicates the best result.
5. Conclusions

In this study, a deep CS model is proposed and introduced to specific CS applications (e.g., plant seedling or face images acquisition). The proposed model is capable of reconstructing images more effectively than existing models, significantly reducing the amount of data to be transmitted. As verified by experiments on both datasets, the performance of the method proposed here outperforms existing algorithms. The major advantage of the proposed algorithm is that it can effectively reconstruct the images at high compression rates (above 20). Thanks to convolutional neural networks, it is a data-driven method. Thus, another important advantage is that, with the increase of the training set, the generative model’s performance will be enhanced and the details of images can be restored more effectively. It also exhibits strong anti-noise ability. For the mentioned reason, it is a more practical method.

Moreover, the reconstruction of images exhibiting complex backgrounds is an open question. Given the different characteristics of the target and background, multiple generative networks are considered helpful. The recovering of details refers to another question. In subsequent studies, we will introduce some post-processing methods to improve reconstruction performance in depth.

Author Contributions: Conceptualization, B.Z. and J.Z.; methodology, B.Z.; software, B.Z.; validation, J.Z.; formal analysis, B.Z.; investigation, X.R.; resources, X.R.; data curation, J.Z.; writing—original draft preparation, B.Z.; writing—review and editing, G.S.; supervision, G.S.; project administration, G.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (No.61771262), Tianjin Science and Technology Major Project and Engineering (No.18ZXRHNC00140).

Data Availability Statement: All data generated or analyzed during this study are included in this article.

Acknowledgments: The authors would like to thank Tianjin Key Laboratory of Optoelectronic Sensor and Sensor Network Technology for providing working conditions.

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

References

1. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* 2006, 52, 1289–1306. [CrossRef]
2. Fracastoro, G.; Fosson, S.M.; Magli, E. Steerable discrete cosine transform. *IEEE Trans. Image Process.* 2016, 26, 303–314. [CrossRef]
3. Rein, S.; Reisslein, M. Scalable line-based wavelet image coding in wireless sensor networks. *J. Vis. Commun. Image Represent.* 2016, 40, 418–431. [CrossRef]
4. Candes, E.J.; Tao, T. Decoding by linear programming. *IEEE Trans. Inf. Theory* 2005, 51, 4203–4215. [CrossRef]
5. Candes, E.J. The restricted isometry property and its implications for compressed sensing. *C. R. Math.* 2008, 346, 589–592. [CrossRef]
6. Raskutti, G.; Wainwright, M.J.; Yu, B. Restricted eigenvalue properties for correlated Gaussian designs. *J. Mach. Learn. Res.* 2010, 11, 2241–2259.
7. Tropp, J.A.; Gilbert, A.C. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Inf. Theory* 2007, 53, 4655–4666. [CrossRef]
8. Bickel, P.J.; Ritov, Y.; Tsybakov, A.B. Simultaneous analysis of Lasso and Dantzig selector. *Ann. Stat.* 2009, 37, 1705–1732. [CrossRef]
9. Ji, S.; Xue, Y.; Carin, L. Bayesian compressive sensing. *IEEE Trans. Signal Process.* 2008, 56, 2346–2356. [CrossRef]
10. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv 2015*, arXiv:1511.06434.
11. Kingma, D.P.; Welling, M. Auto-encoding variational bayes. *arXiv 2013*, arXiv:1312.6114.
12. 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.
13. Larsen, A.B.L.; Sønderby, S.K.; Larochelle, H.; Winther, O. Autoencoding beyond pixels using a learned similarity metric. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 1558–1566.
14. Bell, J.; Dee, H. Aberystwyth leaf evaluation dataset. *2016*, 168158, 2. Available online: https://doi.org/10.5281/zenodo (accessed on 18 January 2021).
15. Liu, Z.; Luo, P.; Wang, X.; Tang, X. Deep learning face attributes in the wild. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3730–3738.

16. Chartrand, R.; Yin, W. Iteratively reweighted algorithms for compressive sensing. In Proceedings of the 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, Las Vegas, NV, USA, 31 March–4 April 2008; pp. 3869–3872.

17. Zhang, J.; Xiong, T.; Tran, T.; Chin, S.; Etienne-Cummings, R. Compact all-CMOS spatiotemporal compressive sensing video camera with pixel-wise coded exposure. Opt. Express 2016, 24, 9013–9024. [CrossRef] [PubMed]

18. Kwan, C.; Gribben, D.; Chou, B.; Budavari, B.; Larkin, J.; Rangamani, A.; Tran, T.; Zhang, J.; Etienne-Cummings, R. Real-Time and Deep Learning Based Vehicle Detection and Classification Using Pixel-Wise Code Exposure Measurements. Electronics 2020, 9, 1014. [CrossRef]

19. Wu, Y.; Rosca, M.; Lillicrap, T. Deep compressed sensing. arXiv 2019, arXiv:1905.06723.

20. Shi, W.; Jiang, F.; Liu, S.; Zhao, D. Image Compressed Sensing Using Convolutional Neural Network. IEEE Trans. Image Process. 2019, 29, 375–388. [CrossRef] [PubMed]

21. Yang, Y.; Sun, J.; Li, H.; Xu, Z. ADMM-CSNet: A deep learning approach for image compressive sensing. IEEE Trans. Pattern Anal. Mach. Intell. 2018, 42, 521–538. [CrossRef]

22. Seitzer, M.; Yang, G.; Schlemper, J.; Oktay, O.; Würl, T.; Christlein, V.; Wong, T.; Mohiaddin, R.; Firmin, D.; Keegan, J.; et al. Adversarial and perceptual refinement for compressed sensing MRI reconstruction. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Granada, Spain, 16–20 September 2018; pp. 232–240.

23. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. arXiv 2014, arXiv:1412.6980.

24. Odaibo, S. Tutorial: Deriving the Standard Variational Autoencoder (VAE) Loss Function. arXiv 2019, arXiv:1907.08956.

25. Li, C.; Yin, W.; Jiang, H.; Zhang, Y. An efficient augmented Lagrangian method with applications to total variation minimization. Comput. Optim. Appl. 2013, 56, 507–530. [CrossRef]

26. Dong, W.; Shi, G.; Li, X.; Ma, Y.; Huang, F. Compressive sensing via nonlocal low-rank regularization. IEEE Trans. Image Process. 2014, 23, 3618–3632. [CrossRef]

27. Metzler, C.A.; Maleki, A.; Baraniuk, R.G. From denoising to compressed sensing. IEEE Trans. Inf. Theory 2016, 62, 5117–5144. [CrossRef]

28. Yuan, X. Generalized alternating projection based total variation minimization for compressive sensing. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 2539–2543.