

Article

A Convolutional Neural Network-Based Quantization Method for Block Compressed Sensing of Images

Jiulu Gong¹, Qunlin Chen², Wei Zhu³ and Zepeng Wang^{1,*} 

¹ School of Mechatronic Engineering, Beijing Institute of Technology, Beijing 100081, China; gongjiulu@126.com

² North Automatic Control Technology Institute, Taiyuan 030006, China; qunlin_chen@163.com

³ Beijing Institute of Astronautical Systems Engineering, Beijing 100076, China; zhuwei9527@163.com

* Correspondence: 7520220149@bit.edu.cn

Abstract: Block compressed sensing (BCS) is a promising method for resource-constrained image/video coding applications. However, the quantization of BCS measurements has posed a challenge, leading to significant quantization errors and encoding redundancy. In this paper, we propose a quantization method for BCS measurements using convolutional neural networks (CNN). The quantization process maps measurements to quantized data that follow a uniform distribution based on the measurements' distribution, which aims to maximize the amount of information carried by the quantized data. The dequantization process restores the quantized data to data that conform to the measurements' distribution. The restored data are then modified by the correlation information of the measurements drawn from the quantized data, with the goal of minimizing the quantization errors. The proposed method uses CNNs to construct quantization and dequantization processes, and the networks are trained jointly. The distribution parameters of each block are used as side information, which is quantized with 1 bit by the same method. Extensive experiments on four public datasets showed that, compared with uniform quantization and entropy coding, the proposed method can improve the PSNR by an average of 0.48 dB without using entropy coding when the compression bit rate is 0.1 bpp.

Keywords: compressed sensing; quantization; convolutional neural network; image compression



Citation: Gong, J.; Chen, Q.; Zhu, W.; Wang, Z. A Convolutional Neural Network-Based Quantization Method for Block Compressed Sensing of Images. *Entropy* **2024**, *26*, 468. <https://doi.org/10.3390/e26060468>

Academic Editor: Jun Chen

Received: 4 March 2024

Revised: 25 May 2024

Accepted: 28 May 2024

Published: 29 May 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Compressed sensing (CS) [1–4] is very suitable for resource-constrained applications due to its low complexity [5–8]. In CS, it is necessary to quantize real-valued CS measurements. The CS framework with the quantization process is called quantized compressed sensing (QCS) [8].

The QCS primarily focuses on optimizing the encoder or decoder to enhance the quality of signal reconstruction for the commonly used quantization methods. The exploration of advanced compression techniques in the realm of block-based compressed sensing (BCS) has been a focal point in contemporary research. One such advancement is the application of Differential Pulse Code Modulation (DPCM) for quantized block-based compressed sensing of images. This strategy leverages DPCM's efficiency in exploiting spatial correlations within image blocks to minimize quantization errors, demonstrating improved bit-rate efficiency without sacrificing reconstruction quality [9]. However, the quantized measurements must undergo entropy coding to attain ideal performance. Ref. [10] proposes a progressive quantization method, which essentially improves the encoding and decoding strategies while still utilizing uniform scalar quantization. The reconstruction algorithm [11–14] has been a primary focal point in CS for optimizing the decoder.

In order to improve the quantization performance, vector quantization has been used to quantify the CS measurements [15,16]. Subsequently, to further amplify the efficiency

of the vector quantization technique, ref. [17] has leveraged deep neural networks. Due to the high computational complexity of vector quantization, scalar quantization is more suitable for CS measurements. In data compression theory, the scalar quantizer that performs entropy coding on quantized output data is usually called an entropy-constrained quantizer or entropy-coded quantizer [18]. When quantization error is used as a distortion measurement, the uniform scalar quantizer is the optimal entropy coding quantizer in rate-distortion performance [19,20]. In other words, for the BCS measurements, the rate-distortion performance of the uniform or non-uniform scalar quantization methods will be inferior to the joint performance of uniform quantization and entropy coding. Therefore, in current research on CS for images, the CS measurements are usually quantified using the uniform scalar quantization method [21], and the quantized measurements are encoded by entropy coding to improve the compression performance [22,23]. However, since the computational cost involved in entropy coding is usually high [23,24], using entropy coding will reduce the low-complexity advantage of the CS encoder.

There are two main ways to improve rate-distortion performance. The first way is to reduce the bitrate while keeping the distortion constant, while the second way is to reduce the distortion while keeping the bitrate constant. Using entropy coding on quantized data is the first way. Without considering entropy coding, the second way is the only method to improve rate-distortion performance. Moreover, there are strategies aimed at enhancing the compression efficiency of CS. For example, ref. [25] introduces a novel application of Zadoff–Chu sequences, renowned for their excellent autocorrelation properties. By utilizing this sequence in the measurement matrix, sparsity in the compressive domain is enhanced, leading to improved recovery accuracy for quantized CS data. Ref. [26] explores the use of Discrete Fourier Transform (DFT) for measurement, enabling parallel processing capabilities and enhancing the efficiency of block compressive sensing. This strategy capitalizes on the computational advantages of FFT to accelerate the CS process while maintaining reconstruction fidelity. While many approaches can improve compression efficiency at the encoding stage [25–28], they are not necessarily applicable or effective for QCS.

While the field of block compressed sensing (BCS) has witnessed significant advancements in recent years, traditional quantization techniques employed in image/video coding applications continue to face pivotal challenges. Specifically, uniform quantization, a common choice due to its simplicity and compatibility with entropy coding, often incurs substantial quantization errors, compromising the fidelity of the reconstructed images. Moreover, it tends to overlook the inherent structure and correlations within the data, leading to encoding redundancy and inefficiency. On the other hand, Lloyd–Max quantization, although theoretically optimal for minimizing mean squared error, necessitates computationally intensive offline training and struggles to adapt dynamically to varying image characteristics. Furthermore, the reliance on entropy coding as a supplementary step to mitigate the loss from quantization adds to the computational burden and complexity of the encoding process.

In light of these challenges, our work introduces a novel convolutional neural network (CNN)-based quantization method specifically designed to overcome the drawbacks of traditional approaches. By leveraging the power of deep learning, our method transcends the uniformity imposed by classic quantizers, achieving a more nuanced mapping of measurements that closely follows their underlying distribution. This adaptive quantization, coupled with a sophisticated dequantization mechanism that harnesses correlation from quantized data, significantly reduces quantization errors without resorting to additional entropy coding.

In this paper, we propose a quantization method of BCS measurements that can reduce distortion while maintaining the bitrate. At the end of encoding, the proposed method models the measurements' distribution of each image block. Subsequently, it quantizes the measurements of data that conform to a uniform distribution based on the distribution model. The proposed method uses the distribution parameters of each image block as side

information of the encoder and adopts the same strategy to quantize the side information with 1 bit. At the end of decoding, the proposed method first restores the quantized data to data that conform to the distribution of the original measurements, then extracts the correlation information of the measurements from the quantized data of adjacent blocks to correct the measurements. Before the dequantization of the measurements, the same strategy is used to dequantize the side information. All quantization and dequantization processes are implemented as convolutional neural networks (CNN), and all networks are jointly learned offline. The CS coding structure based on the proposed method is shown in Figure 1.

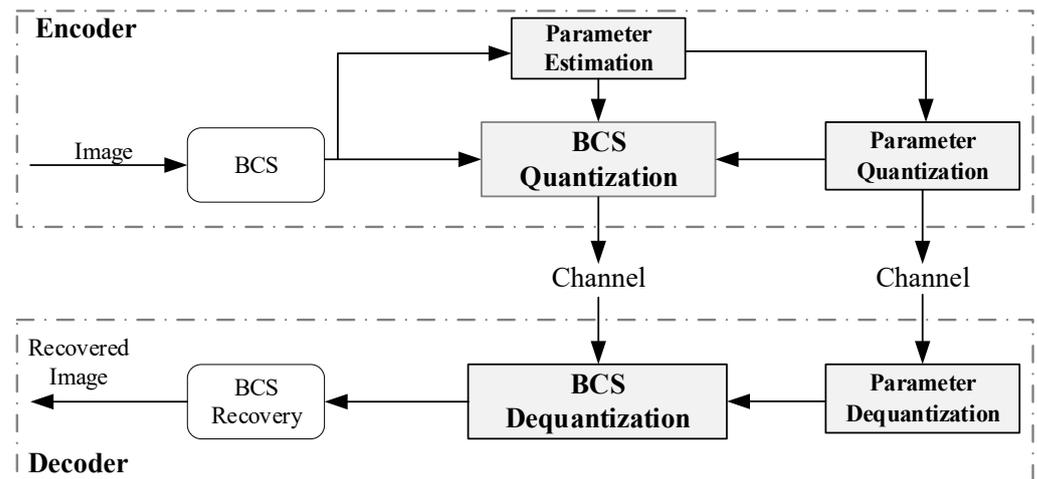


Figure 1. The CS coding structure based on the proposed method.

The main contributions of this article are as follows:

- A quantization method of BCS measurement based on CNN is proposed. The proposed method constructs and jointly trains the CNN of the quantization and dequantization processes, which aims to maximize the coding output entropy and minimize the quantization error.
- A quantization process based on measurements' distribution is proposed. Based on the properties of the cumulative distribution function (CDF), a neural network model was constructed to map measurements to the quantized data following a uniform distribution, which maximizes the amount of information carried in the quantized data. An activation function with a constrained output range was designed to reduce the computational complexity of the network's activation function.
- A dequantization process based on the neighborhood information of measurements is proposed. The inverse process of quantization is used as a module to restore the quantized data to data that conform to the distribution of the original measurements. Furthermore, an information correction module is introduced to extract correlation information from multiple quantized values for correcting the measurements. The two modules are used to improve the quality of the dequantized measurements through residual connection.
- The distribution parameters of the block measurements are used as side information, which is quantized with 1 bit by the same quantization process.

While conventional approaches such as uniform quantization and Lloyd–Max quantization with entropy coding have been widely employed, they often introduce significant quantization errors and encoding inefficiencies. Our work diverges from these methodologies by introducing a CNN-based quantization strategy that not only maps measurements to a uniformly distributed quantized space to maximize the information content but also incorporates a novel dequantization process that leverages correlations from the quantized data to minimize reconstruction errors. This innovative method bypasses the need for entropy coding, offering a more efficient and adaptive solution for BCS applications.

In comparison to uniform quantization, which assigns equal intervals to the entire dynamic range, and Lloyd–Max quantization, known for minimizing the mean squared error but requiring complex optimization, our CNN-based approach dynamically adapts to the underlying data distribution. Unlike entropy coding methods, which reduce redundancy at the expense of computational complexity, our method directly optimizes the quantization process through learning, achieving superior performance without additional encoding steps.

The core of our method lies in the design of the CNN architecture, which jointly learns the quantization and dequantization processes. This contrasts with traditional quantization techniques that rely on predetermined, static decision boundaries. Our network specifically tailors the quantization levels based on the input data’s statistical properties, ensuring a closer match to the original signal characteristics. Additionally, the utilization of correlation information from the quantized data for post-processing further distinguishes our method, leading to reduced quantization artifacts.

The rest of this paper is organized as follows. Section 2 introduces the proposed method, which mainly includes the BCS quantization process, parameter estimation, parameter quantization and dequantization, and the BCS dequantization process. Section 3 presents the experimental results. The conclusion is given in Section 4.

2. Proposed Method

In this paper, we aim to improve the amount of information of quantized data and the information extraction ability of the dequantization process. The proposed method mainly includes the following aspects: (1) BCS quantization process, (2) parameter estimation, (3) parameter quantization and dequantization, and (4) BCS dequantization process.

2.1. BCS Quantization Process

Ref. [29] demonstrates that a random variable’s CDF can transform its data distribution into a uniform distribution. The random variables are usually modeled by probability density functions (PDF) to describe their distributions. It is difficult to obtain a closed-form expression for the CDF due to the need for integral calculations involving the PDF. Due to the strong function approximation ability of feedforward neural networks, we propose utilizing a feedforward neural network to model the CDF of the CS measurements.

Assuming that $P(y)$ and $F(y)$ are the PDF and CDF of the measurement variable y , respectively, $P(y)$ and $F(y)$ need to satisfy the following conditions:

$$\begin{cases} F(y) \geq 0 \\ F(y) \leq 1 \\ P(y) = \frac{\partial F(y)}{\partial x} \geq 0 \end{cases} \quad (1)$$

Compared with a three-layer feedforward neural network, a four-layer feedforward neural network can accurately establish the relationship between input and target variables with fewer hidden neurons [30]. When employing a four-layer feedforward neural network to build the CDF of the measurements, it can be represented as:

$$\begin{cases} u_1 = y \\ u_2 = g_1(W_1 u_1 + d_1) \\ u_3 = g_2(W_2 u_2 + d_2) \\ F(y) = g_3(W_3 u_3 + d_3) \end{cases} \quad (2)$$

where g_1, g_2, g_3 represent the activation functions, and $W_1 \in \mathbb{R}^{L_1 \times 1}$, $W_2 \in \mathbb{R}^{L_2 \times L_1}$, $W_3 \in \mathbb{R}^{1 \times L_2}$, $d_1 \in \mathbb{R}^{L_1 \times 1}$, $d_2 \in \mathbb{R}^{L_2 \times 1}$, $d_3 \in \mathbb{R}^{1 \times 1}$ denote the model parameters.

Based on Equation (1), it is necessary to ensure that the output values of $F(y)$ fall within $[0, 1]$. In model (2), the activation function g_3 determines the output range of $F(y)$. Among the commonly used activation functions, the sigmoid function can ensure that the output values fall within $[0, 1]$. However, the sigmoid is a highly complex nonlinear

activation function. We propose a rectified linear activation function with a limited output range, which can be expressed as:

$$G(x) = \begin{cases} 1 & x \geq \alpha \\ \frac{x}{2\alpha} + \frac{1}{2} & -\alpha \leq x \leq \alpha \\ 0 & x \leq -\alpha \end{cases} \quad (3)$$

where α is a finite constant greater than 0. The curve graph and gradient curve of $G(x)$ are shown in Figure 2.

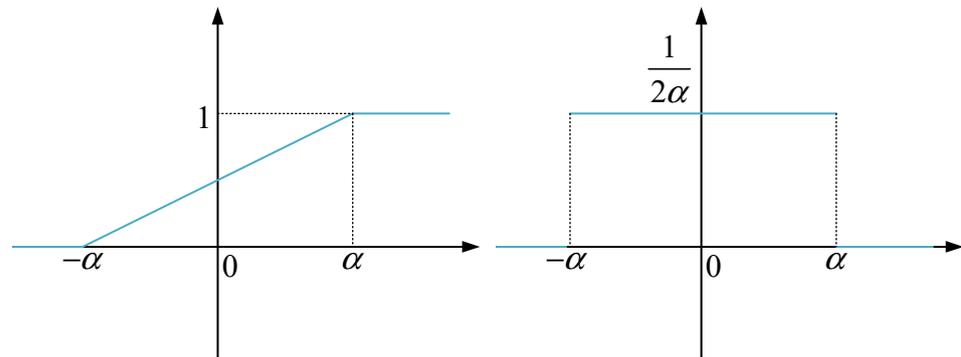


Figure 2. The curve and gradient curve of the proposed activation function.

To improve the adaptiveness of the activation function, we take α as a learnable parameter of the CDF model. Because activation functions usually do not come with trainable parameters, we transform Equation (3) as:

$$g_3(z) = \begin{cases} 1 & z \geq 1 \\ \frac{z}{2} + \frac{1}{2} & -1 \leq z \leq 1 \\ 0 & z \leq -1 \end{cases} \quad (4)$$

By comparing Equations (3) and (4), we obtain $G(x) = g_3(\frac{x}{\alpha})$. Let $\beta = \frac{1}{\alpha}$, and then the neural network model of CDF can be expressed as:

$$F(y) = g_3((\beta(W_3g_2((W_2g_1(W_1y + d_1) + d_2) + d_3))) \quad (5)$$

where $\beta \in \mathbb{R}$, and g_1, g_2 are the LeakyReLU activation function [31].

In BCS, all measurements are usually stored in the form of a matrix, which can be expressed as:

$$Y = [y_1, \dots, y_{N_B}] = \Phi [x_1, \dots, x_{N_B}] \quad (6)$$

where $Y \in \mathbb{R}^{M \times N_B}$ represents the measurements' matrix, Φ represents the measurement matrix, M represents the number of measurements for each image block, N_B represents the number of image blocks, x_i represents the pixel value vector of the i -th block, and y_i represents the measurements' vector of the i -th block.

When the input becomes a matrix, we can convert the feedforward neural network to a CNN. The number of convolutional layers in the CNN equals the number of parameter layers in the feedforward neural network. The number of channels in the convolutional network is equal to the number of neurons in the hidden layer. Since the feedforward neural network model has a single input and output, the kernel size of the convolutional network is 1×1 . Considering both quality and complexity, we set the output feature of the intermediate convolutional layer to six channels. The network structure diagram of CNN based on model (5) is shown in Figure 3.

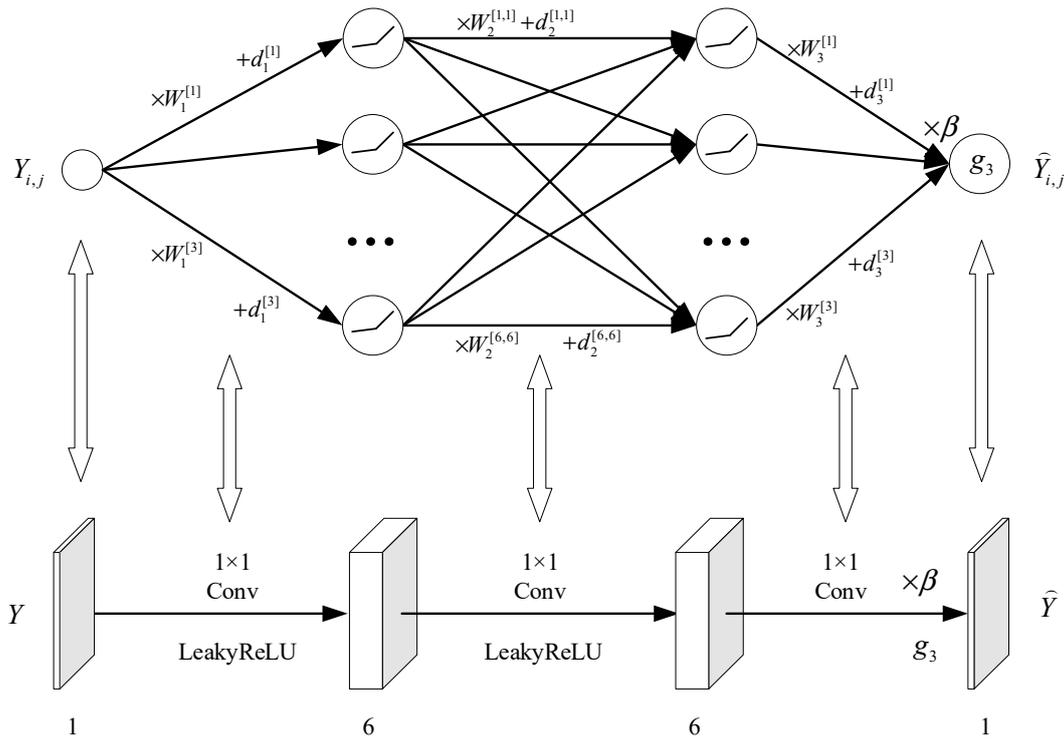


Figure 3. Network structure diagram of CNN based on model (5).

In Figure 3, “Conv” denotes convolutions, the numbers above “Conv” indicate the kernel size, “LeakyReLU” and “g3” represent the activation functions used in the current convolutional layer, and the rectangular boxes represent the output feature maps of the convolutional layer. The numbers below the rectangular boxes indicate the number of channels. To balance the computational cost and fitting performance of the model (5), we adopt $L_1 = L_2 = 6$ in this paper.

Due to variations between image blocks, the distribution parameters and CDF of different block measurements also differ. However, the model (5) does not account for the influence of distribution parameters, which limits its adaptability.

When using a Gaussian measurement matrix, the measurements of the same image block approximately follow a Gaussian distribution. Assuming that the measurement variable y_j of the j -th image block follows a Gaussian distribution $\mathcal{N}(y_j; \mu_j, \sigma_j)$, it can be transformed into a standard normal distribution $\mathcal{N}(z; 0, 1)$ through Equation (7), which can be expressed as:

$$z = \frac{y_j - \mu_j}{\sigma_j} \tag{7}$$

where μ_j and σ_j represent the location parameter (mean) and scale parameter (standard deviation), respectively. For the measurements’ matrix Y , Equation (7) can be expressed as:

$$Z = (Y - U) ./ \Lambda \tag{8}$$

where $U = [\{\mu_j\}]$ represents the matrix of position parameters, $\Lambda = [\{\sigma_j\}]$ represents the matrix of scale parameters, and $./$ represents the element-wise division of matrices.

After removing the impact of distribution parameters through Equation (8), the same CNN can transform the measurements of all blocks into a uniform distribution.

In order to reduce the computational burden of parameter extraction, the matrix $Y_0 \in \mathbb{R}^{M_0 \times N_B}$ of a small number of measurements is used for extracting the distribution parameters. The proposed BCS quantization process is shown in Figure 4. In Figure 4, “Repmat” represents repeat copies of an array, which is used to copy the parameters into the same size as the measurements’ matrix. Because the measurements of an image block

have the same position and scale parameters, the size of the position and scale parameters are $1 \times N_B$.

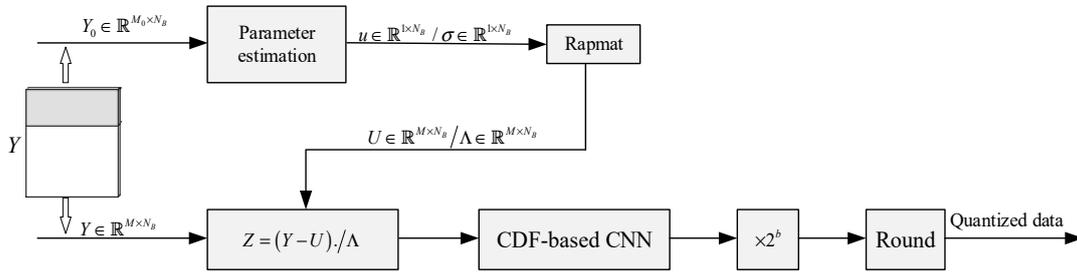


Figure 4. Structure diagram of the proposed BCS measurement quantization process.

2.2. Parameter Estimation

In general, the expressions of the mean (position parameter) μ and variance (scale parameter) σ^2 are as follows:

$$\begin{aligned} \mu_j &= \frac{\sum_i y_{j,i}}{N} \\ \sigma_j^2 &= \frac{1}{N} \sum_i (y_{j,i} - \mu)^2 \end{aligned} \tag{9}$$

Since using partial measurements to estimate distribution parameters may introduce some errors, we use a neural network to estimate the distribution parameters. According to the definition of convolution, Equation (9) can be expressed as:

$$\begin{aligned} \mu_j &= \frac{\sum_i y_{j,i}}{N} = [y_{j,1}, y_{j,2}, \dots] * W_\mu \\ \sigma_j^2 &= \frac{1}{N} \sum_i (y_{j,i} - \mu)^2 \\ &= [([y_{j,1}, y_{j,2}, \dots] - [y_{j,1}, y_{j,2}, \dots] * W_\mu) \\ &\quad \odot ([y_{j,1}, y_{j,2}, \dots] - [y_{j,1}, y_{j,2}, \dots] * W_\mu)] * W_\mu \end{aligned} \tag{10}$$

where $*$ denotes the convolution operation, $./$ represents the element-wise multiplication of matrices, and $W_\mu = [\frac{1}{N}, \dots, \frac{1}{N}]$ denotes the convolution kernel used for computing the mean.

Equation (10) shows that the mean and variance can be estimated using convolutions. Based on Equation (10), we construct CNNs to estimate the position and scale parameters, as shown in Figures 5 and 6, respectively. Since the mean and standard deviation functions are relatively simple in form, three channels are used for the middle convolution layer.

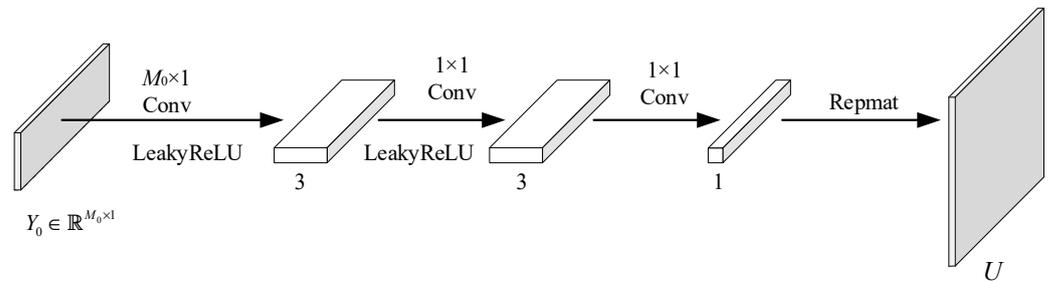


Figure 5. Network structure diagram for estimating the location parameters.

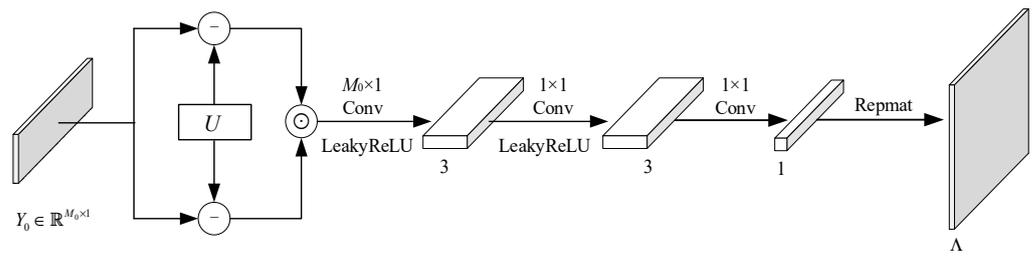


Figure 6. Network structure diagram for estimating the scale parameters.

2.3. BCS Dequantization Process

The information of the quantized data is not only present in a single quantized value but also exists among multiple quantized values. Typically, the dequantization process at the decoding end is the inverse of the quantization process at the encoding end. However, the proposed quantization process and its inverse process only operate on individual quantized values. If dequantization only utilizes the inverse process of the proposed quantization process, it cannot use the correlated information among the multiple quantized values. Therefore, we propose adding a measurements' information correction module to extract the measurement correction from the multiple neighboring quantized values. The dequantization process is shown in Figure 7.

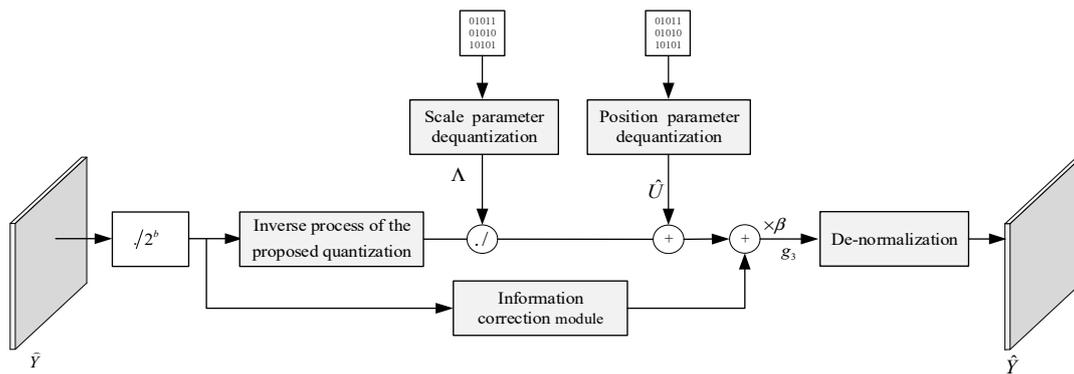


Figure 7. Structure diagram of the proposed dequantization process.

The quantization process is built based on the CDF and its inverse process can be constructed with the inverse function of the CDF. We adopt the same network architecture for the inverse process module. Since the similarity between adjacent image blocks is significant, we use convolution kernels of 1×3 to extract the compensation information from the neighborhood measurements. In order to improve the information extraction, five convolution layers are used in the measurement compensation information module and the inverse process of the proposed quantization. The network structure of the inverse process of the proposed quantization is shown in Figure 8. The network structure of the measurement compensation information module is illustrated in Figure 9.

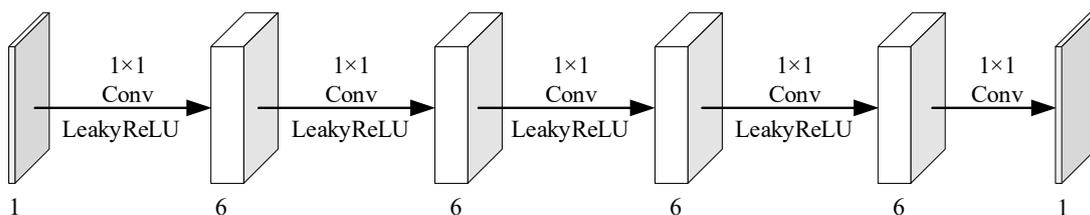


Figure 8. Network structure diagram of the inverse process of the proposed quantization.

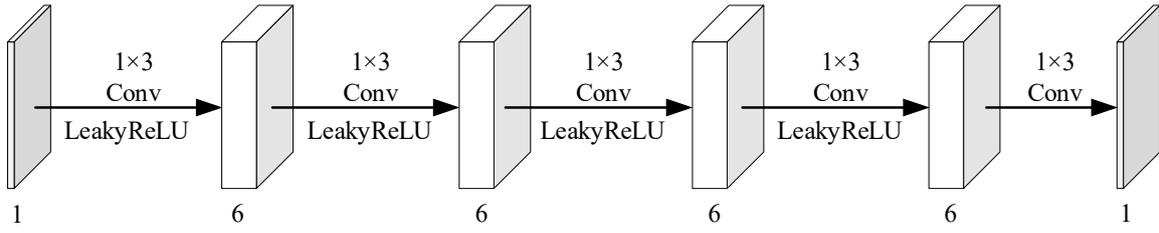


Figure 9. Network structure diagram of the measurement information correction module.

2.4. Parameter Quantization and Dequantization

Due to the high similarity between adjacent image blocks, the distribution parameters exhibit some similarity. Therefore, we use the same quantization and dequantization processes to quantize and dequantize the distribution parameters. To reduce the extra bits of the side information, we use 1 bit to quantize the side information. The network structures of quantization and dequantization for the parameters are shown in Figures 10 and 11, respectively.

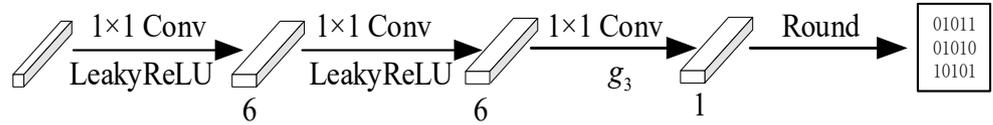


Figure 10. Network structure diagram for quantizing parameters.

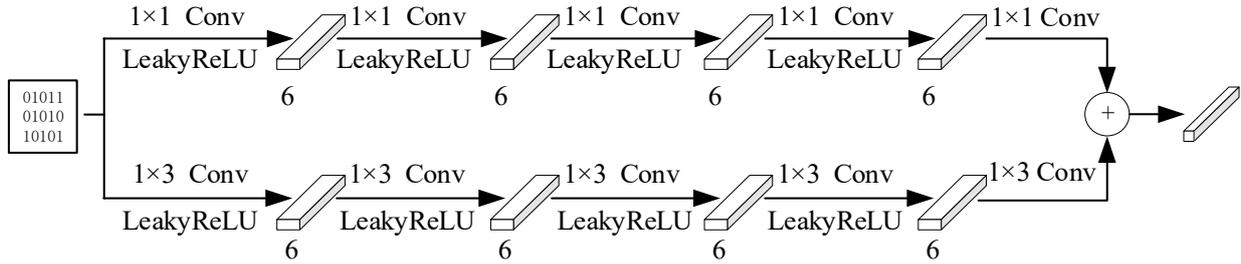


Figure 11. Network structure diagram for dequantizing parameters.

2.5. Local Normalization and Loss Function

2.5.1. Local Normalization of Measurements' Matrix

The element in the i -th row and j -th column of the measurements' matrix can be expressed as:

$$Y_{i,j} = \Phi_i \mathbf{x}_j = \sum_{k=1}^N \Phi_{i,k} \mathbf{x}_{k,j} \tag{11}$$

where Φ_i represents the i -th row of the measurement matrix Φ . The grayscale values of the images are typically represented by 256 levels, the pixel values satisfy the following:

$$0 \leq \mathbf{x}_{k,j} \leq 255, \mathbf{x}_{k,j} \in \mathbb{Z} \tag{12}$$

Combining Equations (11) and (12), we have:

$$255 \sum_{k=1}^N \min\{\Phi_{i,k}, 0\} \leq \sum_{k=1}^N \Phi_{i,k} \mathbf{x}_{k,j} \leq 255 \sum_{k=1}^N \max\{\Phi_{i,k}, 0\} \tag{13}$$

Based on Equation (13), we take $255 \sum_{k=1}^N \min\{\Phi_{i,k}, 0\}$ and $255 \sum_{k=1}^N \max\{\Phi_{i,k}, 0\}$ as the minimum and maximum values of $Y_{i,j}$, which can be expressed as:

$$\begin{aligned} Y_{i,j}^{max} &= 255 \sum_{k=1}^N \max\{\Phi_{i,k}, 0\} \\ Y_{i,j}^{min} &= 255 \sum_{k=1}^N \min\{\Phi_{i,k}, 0\} \end{aligned} \tag{14}$$

Based on Equation (14), each measurement can be normalized, which can be expressed as:

$$\tilde{Y}_{i,j} = \frac{Y_{i,j} - Y_{i,j}^{min}}{Y_{i,j}^{max} - Y_{i,j}^{min}} \tag{15}$$

According to Equation (14), it is known that the same row of Y shares the same maximum and minimum values. Therefore, Equation (15) is referred to as the local normalization method for the measurements' matrix.

Since Equation (14) only requires computation based on the measurement matrix, the row normalization method does not require transmitting the maximum and minimum value. Equation (15) transforms all measurements into real numbers between 0 and 1, and the input and output of the CNNs in the quantization and the dequantization processes are also real numbers between 0 and 1. Therefore, the quantization of the measurements' matrix can be expressed as:

$$Q[Y] = \text{Round}\left(F_{q-net}(\tilde{Y})(2^b - 1)\right) \tag{16}$$

where b represents the bit-depth of quantization, and F_{q-net} represents the CNN in the quantization process.

At the decoding end, the output value of the dequantization network needs to be denormalized, which can be represented as:

$$\hat{Y} = F_{dq-net}\left(Q^{-1}\left[Q\left[F_{q-net}(\tilde{Y})\right]\right]\right)\left(Y_{i,j}^{max} - Y_{i,j}^{min}\right) + Y_{i,j}^{min} \tag{17}$$

where F_{dq-net} represents the CNN in the dequantization process.

2.5.2. Loss Function

The main objective of the proposed method is to optimize the amount of information retained in the quantized data while minimizing the quantization error. The average information of data is usually measured by information entropy. However, since the information entropy depends on probability statistics, it cannot propagate the gradient. In this paper, we propose a continuous function to estimate the information entropy when training the network.

The entropy of the quantized measurements' matrix \hat{Y} can be represented as follows:

$$H(\hat{Y}) = -\sum_k P(s_k) \log_2(P(s_k)) \tag{18}$$

where $P(s_k)$ represents the probability of the symbol s_k in \hat{Y} , which requires counting the number of occurrences of symbol s_k .

We can count the numbers greater than s_k in the measurements' matrix by using a step function, which is defined as follows:

$$h(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{19}$$

The numbers greater than s_k in the measurements' matrix can be represented as:

$$\sum_i \sum_j h(\widehat{Y}_{i,j} - s_k) \tag{20}$$

The step function $h(x)$ is not differentiable at 0. To ensure differentiability, we use the sigmoid activation function to approximate $h(x)$. The approximate function can be represented as follows:

$$h(x) \approx \text{sigmoid}(\eta x) \tag{21}$$

When $\eta \rightarrow +\infty$, we can get:

$$\text{sigmoid}\left(\eta\left(\widehat{Y}_{i,j} - s_k + \varepsilon\right)\right) = \begin{cases} 1 & \widehat{Y}_{i,j} \geq s_k \\ 0 & \widehat{Y}_{i,j} < s_k \end{cases} \tag{22}$$

where $0 < \varepsilon < 1$ is used to ensure that the function maps to 1 when $\widehat{Y}_{i,j} = s_k$, and we set $\varepsilon = 0.5$. The parameter η can be used to adjust the variation of the sigmoid function. Based on practical experience, setting $\eta = 64$ can make the output values of the sigmoid function approach 0 or 1 as much as possible.

Based on Equation (22), the probability greater than s_k in the measurements' matrix can be represented as:

$$P(\widehat{Y} \geq s_k) \approx \frac{\sum_i \sum_j \text{sigmoid}\left(\eta\left(\widehat{Y}_{i,j} - s_k + 0.5\right)\right)}{MN_B} \tag{23}$$

Assuming that $s_{k+1} > s_k$, we have:

$$P(s_k) \approx P(\widehat{Y} \geq s_k) - P(\widehat{Y} \geq s_{k+1}) \\ \approx \frac{\sum_i \sum_j \text{sigmoid}\left(\eta\left(\widehat{Y}_{i,j} - s_k + 0.5\right)\right)}{MN_B} - \frac{\sum_i \sum_j \text{sigmoid}\left(\eta\left(\widehat{Y}_{i,j} - s_{k+1} + 0.5\right)\right)}{MN_B} \tag{24}$$

Based on the above, the computation of information entropy can be estimated as:

$$H(\widehat{Y}) = -\sum_k \left(P(\widehat{Y} \geq s_k) - P(\widehat{Y} \geq s_{k+1})\right) \log_2 \left(P(\widehat{Y} \geq s_k) - P(\widehat{Y} \geq s_{k+1})\right) \tag{25}$$

To simultaneously minimize the quantization error and maximize the information entropy, the objective function is composed of two parts. The first part is the mean square error (MSE) between the CS measurements before and after quantization, and the second part is the information entropy of the quantized measurements. The loss function is as follows:

$$\text{Loss} = \text{MSE}(Y, \widehat{Y}) - \lambda H(\widehat{Y}) \tag{26}$$

where $\lambda > 0$ is a parameter that controls the importance of the information entropy of quantized data. \widehat{Y} represents the dequantized measurements.

3. Results

In this section, we present various experimental results that validate the performance of our method. The proposed method is primarily implemented through CNNs, which requires the collection of a training dataset for network training. The training dataset comprises 200 training images taken from the BSDS500 dataset [32]. Each image has been cropped into grayscale images of size 256×256 with a stride of 60 pixels. The block size

utilized in BCS is set at 16×16 . The samples and labels of the training data set are both a matrix of BCS measurements for each image. The matrix used to collect the measurements has a sampling rate of 0.8, so the trained network can be applied to any measurements' matrix with a sampling rate lower than 0.8. Each block typically requires at least ten measurements to reconstruct an image from the BCS measurements efficiently, so we take $M_0 = 10$ as the number of the partial measurements.

All CNNs were implemented using the Pytorch framework. We trained the CNNs of the quantization and dequantization processes together. The batch size was set to 32, with the optimization process performed using the Adam algorithm, initialized with a learning rate of 0.001. After the initial training of 10,000 epochs, the learning rate was reduced by a factor of 10, and all networks were trained for an additional 20,000 epochs. The training process was conducted on a server powered by an Intel Xeon CPU, a Nvidia RTX 2080Ti GPU with 11 GB of memory, and 128 GB of DDR4 RAM. The test images consisted of an APC, aerial, airplane, airport, building, moon surface, tank, and truck, as illustrated in Figure 12. Publicly available datasets such as Set5 [33] (5 images), Set11 [34] (11 images), Set14 [35] (14 images), and BSD68 [36] (68 images) were also employed. All experiments were conducted on an Intel Core i5-8300H CPU @ 2.30GHz, and the proposed method's performance was measured using the PSNR of the reconstructed images.

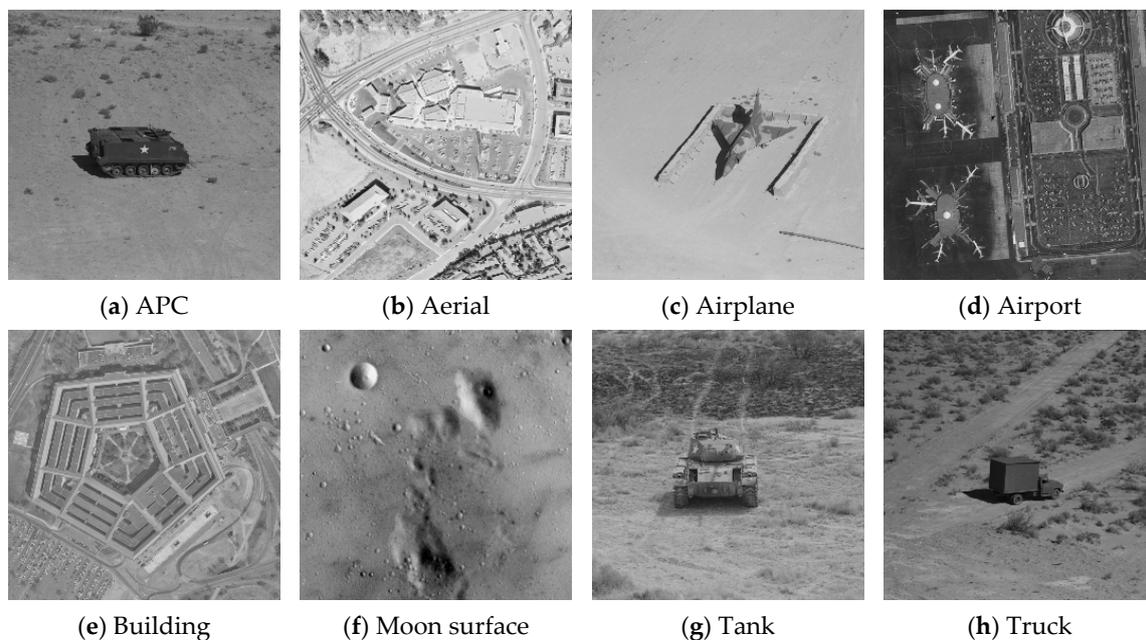


Figure 12. Eight test images.

3.1. Analysis of Measurement Reconstruction Results

In this section, we analyzed the number of reconstruction levels and the quantization errors of the quantization method.

The current studies on improving CS quantization methods typically focus on sparse signals, but these methods are not suitable for images with a large number of elements. To ensure a low complexity of the encoder, the BCS encoder usually uses uniform quantization and entropy coding to process the BCS measurements. In addition, uniform quantizer is considered as the optimal quantizer for entropy-coded quantization in data compression theory [23,24], which is why BCS encoders tend to use uniform quantization and entropy coding. Currently, the most advanced quantization techniques for BCS of images are believed to be the prediction quantization method [9] and the progressive quantization method [10]. However, they essentially improve the coding strategy, while the quantization method employed is still uniform quantization, which can explore improvements using the commonly used quantization methods and the proposed method. To simplify the experi-

mental process, this paper only compares the quantization techniques, using entropy-coded uniform quantization, μ -law quantization [37,38], and Lloyd–max quantization [39,40] as benchmarks. The entropy-coded uniform quantization method refers to the use of entropy coding after performing uniform quantization on the measurements. The codebook for Lloyd–Max quantization was obtained through offline training.

In scalar quantization methods, the number of reconstruction levels is typically equal to the number of quantization levels, as shown in Table 1. Table 2 shows the number of reconstructed levels of the proposed method for eight test images at a measurement rate of 0.2. Comparison between Tables 1 and 2 shows that the proposed method have more different elements in the dequantized measurements. This is mainly because the proposed method utilizes the information of multiple quantized values for the dequantization, which gives the proposed method the advantage of many-to-many mapping. Moreover, each row of the measurements' matrix adopts different maximum and minimum values for local denormalization. The local normalization approach also increases the number of reconstruction levels in accordance with the increase in measurement rate.

Table 1. Number of reconstruction levels for scalar quantization methods.

Quantization Bit-Depth	2	3	4	5	6	7	8
Number of reconstruction levels	4	8	16	32	64	128	256

Table 2. Number of reconstruction levels for the proposed quantization.

Quantization Bit-Depth	2	3	4	5	6	7	8
APC	147	236	397	690	1147	1860	2921
Aerial	199	321	559	981	1712	2863	4368
Airplane	181	273	453	724	1122	1644	2336
Airport	186	283	488	845	1483	2545	4009
Building	161	243	414	739	1302	2277	3699
Moon surface	171	270	455	782	1328	2205	3515
Tank	161	241	413	744	1345	2422	3963
Truck	144	227	396	699	1220	2140	3527
Average	168.7	261.7	446.8	775.5	1332.3	2244.5	3542.2

The greater the number of reconstruction levels of dequantized measurements, the greater the quantization error can be reduced. Tables 3–5 display the MSE of the various quantization methods for the measurements quantized with 3-bit, 6-bit, and 8-bit, respectively.

Table 3. MSE of the measurements quantized with 3-bit.

Image	Uniform Quantization	μ -Law Quantization	Lloyd–Max	Proposed Method
APC	567.30	675.16	493.99	115.76
Aerial	1468.86	1250.97	1223.14	192.34
Airplane	1147.16	1280.45	1279.69	161.52
Airport	704.45	435.69	460.30	155.19
Building	864.87	768.95	629.43	113.59
Moon surface	1327.97	831.08	573.44	115.48
Tank	780.74	730.11	634.44	121.87
Truck	540.80	489.04	486.02	121.88
Average	925.27	807.68	722.56	137.20

Table 4. MSE of the measurements quantized with 6-bit.

Image	Uniform Quantization	μ -Law Quantization	Lloyd–Max	Proposed Method
APC	6.86	7.15	11.43	2.53
Aerial	17.77	16.11	21.37	3.87
Airplane	14.04	13.86	19.62	3.20
Airport	8.27	5.41	9.03	3.33
Building	11.02	9.78	12.89	2.51
Moon surface	15.11	10.68	11.66	2.58
Tank	9.61	8.58	12.32	2.63
Truck	6.60	5.88	10.21	2.63
Average	11.16	9.68	13.57	2.91

Table 5. MSE of the measurements quantized with 8-bit.

Image	Uniform Quantization	μ -Law Quantization	Lloyd–Max	Proposed Method
APC	0.4194	0.4337	1.8716	0.1742
Aerial	1.0692	1.0155	2.6658	0.2978
Airplane	0.8546	0.8304	2.5470	0.2356
Airport	0.5031	0.3274	1.5868	0.2543
Building	0.6684	0.5927	2.0241	0.1770
Moon surface	0.9308	0.6440	1.8995	0.1778
Tank	0.5884	0.5104	1.9121	0.1980
Truck	0.4032	0.3630	1.7278	0.1902
Average	0.6796	0.5896	2.0293	0.2131

Table 3 shows that when using 3-bit quantization, the proposed method reduces the MSE by 788.07, 670.48, and 585.35 compared with uniform quantization, μ -law quantization, and Lloyd–Max quantization, respectively. Similarly, in Table 4, the proposed method accomplishes a reduced MSE by 8.25, 6.77, and 10.66 when 6-bit quantization is employed. Table 5 reveals that when 8-bit quantization is applied, the proposed method reduces the MSE by 0.4665, 0.3765, and 1.8162 compared with uniform quantization, μ -law quantization, and Lloyd–Max quantization, respectively. Tables 3–5 demonstrate that the proposed method has a significantly lesser MSE than other quantization methods.

3.2. Analysis of the Impact of Entropy Loss Constraints

In this section, we analyzed the effect of entropy constraint. The parameter λ in the loss function determines the degree of entropy constraint. In order to analyze the appropriate value of λ , we only select a few common values for training the CNNs of the quantization and dequantization processes. Table 6 shows the MSE of the dequantized measurements and the information entropy of the quantized measurements when 8-bit is used to quantize the measurements of the BSDS500 dataset.

Table 6. MSE and entropy for different λ .

λ	MSE	Entropy
0	0.2883	7.2624
0.05	0.2841	7.3376
0.1	0.2867	7.3539
0.2	0.2896	7.3787
0.5	0.3027	7.4487

It can be seen from Table 6 that the entropy constraint can increase the information entropy of the quantized measurements but it has a slight impact on reducing the MSE of

the dequantized measurements. When $\lambda = 0.05$, the MSE of the dequantized measurements is the smallest. Therefore, when training the proposed network, we use $\lambda = 0.05$.

In addition, it can be observed that the entropies of the quantified measurements are very close to the bit-depth. Some images may not be compressed when using a fixed code table for entropy coding. In other words, the measurements quantized by the proposed method do not need entropy coding.

3.3. Analysis of the Impact of the Measurement Information Correction Module

In this section, we analyzed the impact of the information correction module on the dequantization process. Table 7 shows the quantization performance of the BSDS500 dataset when different information correction modules are used in the proposed method. In Table 7, Contrast Scheme 1 did not use a measurement information correction module. Contrast Scheme 2 used a measurement information correction module composed of three convolutional layers. Contrast Scheme 3 used a measurement information correction module composed of six convolutional layers.

Table 7. Quantization performance of different information correction modules.

	MSE	Entropy
μ -law quantification	1.0941	7.1510
Contrast Scheme 1	0.3298	7.1760
Contrast Scheme 2	0.3027	7.2430
Contrast Scheme 3	0.2841	7.3376

Table 7 reveals that the proposed method has significant advantages over the μ -law quantization method. Compared with Contrast Scheme 1, the MSE of Contrast Scheme 2 is reduced by 0.0271, while its entropy is increased by 0.067. Similarly, the MSE of Contrast Scheme 3 is reduced by 0.0457 and the entropy is increased by 0.1616. These results illustrate that the information correction module effectively improves the quantization performance. Furthermore, the information correction module exhibits a stronger correction capability with the increase of convolutional layers.

3.4. Rate-Distortion Performance Comparison

In this section, we compared the rate-distortion performance of the proposed method with the entropy-coded uniform quantization, μ -law quantization, and Lloyd–Max quantization. The entropy-coded uniform quantization method refers to the use of entropy coding after performing uniform quantization on the measurements, which is expressed by “uniform quantization + entropy coding” in this paper. When drawing the rate-distortion curve, we traverse multiple quantization bit-depths and sampling rates to encode and decode the test images. Then, we choose the optimum Bitrate-PSNR points and connect them with a line. The bit-depth adopts seven values in $\{2, 3, 4, \dots, 8\}$, and the sampling rate chooses 77 values in $\{0.04, 0.05, 0.06, \dots, 0.8\}$. The image reconstruction algorithm used is the BCS-SPL-DCT algorithm [41]. When calculating the bitrate of “uniform quantization + entropy coding,” the average codeword length of entropy coding is replaced by information entropy. Figure 13 shows the PSNR curve of the eight test images.

In Figure 13, the proposed method has the best rate-distortion performance on all eight test images, particularly for the aerial, building, and tank images. The PSNR curve of “uniform quantization + entropy coding” is better than the μ -law and Lloyd–Max quantization methods. This observation confirms that the existing quantizers without entropy coding have inferior rate-distortion performance compared with “uniform quantization + entropy coding”.

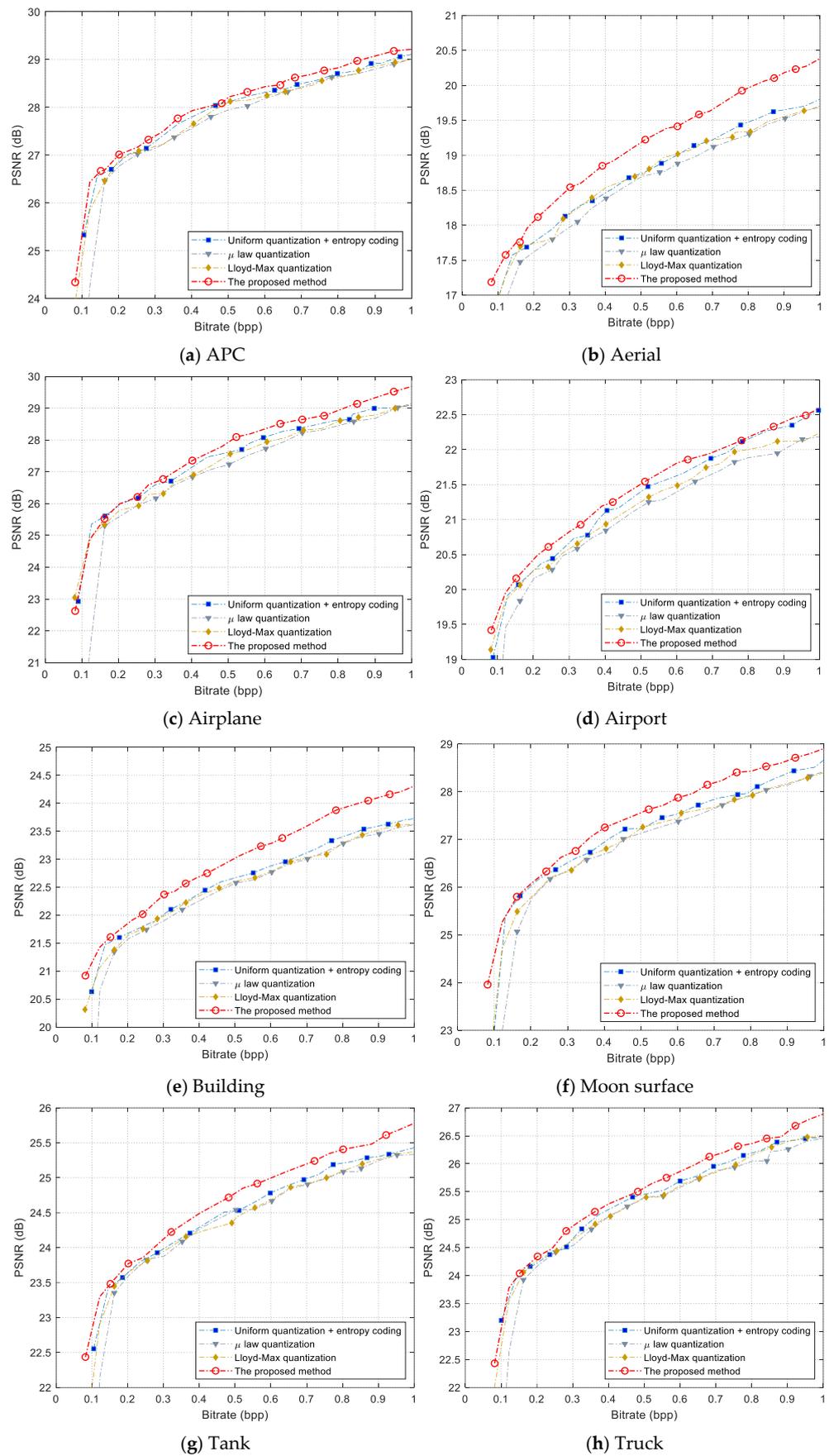


Figure 13. PSNR curves of the eight test images.

Figure 14 shows the reconstructed images of the eight test images with different methods at a compression bit rate of 0.1. The Lloyd–Max quantization approach generates an adaptive quantization dictionary for each image. We do not count the bits of the quantization dictionary in the compression bit rate of the Lloyd–Max quantization method. Therefore, the results of the Lloyd–Max quantization method in Figure 14 are equivalent to the optimal results of the conventional quantizer.

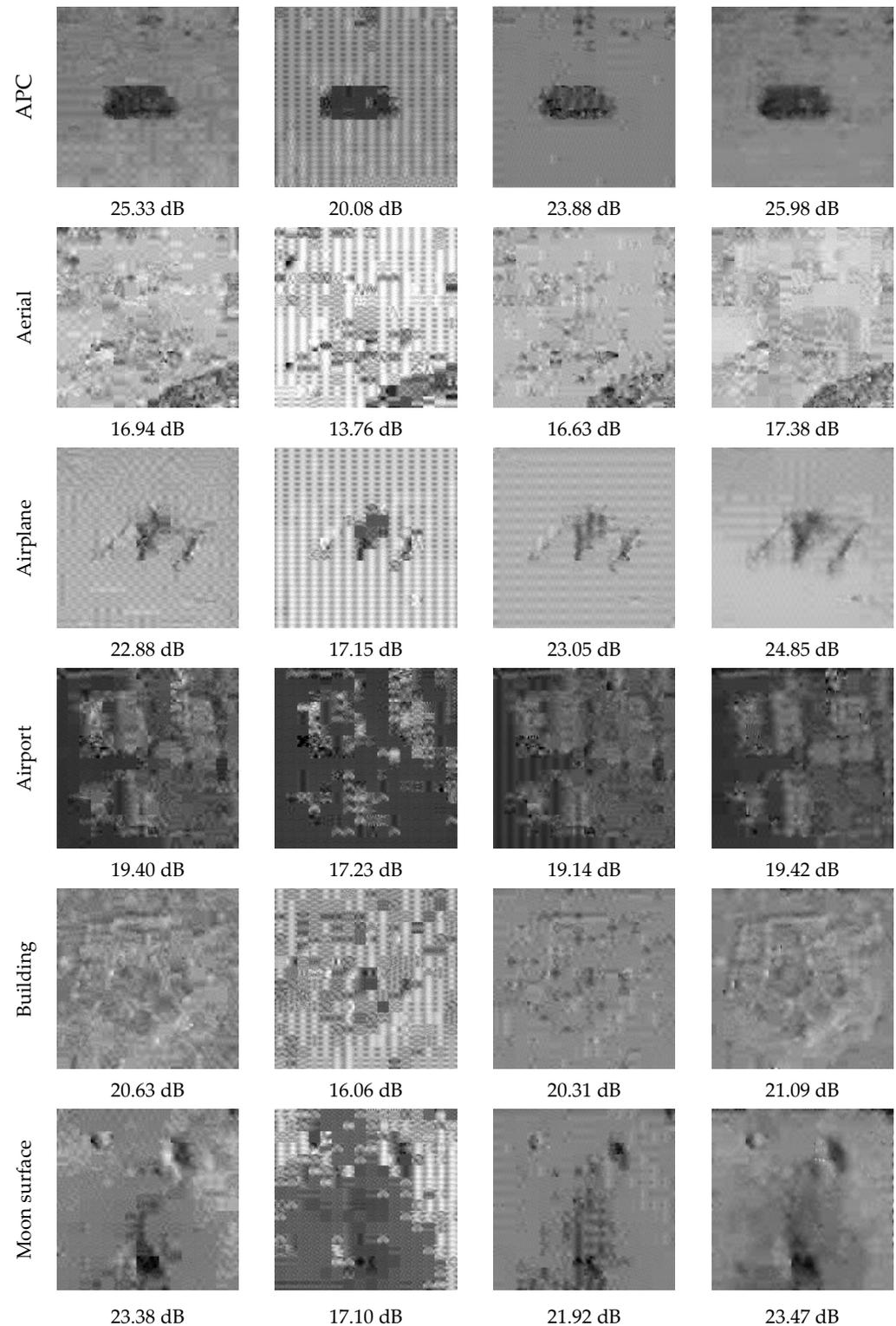


Figure 14. Cont.

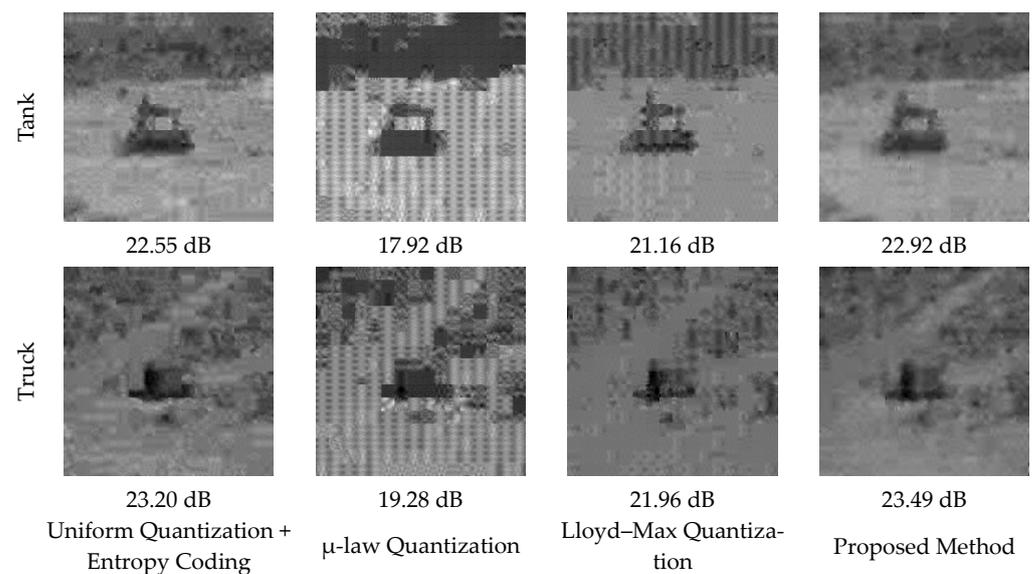


Figure 14. Visual comparison of different methods at a compression bit rate of 0.1.

As shown in Figure 14, the proposed method exhibits the best visual effect and PSNR, followed by “uniform quantization + entropy coding” and Lloyd–Max quantization. Compared with “uniform quantization + entropy coding,” for the eight test images, the PSNR of the proposed method increased by 0.65 dB, 0.44 dB, 1.97 dB, 0.02 dB, 0.46 dB, 0.09 dB, 0.37 dB, and 0.29 dB, respectively. Compared with the Lloyd–Max quantization, for the eight test images, the PSNR of the proposed method increased by 2.1 dB, 0.75 dB, 1.8 dB, 0.28 dB, 0.78 dB, 1.55 dB, 1.76 dB, and 1.53 dB, respectively.

The four quantization methods were also tested on the four test image datasets, and the average PSNR curves are shown in Figure 15.

In Figure 15, the average PSNR is the mean of the PSNR of the reconstructed images at a given bit rate for all images in the dataset. The PSNRs at a given bit rate are obtained by linear interpolation from the Bitrate-PSNR curve for each image. The given bit rates are set to $\{0.1, 0.2, \dots, 1 \text{ bpp}\}$. For datasets Set5, Set11, Set14, and BSD68, the average PSNR curve of the proposed method is better than “uniform quantization + entropy coding,” μ -law quantization and Lloyd–Max quantization. Particularly, at a low bit rate (around 0.1 bpp), the proposed method’s PSNR is much higher than the other methods.

The datasets SunHays80 [42] and Urban100 [43] have been extended for testing (all data are converted to grayscale images with 256×256). The quality of reconstruction is evaluated by the peak signal to noise ratio (PSNR) and the structural similarity (SSIM) between the reconstructed image and the original image. Table 8 shows the PSNRs and SSIMs of the four datasets at a bit rate of 0.1 bpp. Table 9 shows the PSNRs and SSIMs of the four datasets at a bit rate of 0.2 bpp.

For all images of the six datasets, when the bitrate is set to 0.1 bpp, the proposed method, “uniform quantization + entropy coding,” μ -law quantization, and Lloyd–Max quantization achieve average PSNRs of 19.69 dB, 19.24 dB, 17.54 dB, and 18.67 dB, respectively. Compared with “uniform quantization + entropy coding,” the proposed method improves the PSNR by an average of 0.45 dB without entropy coding. The proposed method, “uniform quantization + entropy coding,” μ -law quantization, and Lloyd–Max quantization achieve average SSIMs of 0.1855, 0.1739, 0.1408, and 0.1547 respectively. Compared with “uniform quantization + entropy coding,” the proposed method improves the SSIM by an average of 0.0116 without entropy coding.

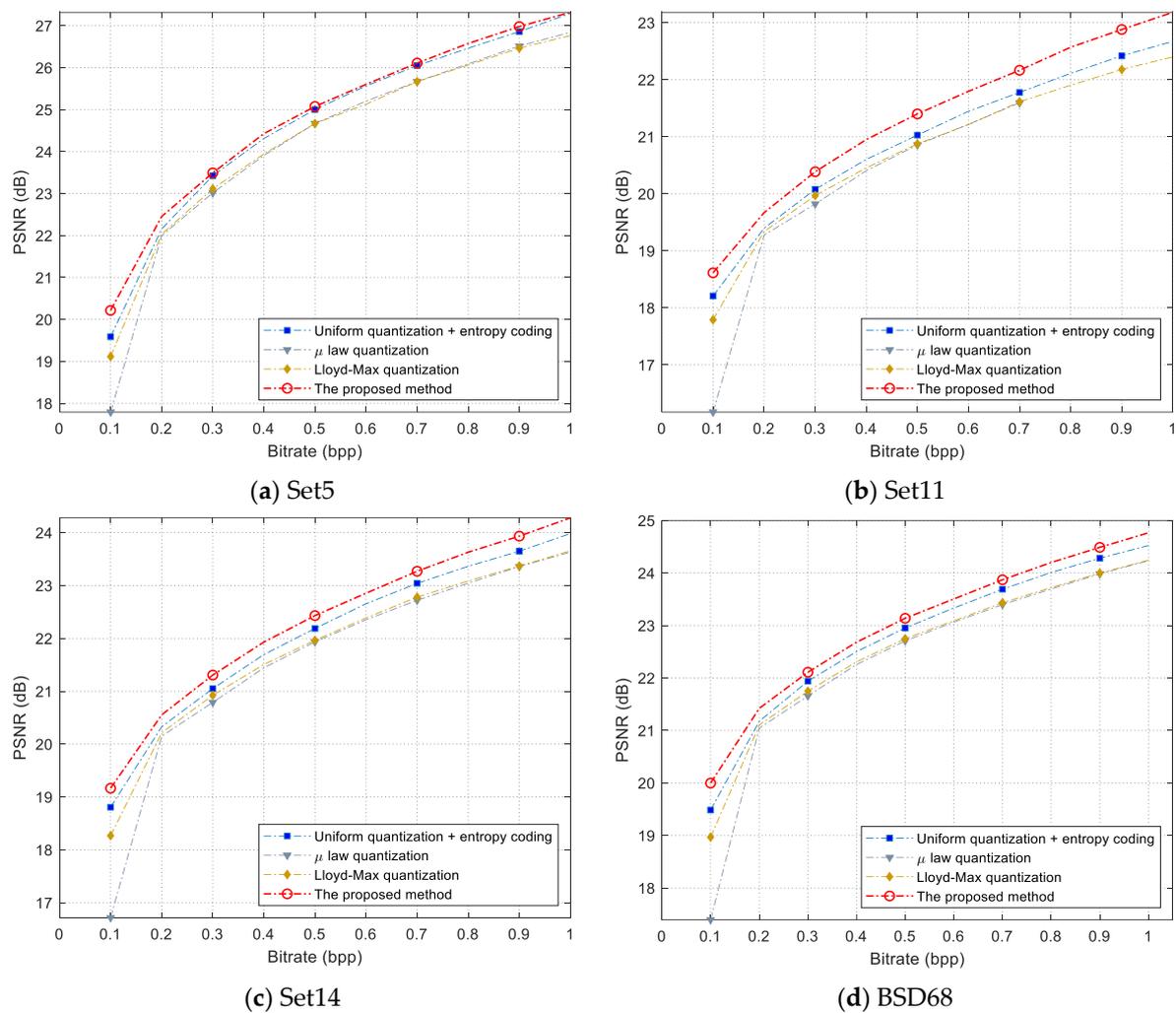


Figure 15. Average PSNR curves of test image sets.

Table 8. The PSNRs of the four datasets at a bit rate of 0.1 bpp.

		Set5	Set11	Set14	BSD68	SunHays80	Urban100	Average
Uniform Quantization + Entropy Coding	PSNR	19.59	18.21	18.81	19.49	19.32	20.01	19.24
	SSIM	0.2117	0.1526	0.1769	0.1547	0.1361	0.2117	0.1739
μ-law Quantization	PSNR	17.79	16.17	16.72	17.40	17.30	19.86	17.54
	SSIM	0.1546	0.1176	0.1336	0.1080	0.1001	0.2311	0.1408
Lloyd–Max Quantization	PSNR	19.12	17.79	18.27	18.97	18.75	19.12	18.67
	SSIM	0.1909	0.1355	0.1587	0.1328	0.1196	0.1909	0.1547
Proposed Method	PSNR	20.22	18.61	19.17	20.00	19.90	20.25	19.69
	SSIM	0.2229	0.1672	0.1875	0.1636	0.1489	0.2229	0.1855

For all images of the six datasets, when the bitrate is set to 0.2 bpp, the proposed method, “uniform quantization + entropy coding,” μ-law quantization, and Lloyd–Max quantization achieve average PSNRs of 21.27 dB, 21.09 dB, 20.88 dB, and 20.93 dB, respectively. Compared with “uniform quantization + entropy coding,” the proposed method improves the PSNR by an average of 0.18 dB without entropy coding. The proposed method, “uniform quantization + entropy coding,” μ-law quantization, and Lloyd–Max quantization achieve average SSIMs of 0.2738, 0.2683, 0.2543, and 0.2567, respectively. Com-

pared with “uniform quantization + entropy coding,” the proposed method improves the SSIM by an average of 0.0055 without entropy coding.

Table 9. The PSNRs of the four datasets at a bit rate of 0.2 bpp.

		Set5	Set11	Set14	BSD68	SunHays80	Urban100	Average
Uniform Quantization + Entropy Coding	PSNR	22.26	19.41	20.32	21.27	20.99	22.26	21.09
	SSIM	0.3246	0.2269	0.2653	0.2421	0.2260	0.3246	0.2683
μ -law Quantization	PSNR	22.00	19.28	20.15	21.05	20.84	21.94	20.88
	SSIM	0.3127	0.2100	0.2516	0.2275	0.2096	0.3141	0.2543
Lloyd–Max Quantization	PSNR	22.03	19.34	20.22	21.11	20.87	22.03	20.93
	SSIM	0.3127	0.2168	0.2542	0.2298	0.2139	0.3127	0.2567
Proposed Method	PSNR	22.43	19.6707	20.55	21.42	21.13	22.43	21.27
	SSIM	0.3249	0.2421	0.2725	0.2473	0.2313	0.3249	0.2738

Across all images in the six datasets, the proposed method demonstrates superior performance compared to standard uniform quantization combined with “uniform quantization + entropy coding,” as well as the μ -law and Lloyd–Max quantization schemes.

3.5. Analysis of Computational Complexity

On the encoding side, the calculation of the proposed quantization method involves four networks: the position parameter estimation network, the scale parameter estimation network, the parameter quantization network, and the measurement quantization network. The network structure of the position parameter estimation network and scale parameter estimation network are identical, as shown in Table 10. Similarly, the network structures of the parameter quantization and measurement quantization are identical, as shown in Table 11.

Table 10. Detailed network structures of parameter estimation.

Convolution Layer	Kernel Size	Stride	Input Channels	Output Channels	Activation Function
Conv1	10×1	1	1	3	LeakyReLU
Conv2	1×1	1	3	3	LeakyReLU
Conv3	1×1	1	3	1	-

Table 11. Detailed network structures of the quantization process.

Convolution Layer	Kernel Size	Stride	Input Channels	Output Channels	Activation Function
Conv1	1×1	1	1	6	LeakyReLU
Conv2	1×1	1	6	6	LeakyReLU
Conv3	1×1	1	6	1	g_3

The position and scale parameters are derived from partial measurements $Y_0 \in \mathbb{R}^{10 \times N_B}$. According to Table 10, convolutional layer 1 requires around $10 \times N_B \times 3$ multiplications and $10 \times N_B \times 3$ additions, convolutional layer 2 requires $3 \times N_B \times 3$ multiplications and $3 \times N_B \times 3$ additions, and convolutional layer 3 requires $3 \times N_B$ multiplications and $3 \times N_B$ additions. In total, location parameter estimation and scale parameter estimation need about $84N_B$ times multiplications, $84N_B$ times additions, and $12N_B$ times LeakyReLU operations.

According to Table 11, to quantize a parameter or measurement, the quantization network typically requires 48 times multiplications, 48 times additions, 12 times LeakyReLU operations, and one time g_3 operation. For the measurements’ matrix $Y \in \mathbb{R}^{M \times N_B}$, the

numbers of measurements and parameters that need to be quantized are $M \times N_B$ and $2N_B$, respectively. In total, it is necessary to compute $(2 + M)48N_B$ times multiplications, $(2 + M)48N_B$ times additions, $(2 + M)N_B$ times LeakyReLU, and $(2 + M)N_B$ times g_3 .

The proposed method requires approximately the same number of multiplication and addition operations, and the activation function to be computed only involves linear operations. Since addition is much faster than multiplication in practical operations, we only compared the number of multiplication operations. With an image size of 256×256 and a block size of 16×16 , the total number N_B of blocks would be 256. Assuming that the measurement rate of BCS is 0.1, each block obtains 26 measurements. Each measurement needs about 256 times multiplications and 255 times additions, and the calculation of measurements is about $26 \times 256N_B$ times multiplications and $26 \times 255N_B$ times additions. The proposed quantization method requires $660N_B$ times multiplications, $660N_B$ times additions, $156N_B$ times LeakyReLUs, and $12N_B$ times g_3 . Compared with the calculation for BCS measurements, the calculation of the proposed quantization process is about 9.92% of that of the BCS measurements.

4. Conclusions

In this paper, we propose a BCS measurement quantization method based on CNNs. The method uses a CNN based on measurements' CDF to map BCS measurements into quantized data following a uniform distribution. The block measurements' distribution parameters are quantized as the side information of the encoder. In dequantization, a CNN based on information correction is designed by using the correlations between the BCS measurements. The proposed method uses partial measurements to extract parameter features and jointly optimizes the CNNs of the quantization and dequantization. The experimental results indicate that the proposed quantization method has better performance than joint uniform quantization and entropy coding.

In the future, investigating the performance of our method under adverse conditions, such as noisy or blurred inputs, is critical for real-world deployment. While the CNNs have shown remarkable performance, their computational and memory requirements are points for consideration. Exploring lightweight CNN architectures, pruning techniques, and hardware acceleration strategies would be instrumental in making our method more practical for resource-constrained devices.

Author Contributions: Conceptualization, J.G.; Methodology, J.G., Q.C. and Z.W.; Software, J.G. and Q.C.; Validation, J.G., Q.C., W.Z. and Z.W.; Formal analysis, Q.C., W.Z. and Z.W.; Resources, J.G.; Data curation, J.G., Q.C., W.Z. and Z.W.; Writing—original draft, Q.C.; Writing—review & editing, J.G., W.Z. and Z.W.; Visualization, Q.C. and Z.W.; Project administration, J.G.; Funding acquisition, Z.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by [State Key Laboratory of Explosion Science and Safety Protection] grant number [QNKT23-07] And the APC was funded by [Beijing Institute of Technology].

Data Availability Statement: Data are contained within the article.

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

References

1. Candès, E.J.; Romberg, J.; Tao, T. Robust Uncertainty Principles: Exact Signal Frequency Information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509. [CrossRef]
2. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [CrossRef]
3. Candès, E.J. Compressive sampling. *Proc. Int. Congr. Math. Madrid* **2006**, *2006*, 1433–1452. Available online: http://www.ems-ph.org/books/show_abstract.php?proj_nr=39&vol=3&rank=69%5Cnpapers3://publication/doi/10.4171/022-3/69 (accessed on 1 January 2024).
4. Foucart, S.; Rauhut, H. *A Mathematical Introduction to Compressive Sensing*; Birkhäuser: Basel, Switzerland, 2013.
5. Wakin, M.B.; Laska, J.N.; Duarte, M.F.; Baron, D.; Sarvotham, S.; Takhar, D.; Kelly, K.F.; Baraniuk, R.G. An architecture for compressive imaging. In Proceedings of the 2006 International Conference on Image Processing, Atlanta, GA, USA, 8–11 October 2006; pp. 1273–1276. [CrossRef]

6. Romberg, J. Imaging via compressive sampling. *IEEE Signal Process. Mag.* **2008**, *25*, 14–20. [[CrossRef](#)]
7. Li, X.; Lan, X.; Yang, M.; Xue, J.; Zheng, N. Efficient lossy compression for compressive sensing acquisition of images in compressive sensing imaging systems. *Sensors* **2014**, *14*, 23398–23418. [[CrossRef](#)] [[PubMed](#)]
8. Goyal, V.K.; Fletcher, A.K.; Rangan, S. Compressive sampling and lossy compression. *IEEE Signal Process. Mag.* **2008**, *25*, 48–56. [[CrossRef](#)]
9. Mun, S.; Fowler, J.E. DPCM for quantized block-based compressed sensing of images. In Proceedings of the 2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO), Bucharest, Romania, 27–31 August 2012; pp. 1424–1428.
10. Wang, L.; Wu, X.; Shi, G. Binned progressive quantization for compressive sensing. *IEEE Trans. Image Process.* **2012**, *21*, 2980–2990. [[CrossRef](#)] [[PubMed](#)]
11. Yang, Y.; Sun, J.; Li, H.; Xu, Z. ADMM-CSNet: A Deep Learning Approach for Image Compressive Sensing. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 521–538. [[CrossRef](#)] [[PubMed](#)]
12. Zha, Z.; Yuan, X.; Wen, B.; Zhou, J.; Zhu, C. Group Sparsity Residual Constraint with Non-Local Priors for Image Restoration. *IEEE Trans. Image Process.* **2020**, *29*, 8960–8975. [[CrossRef](#)] [[PubMed](#)]
13. Zhou, S.; He, Y.; Liu, Y.; Li, C.; Zhang, J. Multi-Channel Deep Networks for Block-Based Image Compressive Sensing. *IEEE Trans. Multimed.* **2021**, *23*, 2627–2640. [[CrossRef](#)]
14. Cui, W.; Liu, S.; Jiang, F.; Zhao, D. Image Compressed Sensing Using Non-local Neural Network. *IEEE Trans. Multimed.* **2021**, *25*, 816–830. [[CrossRef](#)]
15. Shirazinia, A.; Chatterjee, S.; Skoglund, M. Joint source-channel vector quantization for compressed sensing. *IEEE Trans. Signal Process.* **2014**, *62*, 3667–3681. [[CrossRef](#)]
16. Leinonen, M.; Codreanu, M.; Juntti, M. Distributed Distortion-Rate Optimized Compressed Sensing in Wireless Sensor Networks. *IEEE Trans. Commun.* **2018**, *66*, 1609–1623. [[CrossRef](#)]
17. Leinonen, M.; Codreanu, M. Low-Complexity Vector Quantized Compressed Sensing via Deep Neural Networks. *IEEE Open J. Commun. Soc.* **2020**, *1*, 1278–1294. [[CrossRef](#)]
18. Sayood, K. *Introduction to Data Compression*; Elsevier: Amsterdam, The Netherlands, 2012. [[CrossRef](#)]
19. Farvardin, N.; Modestino, J.W. Optimum Quantizer Performance for a Class of Non-Gaussian Memoryless Sources. *IEEE Trans. Inf. Theory* **1984**, *30*, 485–497. [[CrossRef](#)]
20. Gish, H.; Pierce, J.N. Asymptotically Efficient Quantizing. *IEEE Trans. Inf. Theory* **1968**, *14*, 676–683. [[CrossRef](#)]
21. Cui, W.; Jiang, F.; Gao, X.; Zhang, S.; Zhao, D. An efficient deep quantized compressed sensing coding framework of natural images. In Proceedings of the MM'18: Proceedings of the 26th ACM international conference on Multimedia, Seoul Republic of Korea, Seoul, Republic of Korea, 22–26 October 2018; pp. 1777–1785. [[CrossRef](#)]
22. Chen, Z.; Hou, X.; Shao, L.; Gong, C.; Qian, X.; Huang, Y.; Wang, S. Compressive sensing multi-layer residual coefficients for image coding. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 1109–1120. [[CrossRef](#)]
23. Unde, A.S.; Deepthi, P.P. Rate–distortion analysis of structured sensing matrices for block compressive sensing of images. *Signal Process. Image Commun.* **2018**, *65*, 115–127. [[CrossRef](#)]
24. Goyal, V.K. Theoretical foundations of transform coding. *IEEE Signal Process. Mag.* **2001**, *18*, 9–21. [[CrossRef](#)]
25. Xue, L.; Qiu, W.; Wang, Y.; Wang, Z. An Image-Based Quantized Compressive Sensing Scheme Using Zadoff–Chu Measurement Matrix. *Sensors* **2023**, *23*, 1016. [[CrossRef](#)] [[PubMed](#)]
26. Wang, Z.; Jiang, Y.; Chen, S. Image parallel block compressive sensing scheme using DFT measurement matrix. *Multimed Tools Appl.* **2023**, *82*, 21561–21583. [[CrossRef](#)]
27. Kusetogullari, H.; Yavariabdi, A. Evolutionary multiobjective multiple description wavelet based image coding in the presence of mixed noise in images. *Appl. Soft Comput.* **2018**, *73*, 1039–1052. [[CrossRef](#)]
28. Kusetogullari, H. Dynamic Multiple Description Wavelet Based Image Coding Using Enhanced Particle Swarm Optimization. *IEEE J. Sel. Top. Signal Process.* **2015**, *9*, 125–138. [[CrossRef](#)]
29. Pharr, M.; Jakob, W.; Humphreys, G. *Physically Based Rendering: From Theory to Implementation*, 4th ed. Available online: <https://pbr-book.org/4ed/contents> (accessed on 1 January 2024).
30. Tamura, S.; Tateishi, M. Capabilities of a four-layered feedforward neural network: Four layers versus three. *IEEE Trans. Neural Networks* **1997**, *8*, 251–255. [[CrossRef](#)] [[PubMed](#)]
31. Maas, A.L.; Hannun, A.Y.; Ng, A.Y. Rectifier nonlinearities improve neural network acoustic models. In Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; p. 30.
32. Arbeláez, P.; Maire, M.; Fowlkes, C.; Malik, J. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 898–916. [[CrossRef](#)] [[PubMed](#)]
33. Bevilacqua, M.; Roumy, A.; Guillemot, C.; Morel, M.L.A. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In Proceedings of the 23rd British Machine Vision Conference, Surrey, UK, 3–7 September 2012. [[CrossRef](#)]
34. Kulkarni, K.; Lohit, S.; Turaga, P.; Kerviche, R.; Ashok, A. ReconNet: Non-iterative reconstruction of images from compressively sensed measurements. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016. [[CrossRef](#)]
35. Zeyde, R.; Elad, M.; Protter, M. On single image scale-up using sparse-representations. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes Artificial Intelligence Lecture Notes Bioinformatics)*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 711–730. [[CrossRef](#)]

36. Martin, D.; Fowlkes, C.; Tal, D.; Malik, J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In Proceedings of the Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, Vancouver, BC, Canada, 7–14 July 2001; pp. 416–423. [[CrossRef](#)]
37. Gray, R.M.; Neuhoff, D.L. Quantization. *IEEE Trans. Inf. Theory* **1998**, *44*, 2325–2383. [[CrossRef](#)]
38. Jacques, L.; Hammond, D.K.; Fadili, M.J. Stabilizing nonuniformly quantized compressed sensing with scalar companders. *IEEE Trans. Inf. Theory* **2013**, *59*, 7969–7984. [[CrossRef](#)]
39. Max, J. Quantizing for Minimum Distortion. *IRE Trans. Inf. Theory* **1960**, *6*, 7–12. [[CrossRef](#)]
40. Stuart, P. Lloyd, Least squares quantization in PCM. *IEEE Trans. Inf. Theory* **1982**, *28*, 129–136. Available online: <https://www.doc88.com/p-6631815894116.html> (accessed on 1 January 2024).
41. Mun, S.; Fowler, J.E. Block compressed sensing of images using directional transforms. In Proceedings of the 2009 16th IEEE International Conference on Image Processing (ICIP), Cairo, Egypt, 7–10 November 2009; pp. 3021–3024. [[CrossRef](#)]
42. Sun, L.; Hays, J. Super-resolution from internet-scale scene matching. In Proceedings of the 2012 IEEE International Conference on Computational Photography (ICCP), Seattle, WA, USA, 28–29 April 2012; pp. 1–12.
43. Huang, J.B.; Singh, A.; Ahuja, N. Single image super-resolution from transformed self-exemplars. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 5197–5206.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.