

Article

A General Rate-Distortion Optimization Method for Block Compressed Sensing of Images

Qunlin Chen, Derong Chen and Jiulu Gong *

School of Mechatronical Engineering, Beijing Institute of Technology, Beijing 100081, China;
3120170115@bit.edu.cn (Q.C.); cdrmy@263.net (D.C.)

* Correspondence: lujiugong@bit.edu.cn

Abstract: Block compressed sensing (BCS) is a promising technology for image sampling and compression for resource-constrained applications, but it needs to balance the sampling rate and quantization bit-depth for a bit-rate constraint. In this paper, we summarize the commonly used CS quantization frameworks into a unified framework, and a new bit-rate model and a model of the optimal bit-depth are proposed for the unified CS framework. The proposed bit-rate model reveals the relationship between the bit-rate, sampling rate, and bit-depth based on the information entropy of generalized Gaussian distribution. The optimal bit-depth model can predict the optimal bit-depth of CS measurements at a given bit-rate. Then, we propose a general algorithm for choosing sampling rate and bit-depth based on the proposed models. Experimental results show that the proposed algorithm achieves near-optimal rate-distortion performance for the uniform quantization framework and predictive quantization framework in BCS.

Keywords: data acquisition; compressed sensing; rate-distortion; optimal bit-depth; bit-rate; quantization

Citation: Chen, Q.; Chen, D.; Gong, J. A General Rate-Distortion Optimization Method for Block Compressed Sensing of Images. *Entropy* **2021**, *23*, 1354. <https://doi.org/10.3390/e23101354>

Academic Editors: Nithin Nagaraj, Kushal Shah and Jerry D. Gibson

Received: 23 August 2021

Accepted: 13 October 2021

Published: 16 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 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 (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Compressed sensing (CS) is a signal acquisition framework [1–3] that acquires the signal's measurements by linear projection at the sub-Nyquist rate. Unlike traditional image coding methods with high computational complexity, the CS-based image coding methods are suitable for resource-constrained application scenarios through simultaneous data acquisition and compression [4–7].

When CS is applied to an image, the large measurement matrix will cause enormous computation and memory space for the codec. Gan [8] has proposed a block compressed sensing (BCS) method to decrease the measurement matrix's size for images. BCS uses the same measurement matrix to measure the image block's raster scan vector, significantly reducing the sensor's calculation and transmission cost [9]. BCS processes each image block independently and supports parallel encoding, which can quickly obtain the image measurements. However, the real-valued CS measurements need to be combined with quantization and entropy encoder to output bitstreams for transmission or storage [10].

Although the uniform scalar quantization (SQ) is the most straightforward solution for quantizing CS measurements, it is inefficient in rate-distortion performance [6,11]. Therefore, some researchers have proposed different quantization schemes of CS measurements to enhance the rate-distortion performance. For example, Mun et al. [12] have combined the differential pulse-code modulation (DPCM) with uniform scalar quantization (DPCM-plus-SQ) for BCS measurements. The CS-based imaging system with DPCM-plus-SQ and the smoothed projected Landweber (SPL) reconstruction can compete with JPEG in some cases. Wang et al. [13] have proposed a progressive quantization framework of CS measurements, which is slightly better than JPEG in rate-distortion performance. Chen et al. [14] have proposed a progressive non-uniform quantization framework of CS

measurements using partial Hadamard matrix together with patch-based recovery algorithm, which can reach the rate-distortion performance of CCSDS-IDC (consultative committee for space data systems-image data compression standard). Chen et al. [15] have proposed a multi-layer residual coding framework for CS measurements, which combines prediction with the uniform SQ at the encoder. The framework predicts the CS measurements by using the reconstructed image of the encoded CS measurements and then uses the uniform SQ to quantify the residuals between the predicted measurement and the actual measurement, which can obtain a better rate-distortion performance than JPEG2000. Some other quantization schemes are also used for CS measurements [16–19], but they are rarely used for CS-based image coding because of their complexity.

In a CS-based image coding system, the bit-rate and reconstruction distortion depend on the CS sampling rate and quantization bit-depth, which have competition at a given bit-budget [20]. Therefore, the encoder needs to assign a CS sampling rate and a bit-depth by rate-distortion optimization (RDO).

Some researchers have discussed the optimization problem of CS sampling rate and bit-depth. Chen et al. [21] have proposed a bit-rate model and a relative distortion model to assign CS sampling rate and bit-depth for the CS-based coding system with uniform SQ. Jiang et al. [22] have presented a new Lagrange multiplier method to set quantization step size and number of measurements, whereas they do not consider the complexity. Liu et al. [23] have introduced a distortion model of compressed video sampling to optimize the sampling rate and bit-depth. The model's parameters need to be predicted from video features, which is not suitable for images. However, the above works only apply to uniform SQ scheme. To the knowledge of authors, little attention has been paid to optimize the CS sampling rate and bit-depth for other quantization schemes in BCS.

The purpose of this study is to propose an efficient RDO algorithm to assign the CS sampling rate and bit-depth for the most frequently used quantization schemes in BCS. The RDO algorithm should be designed with low complexity due to the simple coding process of CS. In this paper, we propose a bit-rate model and an optimal bit-depth model to avoid the high complexity of calculating rate-distortion cost. Firstly, we use generalized Gaussian distribution to describe the distribution of objects encoded by entropy encoder and then build a bit-rate model. Secondly, we find that there is a logarithmic relationship between the optimal quantized bit-depth and the bit-rate. Then, we propose an optimal bit-depth model and use a feed-forward neural network to train the model parameters. Finally, we introduce a general method for optimizing the CS sampling rate and bit-depth with the proposed models.

The remainder of this paper is structured as follows. We describe the problem of RDO in a CS-based imaging system in Section 2. Sections 3 and 4 discuss the proposed bit-rate model and the optimal bit-depth model. We propose an algorithm to assign CS sampling rate and bit-depth in Section 5. The experiment results and conclusions are drawn in Sections 6 and 7.

2. Problem Statement

CS theory states that a sparse signal can be recovered through its measurements obtained by linear random projection. Many natural images have a sparse representation in a wavelet transform domain or discrete cosine transform domain [24,25], so they can be acquired by CS. Suppose $x \in \mathbb{R}^{N \times 1}$ denotes a raster-scanned vector of an image block. The CS measurements vector $y \in \mathbb{R}^{M \times 1}$ of x can be acquired by the following expression:

$$y = \Phi x, \quad (1)$$

where $\Phi \in \mathbb{R}^{M \times N}$ ($M \ll N$) is a measurement matrix, and the sampling rate or measurement rate is $m = M/N$.

Since CS measurements are real, they need to be discretized by the quantization before entropy encoding. Based on the most commonly used quantization schemes of CS

measurements, the CS sampling model with quantization can be unified into the following expression:

$$y_Q = Q_b[f(y)] = Q_b[f(\Phi x)], \quad (2)$$

where y_Q is the quantized measurement vector, and it also stands for the input of entropy encoder. $Q_b: \mathbb{R} \rightarrow \mathbb{Q}$ denotes a uniform SQ operation for b-bit (applied element-wise in (2)), which maps $f(y)$ to the discrete alphabet \mathbb{Q} with $|\mathbb{Q}| = 2^b$. In this paper, we define $Q_b[f(y)] = \frac{f(y)}{\Delta}$, where $\Delta = \frac{f_{\max}(y) - f_{\min}(y)}{2^b}$ is the uniform quantization step size, $f_{\max}(y)$ and $f_{\min}(y)$ represent the maximum and minimum of $f(y)$, respectively. $f(\bullet)$ represents a reversible transform, which is used to change the distribution type of y . When the CS measurements are quantized by uniform SQ, $f(\bullet)$ is an identity transformation. When the CS measurements are quantized by A-law or μ -Law non-uniform quantization, $f(\bullet)$ is the law function [26]. When the CS measurements are quantized by prediction with uniform SQ, $f(\bullet)$ is the prediction function [12,17]. For example, in the DPCM-plus-SQ framework, $f(y^{(j)}) = y^{(j+1)} - y^{(j)}$, where $y^{(j)}$ represents the measurement vector of the j -th image block. The progressive quantization methods [13,14] are also prediction frameworks combined with uniform SQ. In the progressive quantization method, the CS measurements are divided into a basic layer and refinement layer for transmission after uniform SQ quantization with B bit. In the basic layer, all B significant bits of the quantization indexes are transmitted, so the prediction function is equivalent to the identity transformation. In the refinement layer, the least $B_1 < B$ significant bits of the quantization index are transmitted, so the dropped highest $B - B_1$ bit is equivalent to the predicted value, and the retained B_1 least significant bits are equivalent to the prediction residual.

The CS-based image coding system is composed of CS sampling, quantization, and entropy encoder [15]. The bitstream of the encoded image is used for transmission or storage. The decoder restores the bitstream to an image through the corresponding entropy decoder, dequantization, and CS reconstruction algorithm. Figure 1 shows the flow chart of the CS-based imaging system [10].

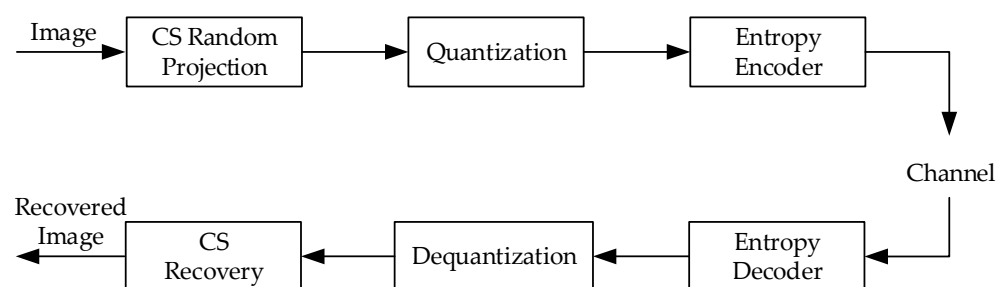


Figure 1. CS-based imaging system.

The average number of bits per pixel [21] of the encoded image can be calculated by the following expression:

$$R = mL, \quad (3)$$

where L is the average codeword length of the quantized CS measurements y_Q after entropy encoding.

There is a positive correlation between average codeword length and quantization bit-depth. When the bit-rate is constrained, sampling rate and quantization bit-depth have

a competitive relationship with each other. We can minimize the distortion to optimize the sampling rate and bit-depth for a given bit-rate R_{goal} , i.e.,

$$\arg \min_{m,b} D(m,b,\mathbf{X}) \text{ s.t. } R(m,b,\mathbf{X}) \leq R_{goal} \quad (4)$$

where $R(m,b,\mathbf{X})$ and $D(m,b,\mathbf{X})$, respectively, represent bit-rate and distortion of the image \mathbf{X} at the sampling rate m and the bit-depth b . The bit-rate $R(m,b,\mathbf{X})$ is the average number of bits per pixel of the encoded image, which can be obtained according to (3). Distortion refers to the dissimilarity between the reconstructed image $\hat{\mathbf{X}}$ and the original image \mathbf{X} . The distortion measures mainly include the mean square error (MSE), the peak signal-to-noise ratio (PSNR), and the structural similarity index measure (SSIM) [27]. The PSNR between the reconstructed image $\hat{\mathbf{X}}$ and the original image \mathbf{X} is used as a measure of distortion in our paper. The mathematical definition of PSNR is $PSNR = 10 \times \log_{10} (255^2 / MSE(\mathbf{X}, \hat{\mathbf{X}}))$, where $MSE(\mathbf{X}, \hat{\mathbf{X}})$ is the mean square error between the reconstructed image $\hat{\mathbf{X}}$ and the original image \mathbf{X} . The calculation of distortion and bit rate depends on the original image and decoded image, and the cost of obtaining decoded image is very expensive.

To avoid calculating the bit-rates and distortions, we first propose a new bit-rate model and an optimal bit-depth model. Then, we propose a general method to optimize the sampling rate and bit-depth for CS-based image coding. Figure 2 is the CS-based encoding system with RDO [21,23]. Our CS framework contains two CS processes. The first one is partial sampling, which aims to extract image features by a few CS measurements for RDO. The second one is to increase the number of CS measurements to achieve optimal sampling and compression by using the sampling rate optimized by RDO.

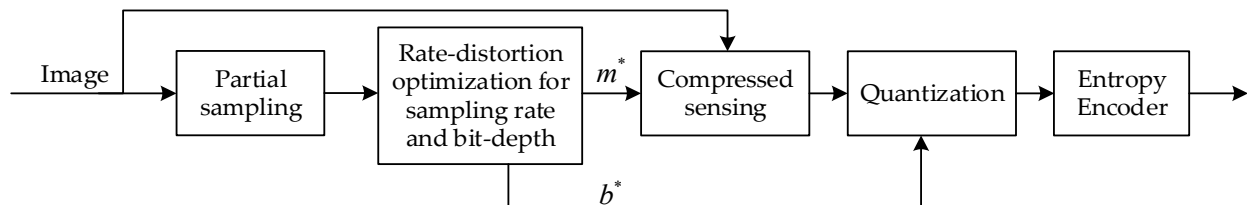


Figure 2. The CS-based coding system with RDO.

3. Bit-Rate Model

Based on (3), the bit-rate R depends on the average codeword length of the quantized CS measurements y_Q after entropy encoding. The average codeword length can be approximated by information entropy of y_Q before entropy encoding [28]. The information entropy is closely related to the distribution characteristics of the CS measurements, so we extract the distribution characteristics from the CS measurements of the first sampling to estimate the information entropy. However, the information entropy is only the lower boundary of the average codeword length. There is an error between the average codeword length and the information entropy estimated by a few measurements. Therefore, we modify the coefficients of the information entropy estimation model by fitting the offline data of the average codeword length and then take it as the average codeword length model.

3.1. Generalized Gaussian Distribution Model of the Quantized CS Measurements

According to (2), the quantized CS measurements can be considered to be obtained by $f(\bullet)$ and $Q_b[\bullet]$. $Q_b[\bullet]$ does not change the distribution type, while $f(\bullet)$ determines how to change the distribution type of CS measurements. The CS measurements using random Gaussian matrix obey Gaussian distribution [13]. When the structurally

random matrix (SRM) is used for CS measurement, the CS measurements corresponding to the first row of SRM are uniformly distributed, and the remaining CS measurements are Laplacian distributed with zero mean [10]. The distribution of DPCM predictive errors without conditioning on contexts is very close to a Laplace distribution [29]. The experiments of [30] show that the prediction errors of DPCM-plus-SQ satisfy the generalized Gaussian distribution. The Gauss distribution, the uniform distribution, and the Laplace distribution belong to the generalized Gauss distribution with specific shape parameters. In order to describe the distribution of CS measurements more generally, we use the generalized Gauss distribution to describe $f(y)$ and y_Q . The generalized Gaussian distribution density function with zero mean can be expressed as:

$$p_x(x|\sigma, \beta) = C(\beta, \sigma) \exp\left\{-[\alpha(\beta, \sigma)|x|]^\beta\right\}, \quad (5)$$

where $\alpha(\sigma, \beta) = \sigma^{-1} \left[\frac{\Gamma(3/\beta)}{\Gamma(1/\beta)} \right]^{\frac{1}{2}}$, $C(\sigma, \beta) = \frac{\beta \alpha(\sigma, \beta)}{2\Gamma(1/\beta)}$. σ is the standard deviation. β is the shape parameter, which determines the attenuation rate of the density function. $\Gamma(t) = \int_0^\infty e^{-u} u^{t-1} du$. The Laplace distribution and Gaussian distribution correspond to generalized Gaussian distribution when $\beta=1$ and $\beta=2$, respectively. Based on the generalized Gaussian distribution, the information entropy [31] of $f(y)$ can be estimated as

$$\begin{aligned} H &\approx - \int_{-\infty}^{+\infty} p_f(x|\beta, \sigma) \log_2(p_f(x|\beta, \sigma)) dx \\ &= -\log_2 \left[\frac{\beta \alpha(\beta, \sigma_f)}{2\Gamma(1/\beta)} \right] + \frac{1}{\beta \ln 2}, \\ &= \log_2(2\sigma_f) - \log_2 \left[\frac{\beta [\Gamma(3/\beta)]^{\frac{1}{2}}}{\Gamma[(1/\beta)]^{\frac{3}{2}}} \right] + \frac{1}{\beta \ln 2} \end{aligned} \quad (6)$$

where σ_f and β are the standard deviation and distribution shape parameters of $f(y)$, respectively. In Equation (2), y_Q is the discretization of $f(y)$ quantized by the quantization step size Δ , so the information entropy [31] of y_Q can be estimated by:

$$\begin{aligned} H_Q &\approx H - \log_2 \Delta \\ &= \log_2(2\sigma_f) - \log_2 \left[\frac{\beta [\Gamma(3/\beta)]^{\frac{1}{2}}}{\Gamma[(1/\beta)]^{\frac{3}{2}}} \right] + \frac{1}{\beta \ln 2} - \log_2 \Delta' \end{aligned} \quad (7)$$

3.2. Average Codeword Length Estimation Model

In Equation (7), σ_f and β are keys to estimating information entropy H . However, σ_f and β cannot be calculated directly because the CS measurements are unknown before the sampling rate and bit-depth are assigned. Since the number of CS measurements required for a high-quality reconstructed signal must satisfy a lower limit, the number of CS measurements used for compression will exceed the lower limit regardless of the goal bit-rate. Therefore, we can acquire a small number of CS measurements by the first sampling and then extract features for RDO.

The CS measurements with different sampling rates are subsets of the measurement population for the same image, so a small number of measurements can be used to estimate the features of measurements with a higher sampling rate. In this paper, m_0 represents the sampling rate of the first sampling.

The entropy-matching method is usually used to estimate the shape parameter of the generalized Gaussian distribution [31]. To simplify the estimation, we assume that there is a proportional relationship between $-\log_2 \left(\beta [\Gamma(3/\beta)]^{\frac{1}{2}} / \Gamma[(1/\beta)]^{\frac{3}{2}} \right) + 1/(\beta \ln 2)$ at different sampling rates for the same bit-depth, i.e.,

$$\begin{aligned} & -\log_2 \left(\beta [\Gamma(3/\beta)]^{\frac{1}{2}} / \Gamma[(1/\beta)]^{\frac{3}{2}} \right) + 1/(\beta \ln 2) \\ & \approx c \left(-\log_2 \left(\beta_0 [\Gamma(3/\beta_0)]^{\frac{1}{2}} / \Gamma[(1/\beta_0)]^{\frac{3}{2}} \right) + 1/(\beta_0 \ln 2) \right), \\ & \approx c \left(H_0 - \frac{1}{2} \log_2 (2\sigma_{f_0}) + \log \Delta \right) \end{aligned} \quad (8)$$

where H_0 and β_0 represent the information entropy and shape parameter of $f(y)$ at sampling rate m_0 and bit-depth b . c is an undetermined parameter.

Combined with the formula (8), the information entropy of y_Q can be estimated by the following expression:

$$H \approx \frac{1}{2} \log_2 (2\sigma_f) + c \left(H_0 - \frac{1}{2} \log_2 (2\sigma_{f_0}) + \log \Delta_{f_0} \right) - \log \Delta, \quad (9)$$

where σ_{f_0} is the standard deviation of $f(y)$ for measurements obtained by the first partial sampling, $\Delta_{f_0} = \frac{f_{\max}(y_0) - f_{\min}(y_0)}{2^b}$, y_0 is the measurement vector obtained by the first sampling.

In statistical theory, the statistic $s^2 = \frac{M}{M-1} \sigma^2$ of the sample variance is an unbiased estimation of the population's variance. Since the CS measurements with different sampling rates have the same population, we assume that the unbiased variance estimates of CS measurements at different measurement rates are approximately equal, that is:

$$\frac{M}{M-1} \sigma_f^2 \approx \frac{M_0}{M_0-1} \sigma_{f_0}^2, \quad (10)$$

where $M = \text{round}(mN^2)$, $M_0 = \text{round}(m_0N^2)$. The expression (10) can be converted into

$\sigma_f^2 \approx \left(\frac{m_0N^2}{(m_0N^2-1)} + \frac{-m_0}{(m_0N^2-1)} \times \frac{1}{m} \right) \sigma_{f_0}^2$. Then, we can obtain the following expression:

$$\begin{aligned} H & \approx \frac{1}{4} \log_2 (\sigma_f^2) + \frac{1}{2} \log_2 (2) + c \left(H_0 - \frac{1}{2} \log_2 (2\sigma_{f_0}) + \log \Delta_{f_0} \right) - \log \Delta \\ & \approx \frac{1}{4} \log_2 \left(\frac{m_0N^2}{(m_0N^2-1)} + \frac{-m_0}{(m_0N^2-1)} \times \frac{1}{m} \right) + \frac{1}{4} \log_2 (\sigma_{f_0}^2) \\ & \quad + \frac{1}{2} \log_2 (2) + c \left(H_0 - \frac{1}{2} \log_2 (2\sigma_{f_0}) + \log \Delta_{f_0} \right) - \log \Delta \end{aligned} \quad (11)$$

Since $\frac{-m_0}{(m_0N^2-1)} \rightarrow \frac{-1}{N^2}$ is very small, and the range of the sampling rate m is limited; the range of $\frac{-m_0}{(m_0N^2-1)} \times \frac{1}{m}$ is also very limited. Therefore, we use a simple linear function to estimate it:

$$\log_2 \left(\frac{m_0 N^2}{(m_0 N^2 - 1)} + \frac{-m_0}{(m_0 N^2 - 1)} \times \frac{1}{m} \right) \approx c' \frac{1}{m} + c'' \quad (12)$$

where c' and c'' are undetermined parameters. Moreover, we substitute the expression of quantization step into (11) to obtain the following expression:

$$\begin{aligned} & c \log_2 \Delta_{f_0} - \log_2 \Delta \\ &= c \log_2 \frac{f_{\max}(y_0) - f_{\min}(y_0)}{2^b} - \log_2 \frac{f_{\max}(y) - f_{\min}(y)}{2^b} \\ &= (1-c)b + c \log_2 (f_{\max}(y_0) - f_{\min}(y_0)) - \log_2 (f_{\max}(y) - f_{\min}(y)) \end{aligned} \quad (13)$$

We used the maximum $f_{\max}(y_0)$ and minimum $f_{\min}(y_0)$ of the first sampled CS measurements to predict the maximum $f_{\max}(y)$ and minimum $f_{\min}(y)$ of the CS measurements with sampling rate m . Therefore,

$$\begin{aligned} H \approx & c' \frac{1}{m} + c'' + \left(\frac{1}{4} - \frac{1}{4}c \right) \log_2 \sigma_{f_0}^2 + \log_2(2) \\ & + cH_0 + (1-c)b + (c-1) \log_2 (f_{\max}(y_0) - f_{\min}(y_0)) \end{aligned} \quad (14)$$

Based on (14), we replaced $(1-c)$, c' , c , $(\frac{1}{4} - \frac{1}{4}c)$, $(c-1)$ and $c'' + \log_2(2)$ with c_1 , c_2 , c_3 , c_4 , c_5 and c_6 , respectively, and establish a model for estimating the average codeword length as follows:

$$L \approx c_1 b + \frac{c_2}{m} + c_3 H_0 + c_4 \log_2 \sigma_{f_0}^2 + c_5 \log_2 (f_{\max}(y_0) - f_{\min}(y_0)) + c_6 \quad (15)$$

To improve the estimation accuracy of the average codeword length, we utilize the model coefficients $c_1 \sim c_6$ learned from offline data. Combining (3) with (15), we can establish the bit-rate model, as follows:

$$R \approx mL \approx m \left[c_1 b + \frac{c_2}{m} + c_3 H_0 + c_4 \log_2 \sigma_{f_0}^2 + c_5 \log_2 (f_{\max}(y_0) - f_{\min}(y_0)) + c_6 \right], \quad (16)$$

4. Optimal Bit-Depth Model

If we first predict the optimal bit-depth b^* , the sampling rate can be estimated based on the bit-rate model (16), i.e.,

$$m^* \approx \frac{R_{\text{goal}} - c_2}{c_1 b^* + C}, \quad (17)$$

where R_{goal} is the target bit-rate, $C = c_3 H_0 + c_4 \log_2 \sigma_{f_0}^2 + c_5 \log_2 (f_{\max}(y_0) - f_{\min}(y_0)) + c_6$ represents the feature of \mathbf{X} at bit-depth b^* . In this section, we propose an optimal bit-depth model, which can directly predict the optimal bit-depth for a given bit-rate.

4.1. Function Mapping Relationship between Optimal Bit-Depth and Bit-Rate

Chen et al. [15] tested the reconstruction performance of some images at different quantization bit-depths. They show that low quantization bit-depths can reconstruct high PSNRs at a low bit-rate, and the high quantization bit-depths can reconstruct high PSNRs at high bit-rate. However, they only give the fixed selection of quantization bit-depths for some bit-rates of all images, and do not give a method for selecting the optimal bit-depth. To find the relationship between the different quantization bit-depths and the PSNRs, we simulated eight test images, as shown in Figure 3. We obtain the optimal bit-depths of eight testing images by traversing different sampling rates ($m \in [0.05, 0.06, \dots, 0.4]$) and

different quantization bit-depths ($b \in [3, 4, \dots, 10]$) for CS-based coding systems with uniform SQ and DPCM-plus-SQ, as shown in Figures 4 and 5.

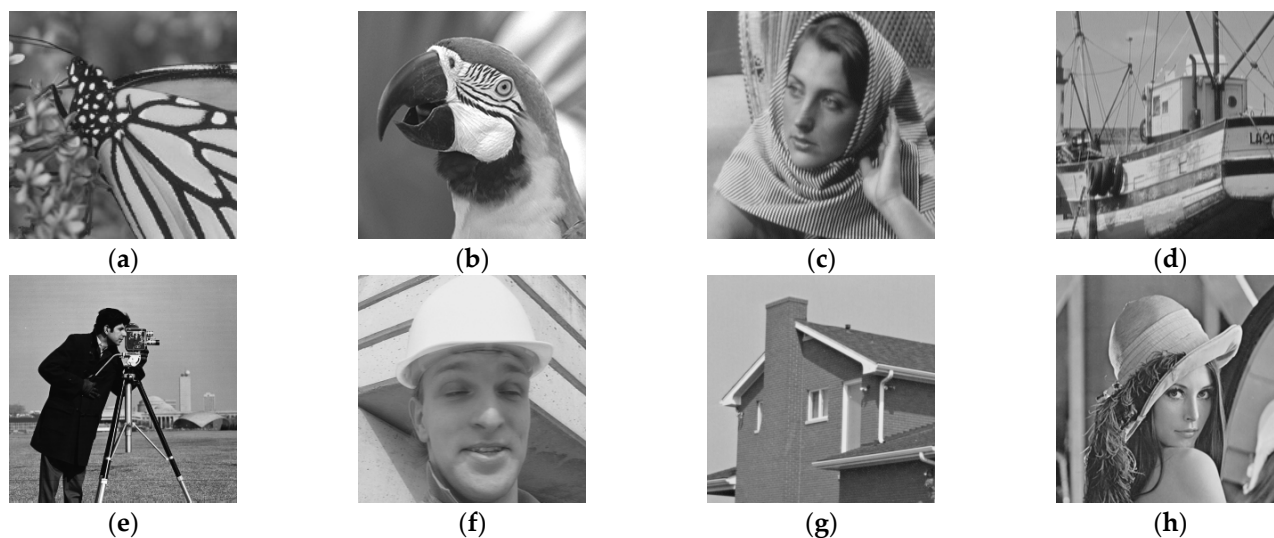


Figure 3. eight testing images. (a) Monarch; (b) Parrots; (c) Barbara; (d) Boats; (e) Cameraman; (f) Foreman; (g) House; (h) Lena.

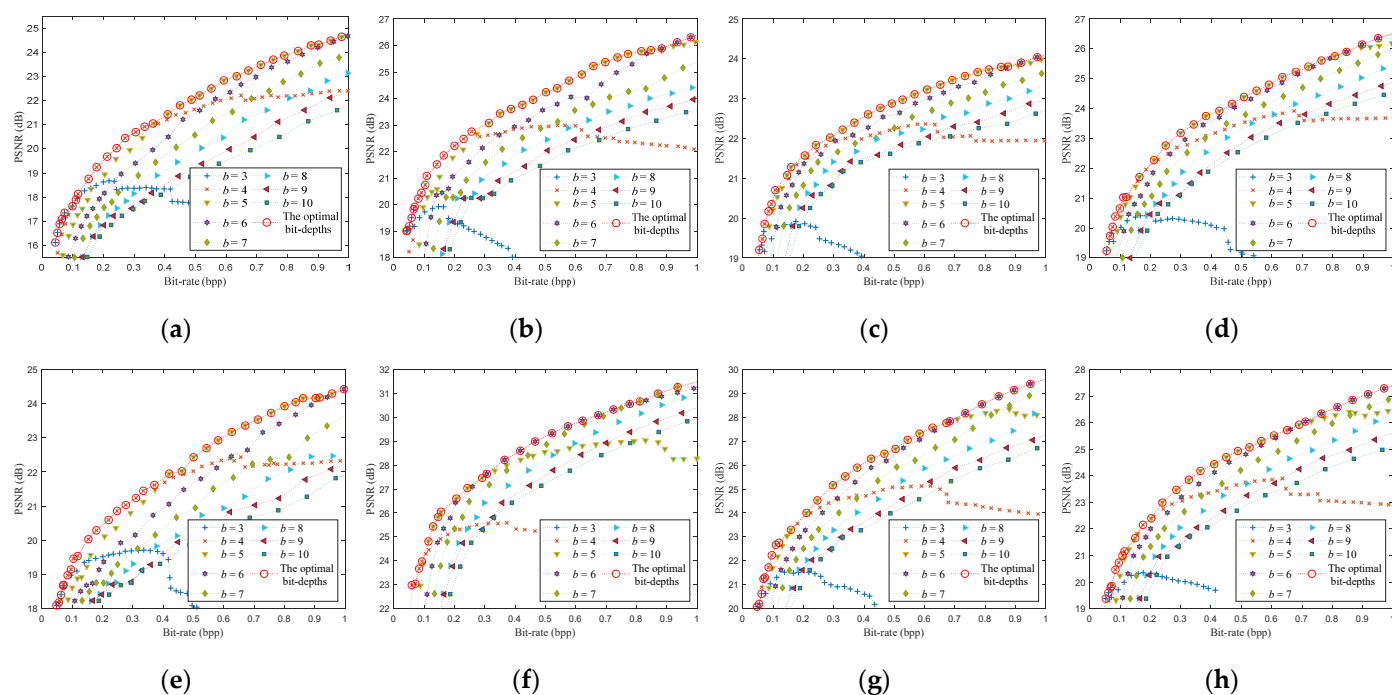


Figure 4. The optimal bit-depths of eight images for CS-based coding system with uniform SQ. (a) Monarch; (b) Parrots; (c) Barbara; (d) Boats; (e) Cameraman; (f) Foreman; (g) House; (h) Lena.

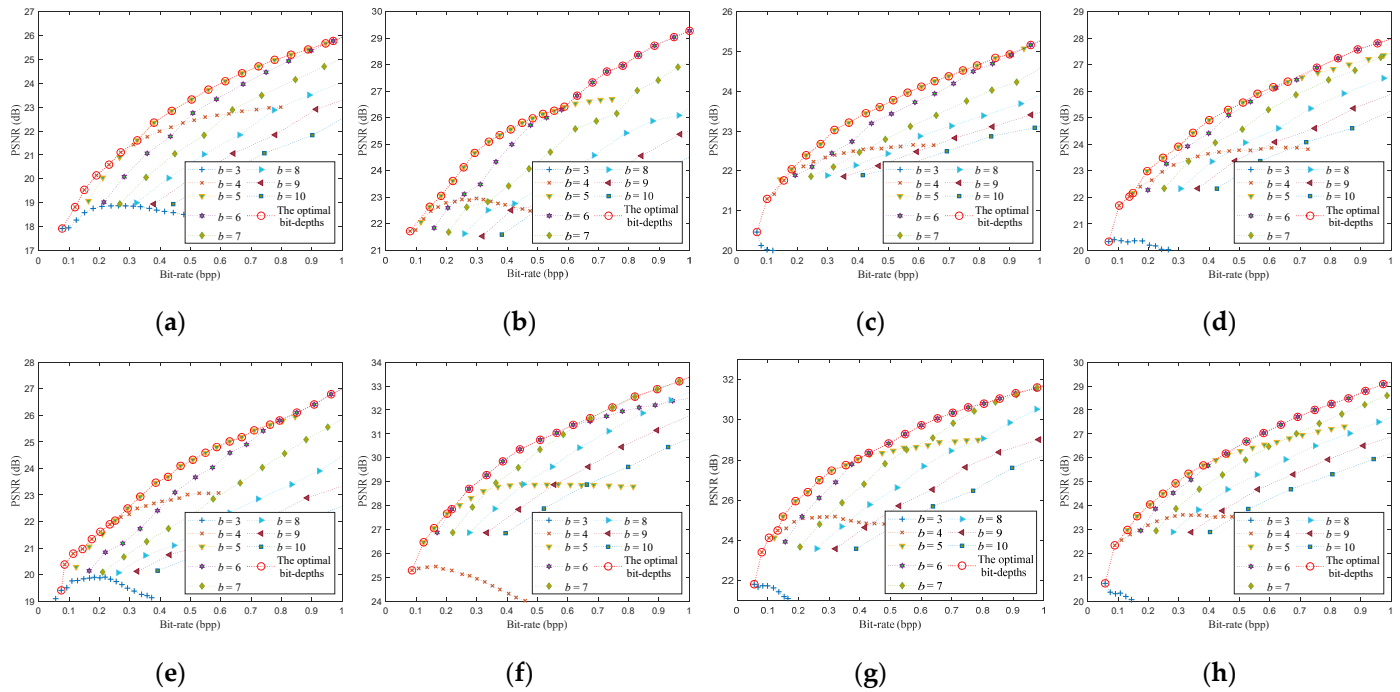


Figure 5. The optimal bit-depths of eight images for CS-based coding system with DPCM-plus-SQ. (a) Monarch; (b) Parrots; (c) Barbara; (d) Boats; (e) Cameraman; (f) Foreman; (g) House; (h) Lena.

It can be found from Figures 4 and 5 that the rate-distortion performance of the DPCM-plus-SQ framework (represents the CS-based coding system with DPCM-plus-SQ) is better than that of the uniform SQ framework (represents the CS-based coding system with uniform SQ), which indicates that the quantization scheme has a significant influence on the rate-distortion performance. However, the current rate-distortion optimization methods for CS are only suitable for a single uniform SQ framework. As far as we know, little attention has been paid to study the rate-distortion optimization method suitable for the prediction framework.

Although the optimal bit-depth of different quantization frameworks is different, Figures 4 and 5 have the following common characteristics: (1) low bit-depths have high PSNRs at low bit-rates, and high bit-depths have high PSNRs at high bit-rates. (2) The optimal bit-depth of almost all images is 4 when the bit-rate is around 0.1 bpp. (3) With the increase of bit-rate, the optimal bit-depth shows a nondecreasing trend. (4) The optimal bit-depth is the same in a bit-rate range, but the range is different for different images. There is a functional relationship between the optimal bit-depth and the bit-rate, which can be expressed as:

$$b_{best} = \begin{cases} 3 & 0 < R \leq r_1 \\ 4 & r_1 < R \leq r_2 \\ \vdots & \\ 8 & r_5 < R \leq r_6 \end{cases}, \quad (18)$$

where $r_1 \sim r_6$ are the endpoints of the bit-rate ranges. It can be found that the bit-rate range increases with the increases of b_{best} . The model (18) is equivalent to the following model:

$$b_{best} = [g(R)], \quad (19)$$

where $[\cdot]$ represents the rounding operation, and $g(R)$ represents a continuous function of the bit-rate. Since the optimal bit-depth increases with the increases of bit-rate, the

first-order derivative of $g(R)$ is required to be no less than 0. The increasing rate of the optimal bit-depth becomes slower with the increase of bit-rate, so the second-order derivative of $g(R)$ is required to be less than 0, that is,

$$\frac{\partial g(R)}{\partial^2 R} < 0 \quad \& \quad \frac{\partial g(R)}{\partial R} \geq 0, \quad (20)$$

Based on the above discussion, we set $g(R) = k_1 \ln(R) + k_2$. The model of the optimal bit-depth is established as follows:

$$\tilde{b}_{best} = [g(R)] = [k_1 \ln(R) + k_2], \quad (21)$$

where k_1 and k_2 are the model parameters, which are learned by a neural network in the Section 4.2. In order to collect offline data samples of k_1 and k_2 for the proposed neural network training, we establish the following optimization problem:

$$\arg \min_{k_1, k_2} \sum_i \omega_i \|b_{best}^{(i)} - [g(R^{(i)})]\|_1^q + \lambda \sum_i \|b_{best}^{(i)} - g(R^{(i)})\|_2^2, \quad (22)$$

where i is the sample index of the offline data. $b_{best}^{(i)}$ represents the actual value of the optimal bit depth of the i -th sample. ω_i represents the weight, which is the difference between the PSNR quantized with $b_{best}^{(i)}$ and the PSNR quantized with $[g(R^{(i)})]$ at the same bit-rate. In order to obtain the PSNR at the same bit rate, we perform linear interpolation on the sample data. The regularization term $\sum_i \|b_{best}^{(i)} - g(R^{(i)})\|_2^2$ guarantees the uniqueness of the solution. λ is a constant coefficient, which takes 0.01 in this work. We take $q = 10$, which avoids an error of more than 2 bits between the predicted value and the actual value.

In (22), the first item ensures the accuracy of the optimal bit-depth model, and the second item ensures the uniqueness of the model coefficient. Since it is difficult to deal with the gradient of the rounding operation, (22) cannot be solved by the traditional gradient-based optimization method. We use the particle swarm optimization algorithm [32,33] to optimize the problem (22). The number of particle swarm is 100 and iterated 300 times. In each iteration, 30 particle swarms in the population are randomly generated in the $[-0.5, 0.5]$ range of the optimal point.

Figures 6 and 7 show the fitted results of the model (21) for the uniform SQ framework and DPCM-plus-SQ framework, respectively. It can be seen that the fitted bit-depths are in good agreement with the actual bit-depths. The errors between the predicted value and the actual value are only one bit at most. The errors of one bit are mainly concentrated between the two adjacent optimal bit-depths, which has little difference on the PSNR for the two bit-depths.

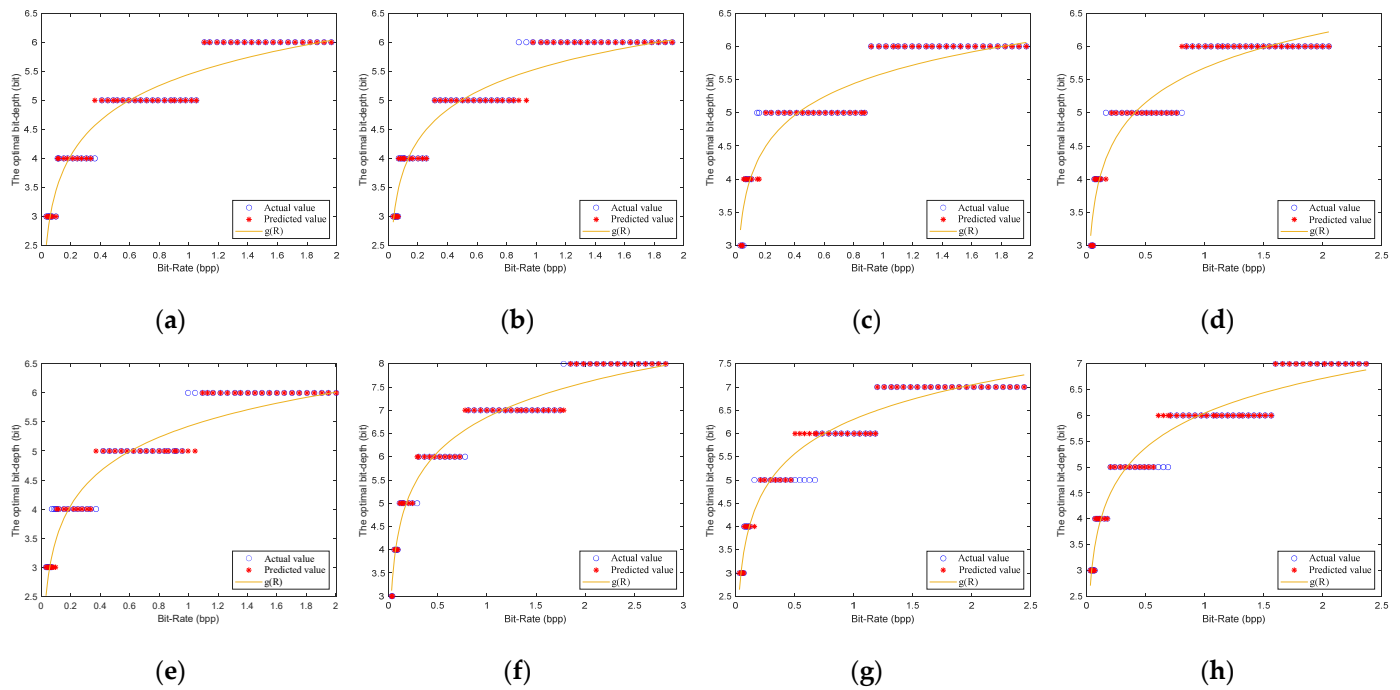


Figure 6. The predicted bit-depths of eight images for the SQ framework. (a) Monarch; (b) Parrots; (c) Barbara; (d) Boats; (e) Cameraman; (f) Foreman; (g) House; (h) Lena.

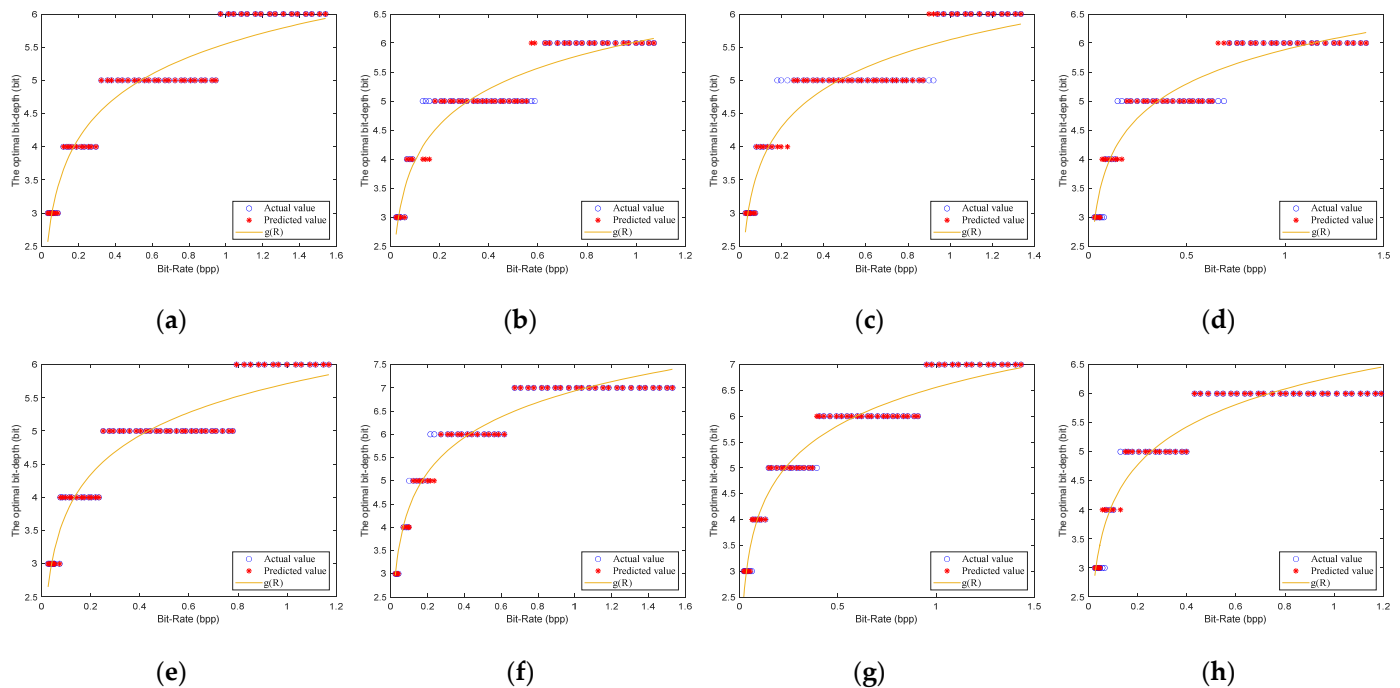


Figure 7. The predicted bit-depths of eight images for the DPCM-plus-SQ framework. (a) Monarch; (b) Parrots; (c) Barbara; (d) Boats; (e) Cameraman; (f) Foreman; (g) House; (h) Lena.

4.2. Model Parameter Estimation Based on Neural Network

It is challenging to design a function for estimating the model parameters accurately. Therefore, we use a four-layer feed-forward neural network [34,35] to learn the mapping relationship between the model parameters and image features rather than designing the function relationship by hand [36,37]. We can imagine that the model (21) would be

beneficial if the model parameters could be predicted based on some content features derived from the compressed sampled image. As model (21) is closely related to the bit-rate, we directly use the image features in the proposed bit-rate model as the characteristics of estimating the parameters. The image features of the proposed bit-rate model are σ_0^2 , H_0 , $f_{\max}(y_0)$, $f_{\min}(y_0)$. A finite set of real numbers usually needs to be quantized before calculating the information entropy. The optimal bit-depth of many images is low when the bit-rate is low, so we choose the information entropy $H_{0,bit=4}$ with a quantization bit-depth of 4 as a feature. Since the CS measurement of the image is sampled block by block, we take the image block as the video frame and design two image features according to the video features in reference [23]. For example, block difference (BD): the mean (and standard deviation) of the difference between the measurements of adjacent blocks, i.e., μ_{BD} and σ_{BD} . We also take the mean of measurements \bar{y}_0 as a feature.

We designed a network including an input layer of seven neurons and an output layer of two neurons to estimate the model parameters $[k_1, k_2]$, as shown in Formula (23) and Figure 8.

$$\begin{aligned} u_1 &= [\sigma_0^2, \bar{y}_0, f_{\max}(y_0), f_{\min}(y_0), \mu_{BD}, \sigma_{BD}, H_{0,bit=4}]^T \\ u_j &= g(W_{j-1}u_{j-1} + d_{j-1}), 2 \leq j < 4 \\ F &= W_{j-1}u_{j-1} + d_{j-1}, j = 4 \end{aligned} \quad (23)$$

where $g(v)$ is the sigmoid activation function, u_j is the input variable vector at the j -th layer, F is the parameters vector $[k_1, k_2]$. W_j, d_j are the network parameters learned from offline data. We take the mean square error (MSE) as the loss function.

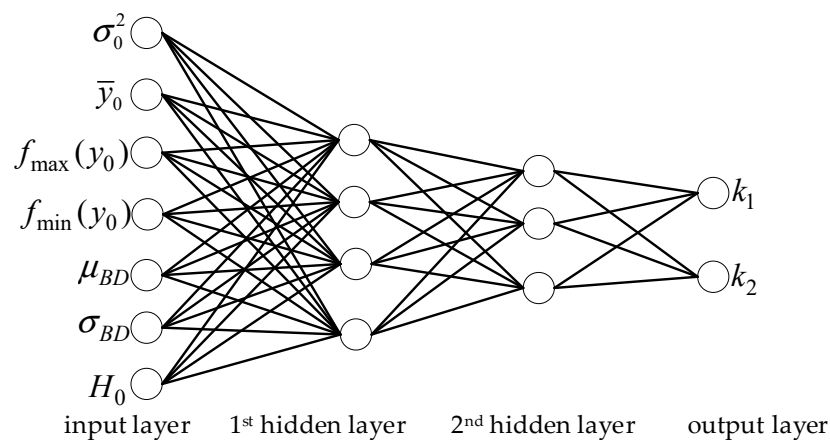


Figure 8. Four-layer feed-forward neural network model for the parameters.

5. A General Rate-Distortion Optimization Method for Sampling Rate and Bit-Depth

5.1. Sampling Rate Modification

The model (16) obtains the model parameters by minimizing the mean square error of all training samples. Although the total error is the smallest, there are still some samples with significant errors. To prevent excessive errors in predicting sampling rate, we propose the average codeword length boundary and sampling rate boundary.

5.1.1. Average Codeword Length Boundary

When the optimal bit-depth is determined, the average codeword length usually decreases with the sampling rate increase. Although the average codeword length of different images varies with the sampling rate, the variation is finite. Therefore, we design an average codeword length boundary.

As the information entropy H_0 is the input of the optimized sampling rate and is very close to the average codeword length L_0 with the sampling rate m_0 , we take H_0 as the reference of the average codeword length to estimate variation. The average codeword length variation is expressed as $L - H_0$. We only take the bit-depth and sampling rate as factors for influencing the upper and lower bound. According to model (16), we establish the upper and lower bound model of the average codeword length variation as follows:

$$\begin{aligned} L_u - H_0 &= a_1 b + \frac{a_2}{m} + a_3 \\ L_l - H_0 &= a_4 b + \frac{a_5}{m} + a_6 \end{aligned} \quad (24)$$

where L_u and L_l describe the upper and lower bounds of average codeword length, respectively. $a_1 \sim a_6$ are the model coefficients fitted by offline samples.

According to (17), we first estimate the sampling rate as

$$m^{(1)} = (R_{goal} - c_3) / (c_1 b^* + C) \quad (25)$$

The corresponding average codeword length is $L = R_{goal} / m^{(1)}$. Then, we calculate the upper $L_u = a_1 b^* + a_2 / m + a_3 + H_0$ and the lower bound $L_l = a_4 b^* + a_5 / m + a_6 + H_0$ based on (24). $L > L_u$ means that the sampling rate is too low; we should increase the sampling rate. So, we take the bit-rate model as $R = mL_u$, the sampling rate is updated to $m_u = (R_{goal} - a_2) / (H_0 + a_1 b^* + a_3)$; if $L < L_l$, we take the bit-rate model as $R = mL_l$, the sampling rate is updated to $m_l = (R_{goal} - a_5) / (H_0 + a_4 b^* + a_6)$. It is summarized as follows:

$$m^{(2)} = \begin{cases} m_u & \text{if } L > L_u \\ m_l & \text{if } L < L_l \\ m^{(1)} & \text{otherwise} \end{cases} \quad (26)$$

5.1.2. Sampling Rate Boundary

The average codeword length boundary uses the information entropy of partial measurements to restrict the estimated value of the average codeword length, so as to modify a sampling rate that is too large or too small. To modify the sampling rate more directly, we establish a linear boundary model of the sampling rate for different bit-depths as follows:

$$\begin{aligned} m_u' &= a_7 R + a_8 \\ m_l' &= a_9 R + a_{10} \end{aligned} \quad (27)$$

where R is the bit-rate, $a_7 \sim a_{10}$ are the model coefficients fitted by offline samples. When the assigned sampling rate exceeds the boundaries in (27), it will be modified by the following expression:

$$m^* = \begin{cases} m_u' & \text{if } m^{(2)} > m_u' \\ m_l' & \text{if } m^{(2)} < m_l' \end{cases} \quad (28)$$

5.2. Rate-Distortion Optimization Algorithm

Based on the proposed bit-rate model and the optimal bit-depth model, we propose an algorithm to assign the bit-depth and sampling rate for a given target bit-rate R_{goal} , as follows.

(1) Partial sampling.

The partial CS measurements are sampled with the sampling rate m_0 .

(2) Features extraction.

$\sigma_0^2, \bar{y}_0, f_{\max}(y_0), f_{\min}(y_0), \mu_{BD}, \sigma_{BD}, H_{0, \text{bit}=4}$ of partial measurements are calculated.

(3) The optimal bit-depth prediction.

The optimal bit-depth is predicted by $b_{best} = \lfloor k_1 \ln(R) + k_2 \rfloor$, where the model parameters are estimated by the trained network.

(4) Features extraction.

The partial measurements are quantized with bit-depth b^* , and then the information entropy H_0 is calculated.

(5) The optimal sampling rate prediction.

The optimal sampling rate is estimated by formula (25).

(6) Sampling rate modification

The sampling rate is updated according to the formula (26) and (28).

(7) CS sampling

The original image is acquired to obtain the remaining CS measurements by the supplementary sampling rate $m = m^* - m_0$.

(8) Quantization and entropy encoding.

All measurements are quantized and then coded by arithmetic coding.

5.3. Computational Complexity Analysis

The extra calculation of the sampling rate and quantization bit-depth optimization comes from three processes, namely feature extraction, the optimal bit-depth prediction, and the sampling rate estimation.

In feature extraction, the extra calculation mainly comes from $\sigma_0^2, \bar{y}_0, f_{\max}(y_0), f_{\min}(y_0), \mu_{BD}, \sigma_{BD}, H_{0,bit=4}$ of the measurements with sampling rate m_0 . The number of measurements is $m_0 \times N^2$. We assume that the calculation of one subtraction is equivalent to that of one addition. The calculation of \bar{y}_0 needs $m_0 \times N^2 - 1$ additions and one multiplication. The calculation of σ_0^2 needs $m_0 \times N^2 \times 2 - 1$ additions and $m_0 \times N^2 + 1$ multiplications. $f_{\max}(y_0)$ and $f_{\min}(y_0)$ need $(m_0 \times N^2 - 1) \times 2$ judgment operations. The calculation of block errors needs $m_0 \times (N^2 - B^2)$ additions. μ_{BD} needs $m_0 \times (N^2 - B^2) - 1$ additions and one multiplication. σ_{BD} needs $m_0 \times (N^2 - B^2) \times 2 - 1$ additions and $m_0 \times (N^2 - B^2) + 1$ multiplications. The extra calculation of $H_{0,bit=4}$ comes from quantization with bit-depth 4, which requires $m_0 \times N^2$ multiplications. The remaining calculation of $H_{0,bit=4}$ mainly comes from counting the number of symbols and calculating the entropy. The calculation of counting the number of symbols requires $m_0 \times N^2$ judgments, $m_0 \times N^2$ additions. As the maximum number of symbols is $2^{4+1} + 1 = 33$, the calculation of entropy needs 66 multiplications, 33 logarithms, and 33 additions at the most.

In the optimal bit-depth prediction, the calculation mainly comes from the neural network model. There are seven neurons in the input layer, two neurons in the output layer, four neurons in the first hidden layer, and three neurons in the second hidden layer. When the activation function is not considered, the calculation of the network includes $7 \times 4 + 4 \times 3 + 3 \times 2 = 46$ multiplications and $6 \times 4 + 4 + 3 \times 3 + 3 + 2 \times 2 + 2 = 46$ additions. In the sampling rate estimation, the amount of calculation mainly comes from the calculation of (25) and (26).

Compared with the computations of the CS measurements, a fixed number of operations can be ignored. The extra calculation includes $m_0 \times N^2 \times 8$ additions, $m_0 \times N^2 \times 3$ multiplications, and $m_0 \times N^2 \times 3$ judgments. Assuming that two additions are needed for one judgment operation, the total amount of additional computation requires $m_0 \times N^2 \times 14$ additions and $m_0 \times N^2 \times 3$ multiplications.

The computations of all CS measurements requires $m \times N^2 \times B^2$ multiplications and $m \times N^2 \times (B^2 - 1)$ additions. B is the size of the image block, which is at least 16. When

$B = 16$, the optimization process needs to increase $3/B^2 \times (m_0/m) \leq 3/256 \approx 1.17\%$ multiplications and $14/(B^2 - 1) \times (m_0/m) \leq 14/255 \approx 5.49\%$ additions. In computer operations, the amount of calculation of addition is at least ten times faster than multiplication. The computations of rate-distortion optimization will not exceed 2% of the computations of the partial measurements. Furthermore, with the increase of image block size or sampling rate m , the percentage of computation in the optimization process will be further reduced.

6. Experimental Results

The proposed method is tested on some images for the DPCM-plus-SQ framework and uniform SQ framework, respectively. The model parameters are obtained by offline training some images of the BSDS500 database [38]. Several images, including eight images (shown in Figure 3), and the BSD68 dataset [39], are tested in our simulations. We take 100 images randomly selected from the BSDS500 database as the training set and the BSD68 dataset (68 images) as the test set. Since the size of the images varies, the images were cropped to a size of 256×256 from the center. All the numerical experiments are performed via MATLAB (R2018b) on a Windows 10 (64 bit) platform with an Intel Core i5-8300H 2.30 GHz processor and 16 GB of RAM.

6.1. Model Parameters Estimation

To obtain the model parameters of the proposed bit-rate model and the optimal bit-depth model, we take 100 images from the BSDS500 database [38] to collect training samples. The training data adopts the way of traversing bit-depths and sampling rates. The bit-depths include $\{3, 4, \dots, 10\}$; the set of sampling rate includes 37 samples in $\{0.04, 0.05, \dots, 0.4\}$ and 7 samples in $\{3/256, 4/256, \dots, 9/256\}$. If the average codeword length compressed by entropy encoding is greater than the quantized bit-depth, we take the average codeword length equal to the quantized bit-depth. One image collects 352 samples of the average codeword length and PSNR. The image block size adopts the optimal size of the corresponding quantization method, in which the DPCM quantization framework uses 16×16 blocks and uniform quantization uses 32×32 blocks. The orthogonal random Gaussian matrix is used for BCS sampling in this work. The entropy encoder adopts arithmetic coding [40]. In the decoder, the SPL-DWT algorithm [41] is used for image reconstruction. We take the first partial sampling rate $m_0 = 0.05$.

We use the least-square method to fit the model (15). Table 1 shows the trained parameters for DPCM-plus-SQ framework and uniform SQ framework. To quantify the accuracy of the fitting, we calculate the mean square error (MSE) and Pearson correlation coefficient (PCC) [42] between the actual value and predicted value. The closer the PCC is to 1, the better the fit of the model. The closer the MSE is to 0, the better the fit of the model. For the DPCM-plus-SQ framework, the MSE and PCC are 0.022 and 0.995, respectively. For the uniform SQ framework, the MSE and PCC are 0.027 and 0.996, respectively. Table 1 shows that the proposed model (15) can well describe the relationship between average codeword length L and bit-depth, sampling rate, and image features. The results show that model (15) can well describe the relationship between the average codeword length, sampling rate, and bit-depth.

Table 1. Parameters of the fitted model (15).

| Quantization Framework | c_1 | c_2 | c_3 | c_4 | c_5 | c_6 | PCC | MSE |
|------------------------|--------------------------|-------------------------|--------------------------|-------------------------|--------|---------|-------|-------|
| DPCM-plus-SQ | -3.0927×10^{-1} | 1.9128×10^{-2} | -1.6845×10^{-1} | 1.6592×10^{-1} | 1.3467 | -1.1718 | 0.995 | 0.022 |
| uniform SQ | -2.0660×10^{-1} | 6.5594×10^{-3} | -2.0673×10^{-1} | 2.3831×10^{-1} | 1.2761 | -1.9910 | 0.996 | 0.027 |

The optimal bit-depth model depends on the model parameters estimated by the proposed neural network. The samples of the model parameters are obtained by solving the

problem (22) and then are used to train the neural network. Due to the random initialization of neural network parameters, the prediction performances of the different trained networks are different. The best network from several trained networks is chosen to estimate the parameters of the proposed optimal bit-depth model. Table 2 shows the prediction performance of the optimal bit-depth model in the training set image and test set.

Table 2. Performances of the training set and test set for the optimal bit-depth model.

| Quantization Framework | DPCM-Plus-SQ | | Uniform SQ | |
|---------------------------------|--------------|----------|--------------|----------|
| | Training Set | Test Set | Training Set | Test Set |
| Accuracy (%) | 80.7 | 70.7 | 76.5 | 70.4 |
| Percentage of one-bit error (%) | 19.3 | 29.0 | 23.5 | 29.3 |
| Sum (above) (%) | 100 | 99.7 | 100 | 99.7 |

As shown in Table 2, for DPCM-plus-SQ framework, 80.7% and 70.7% are the accuracies of predicting the optimal bit-depth in the training set (BSDS500) and the test set (BSD68), respectively. For uniform SQ framework, 76.5% and 70.4% are the accuracies of predicting the optimal bit-depth in the training set (BSDS500) and the test set (BSD68), respectively. In the training set, the differences between the optimal bit-depth and the predicted bit-depth are no more than one bit. In the test set, 99.7% of the samples have a difference of no more than one bit between the optimal bit-depth and the predicted bit-depth. In most cases, the influence of 1-bit error on PSNR is limited, so it is effective to utilize a neural network to learn the optimal bit-depth model parameters.

In the training set, 100 images have 100 the average codeword length curves. We take the upper five curves and the lower five curves as the training samples of the model (24). The parameters are fitted offline by the least square method. We obtain $a_1 = -8.4564 \times 10^{-3}$, $a_2 = 1.8272 \times 10^{-2}$, $a_3 = -1.5871 \times 10^{-1}$, $a_4 = -6.7478 \times 10^{-2}$, $a_5 = 1.4306 \times 10^{-2}$, $a_6 = -1.8052 \times 10^{-1}$ for the DPCM-plus-SQ framework, and $a_1 = 4.5857 \times 10^{-2}$, $a_2 = 8.1957 \times 10^{-3}$, $a_3 = -2.1012 \times 10^{-1}$, $a_4 = 1.0226 \times 10^{-1}$, $a_5 = 1.6633 \times 10^{-3}$, $a_6 = -9.0052 \times 10^{-1}$ for the uniform SQ framework.

In the training set, we take the maximum and minimum sampling rate corresponding to the given bit-rates as the training sample of the model (27). The parameters are obtained by the least-square method. Through experiments, we found that the optimized sampling rates beyond the boundary are mainly near the low bit-rate of 0.1–0.3, and the corresponding optimal bit-depths are mostly 4 bit or 5 bit. So, we impose boundary constraints on the sampling rates when the optimal bit-depths are 4 and 5. The parameters are fitted offline by the least square method. For DPCM-plus-SQ framework with bit-depth of 4, we obtain $a_7 = 4.9164 \times 10^{-1}$, $a_8 = -7.1258 \times 10^{-3}$. For DPCM-plus-SQ framework with bit-depth 5, we obtain $a_7 = 3.4874 \times 10^{-1}$, $a_8 = -6.1371 \times 10^{-3}$. For uniform SQ framework with bit-depth 4, we obtain $a_7 = 3.3181 \times 10^{-1}$, $a_8 = -1.3050 \times 10^{-3}$. For uniform SQ framework with bit-depth 5, we obtain $a_7 = 2.3433 \times 10^{-1}$, $a_8 = 2.3347 \times 10^{-3}$.

6.2. Rate-Distortion Optimization Performance

To verify the accuracy of the bit-rate model, we tested the BSD68 dataset and eight images in Figure 3, respectively. We first use the proposed algorithm to assign the sampling rate and bit-depth for target bit-rates, including 0.1, ... 1 bit per pixel (bpp). Then, we calculate actual bit-rates and PSNRs of the reconstructed image for the estimated sampling rate and bit-depth. Tables 3 and 4 show the optimized bit-rate of BSD68 for the uniform SQ framework and DPCM-plus-SQ framework, respectively. The absolute error percentage denotes the percentage of the absolute error in the target bit-rate, where the absolute error is the absolute of the difference between target bit-rate and actual bit-rate.

Table 3. Comparison of actual bit-rates with target bit-rates for the uniform SQ framework.

| Image | Target Bit-Rate (bpp) | | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 |
|----------------|--|---------|-------|-------|-------|-------|-------|
| BSD68 test set | Actual bit-rate | Maximum | 0.110 | 0.218 | 0.327 | 0.427 | 0.523 |
| | | Minimum | 0.087 | 0.181 | 0.268 | 0.368 | 0.467 |
| | | Average | 0.099 | 0.203 | 0.305 | 0.406 | 0.503 |
| | Average of absolute error percentage (%) | | 3.23 | 2.91 | 2.29 | 2.33 | 1.78 |
| Image | Target Bit-Rate (bpp) | | 0.6 | 0.7 | 0.8 | 0.9 | 1 |
| BSD68 test set | Actual bit-rate | Maximum | 0.619 | 0.727 | 0.831 | 0.934 | 1.037 |
| | | Minimum | 0.565 | 0.664 | 0.771 | 0.831 | 0.916 |
| | | Average | 0.603 | 0.706 | 0.805 | 0.901 | 0.999 |
| | Average of absolute error percentage (%) | | 1.67 | 1.87 | 1.65 | 1.86 | 1.86 |

Table 4. Comparison of actual bit-rates with target bit-rates for the DPCM-plus-SQ framework.

| Image | Target Bit-Rate (bpp) | | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 |
|----------------|--|---------|-------|-------|-------|-------|-------|
| BSD68 test set | Actual bit-rate | Maximum | 0.108 | 0.219 | 0.319 | 0.424 | 0.533 |
| | | Minimum | 0.090 | 0.187 | 0.274 | 0.366 | 0.457 |
| | | Average | 0.101 | 0.200 | 0.299 | 0.399 | 0.499 |
| | Average of absolute error percentage (%) | | 3.17 | 2.75 | 2.30 | 2.09 | 2.03 |
| Image | Target Bit-Rate (bpp) | | 0.6 | 0.7 | 0.8 | 0.9 | 1 |
| BSD68 test set | Actual bit-rate | Maximum | 0.636 | 0.741 | 0.848 | 0.954 | 1.063 |
| | | Minimum | 0.548 | 0.646 | 0.737 | 0.829 | 0.922 |
| | | Average | 0.597 | 0.695 | 0.793 | 0.891 | 0.989 |
| | Average of absolute error percentage (%) | | 2.18 | 2.23 | 2.20 | 2.33 | 2.26 |

As shown in Table 3, the bit-rate average absolute error percentages of BSD68 are between 1.65% and 3.23%, which indicates that the proposed bit-rate model is useful for uniform SQ. As shown in Table 4, the bit-rate average absolute error percentages of BSD68 are between 2.09% and 3.17%, which indicates that the proposed bit-rate model is useful for DPCM-plus-SQ.

Tables 5 and 6 show the actual bit-rate of the eight testing images for uniform SQ and DPCM-plus-SQ, respectively. The results exhibit that actual bit-rates are very close to the target bit-rates.

Table 5. The actual bit-rates of eight images for the uniform SQ framework.

| Target Bit-Rate (bpp) | Actual Bit-Rate (bpp) | | | | | | | |
|-----------------------|-----------------------|---------|---------|-------|-----------|---------|-------|-------|
| | Monarch | Parrots | Barbara | Boats | Cameraman | Foreman | House | Lena |
| 0.1 | 0.102 | 0.102 | 0.098 | 0.106 | 0.101 | 0.096 | 0.105 | 0.100 |
| 0.2 | 0.203 | 0.207 | 0.204 | 0.207 | 0.204 | 0.208 | 0.204 | 0.205 |
| 0.3 | 0.305 | 0.305 | 0.307 | 0.317 | 0.312 | 0.308 | 0.310 | 0.307 |
| 0.4 | 0.408 | 0.402 | 0.412 | 0.416 | 0.419 | 0.415 | 0.414 | 0.412 |
| 0.5 | 0.493 | 0.506 | 0.504 | 0.513 | 0.518 | 0.512 | 0.511 | 0.507 |
| 0.6 | 0.590 | 0.606 | 0.606 | 0.615 | 0.623 | 0.609 | 0.612 | 0.609 |
| 0.7 | 0.694 | 0.713 | 0.713 | 0.725 | 0.731 | 0.712 | 0.720 | 0.705 |
| 0.8 | 0.790 | 0.800 | 0.802 | 0.827 | 0.837 | 0.812 | 0.817 | 0.805 |
| 0.9 | 0.884 | 0.905 | 0.905 | 0.918 | 0.897 | 0.917 | 0.923 | 0.910 |
| 1.0 | 0.981 | 1.003 | 1.003 | 1.019 | 1.030 | 1.017 | 1.024 | 1.007 |

Table 6. The actual bit-rates of eight images for the DPCM-plus-SQ framework.

| Target Bit-Rate (bpp) | Actual Bit-Rate (bpp) | | | | | | | |
|--------------------------|-----------------------|---------|---------|-------|-----------|---------|-------|-------|
| | Monarch | Parrots | Barbara | Boats | Cameraman | Foreman | House | Lena |
| 0.1 | 0.102 | 0.098 | 0.099 | 0.103 | 0.101 | 0.095 | 0.098 | 0.102 |
| 0.2 | 0.197 | 0.189 | 0.201 | 0.210 | 0.201 | 0.202 | 0.190 | 0.187 |
| 0.3 | 0.303 | 0.292 | 0.295 | 0.308 | 0.302 | 0.296 | 0.291 | 0.289 |
| 0.4 | 0.404 | 0.392 | 0.387 | 0.410 | 0.404 | 0.398 | 0.407 | 0.380 |
| 0.5 | 0.504 | 0.487 | 0.490 | 0.511 | 0.500 | 0.494 | 0.505 | 0.477 |
| 0.6 | 0.603 | 0.578 | 0.593 | 0.611 | 0.596 | 0.598 | 0.610 | 0.569 |
| 0.7 | 0.705 | 0.671 | 0.691 | 0.703 | 0.695 | 0.706 | 0.703 | 0.668 |
| 0.8 | 0.799 | 0.756 | 0.779 | 0.811 | 0.794 | 0.792 | 0.804 | 0.759 |
| 0.9 | 0.899 | 0.864 | 0.881 | 0.902 | 0.896 | 0.895 | 0.898 | 0.848 |
| 1.0 | 1.014 | 0.957 | 0.969 | 0.993 | 0.989 | 0.996 | 1.004 | 0.940 |

To test the validity of the optimal bit-depth model, we compare the predicted optimal bit-depth with the best bit-depths by traversing different bit-depths and different sampling rates, as shown in Tables 7 and 8.

Table 7. Performance of the optimal bit-depth model for the uniform SQ framework.

| Image | Target Bit-Rate (bpp) | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 |
|-------------------|------------------------------|-------|-------|-------|-------|-------|
| BSD68 test set | Optimal percentage (%) | 91.18 | 64.71 | 77.94 | 77.94 | 64.71 |
| | One-bit error percentage (%) | 8.82 | 35.29 | 22.06 | 22.06 | 35.29 |
| | Sum of the above (%) | 100 | 100 | 100 | 100 | 100 |
| | Average PSNR error (dB) | −0.04 | −0.13 | −0.12 | −0.06 | −0.08 |
| Image | Target Bit-Rate (bpp) | 0.6 | 0.7 | 0.8 | 0.9 | 1 |
| BSD68 test set | Optimal percentage (%) | 63.24 | 54.41 | 60.29 | 57.35 | 64.71 |
| | One-bit error percentage (%) | 36.76 | 45.59 | 39.71 | 41.18 | 33.82 |
| | Sum of the above (%) | 100 | 100 | 100 | 98.53 | 98.53 |
| | Average PSNR error (dB) | −0.09 | −0.10 | −0.08 | −0.11 | −0.07 |

Table 8. Performance of the optimal bit-depth model for the DPCM-plus-SQ framework.

| Image | Target Bit-Rate (bpp) | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 |
|-------------------|------------------------------|-------|-------|-------|-------|-------|
| BSD68 test set | Optimal percentage (%) | 82.35 | 58.82 | 79.41 | 79.41 | 76.47 |
| | One-bit error percentage (%) | 17.65 | 41.18 | 20.59 | 20.59 | 23.53 |
| | Sum of the above (%) | 100 | 100 | 100 | 100 | 100 |
| | Average PSNR error (dB) | −0.29 | −0.21 | −0.16 | −0.06 | −0.07 |
| Image | Target Bit-Rate (bpp) | 0.6 | 0.7 | 0.8 | 0.9 | 1 |
| BSD68 test set | Optimal percentage (%) | 64.71 | 70.59 | 60.29 | 55.88 | 63.24 |
| | One-bit error percentage (%) | 35.29 | 29.41 | 38.24 | 42.65 | 35.29 |
| | Sum of the above (%) | 100 | 100 | 98.53 | 98.53 | 98.53 |
| | Average PSNR error (dB) | −0.07 | −0.09 | −0.11 | −0.14 | −0.11 |

In Tables 7 and 8, the optimal percentage shows the percentage of images whose predicted bit-depth is consistent with the actual best bit-depth. The one-bit error percentage is the percentage of images with the one-bit error between the predicted bit-depth and the actual best bit-depth. We encode and decode the images according to the predicted parameters (sampling rate and bit-depth) and calculate the bit-rate and PSNR. The PSNR error is the PSNR minus the maximum PSNR, where the PSNRs are obtained by the nearest interpolation method for a bit-rate.

In Table 7, the sum of optimal bit-depth and one-bit error bit-depth obtain a percentage of between 98.53% and 100% for the SQ framework. When the target bit-rates are

0.1~0.8 bpp, the sum of the optimal bit-depth and one-bit error bit-depth percentage is 100%. When the target bit-rates are 0.9 and 1 bpp, the sum of optimal bit-depth and one-bit error bit-depth percentage is 98.53%. As the difference of PSNR between different bit-depths is small at the high target bit-rates, there is an error in estimating the bit-depth. Although only 54.41% to 91.18% of the predicted bit-depths are consistent with the optimal bit-depth, the average PSNR errors are between 0.04 dB and 0.013 dB, which shows that the error of predicted bit-depth has little influence on the reconstruction performance.

In Table 8, the sum of optimal bit-depth and one-bit error bit-depth obtain a percentage of between 98.53% and 100% for the DPCM-plus-SQ framework. When the target bit-rates are 0.1~0.7 bpp, the sum of optimal bit-depth and one-bit error bit-depth obtain a percentage of 100%. When the target bit-rates are 0.8~1 bpp, the sum of optimal bit-depth and one-bit error bit-depth obtains a percentage of 98.53%. Although only 55.88% to 82.35% of the predicted bit-depths are consistent with the optimal bit-depth, the average PSNR errors are between 0.07 dB and 0.029 dB, which shows that the error of predicted bit-depth has little influence on the reconstruction performance.

To demonstrate the performance of the proposed method in detail, we give the optimized rate-distortion curves of the eight testing images, as shown in Figures 9 and 10. We first encode the image for the bit-rates according to the optimized sampling rates and bit-depths, then calculate the PSNRs of the reconstructed image to obtain the rate-distortion curve. All bit-rates and PSNRs obtained by traversing different sampling rates and bit-depths are also shown in Figures 9 and 10. As far as we know, the optimization of sampling rate and bit-depth in the CS-based coding system is mainly focused on the uniform SQ framework, so we compared the proposed method with the latest methods [21] for the uniform SQ framework, as shown in Figure 9.

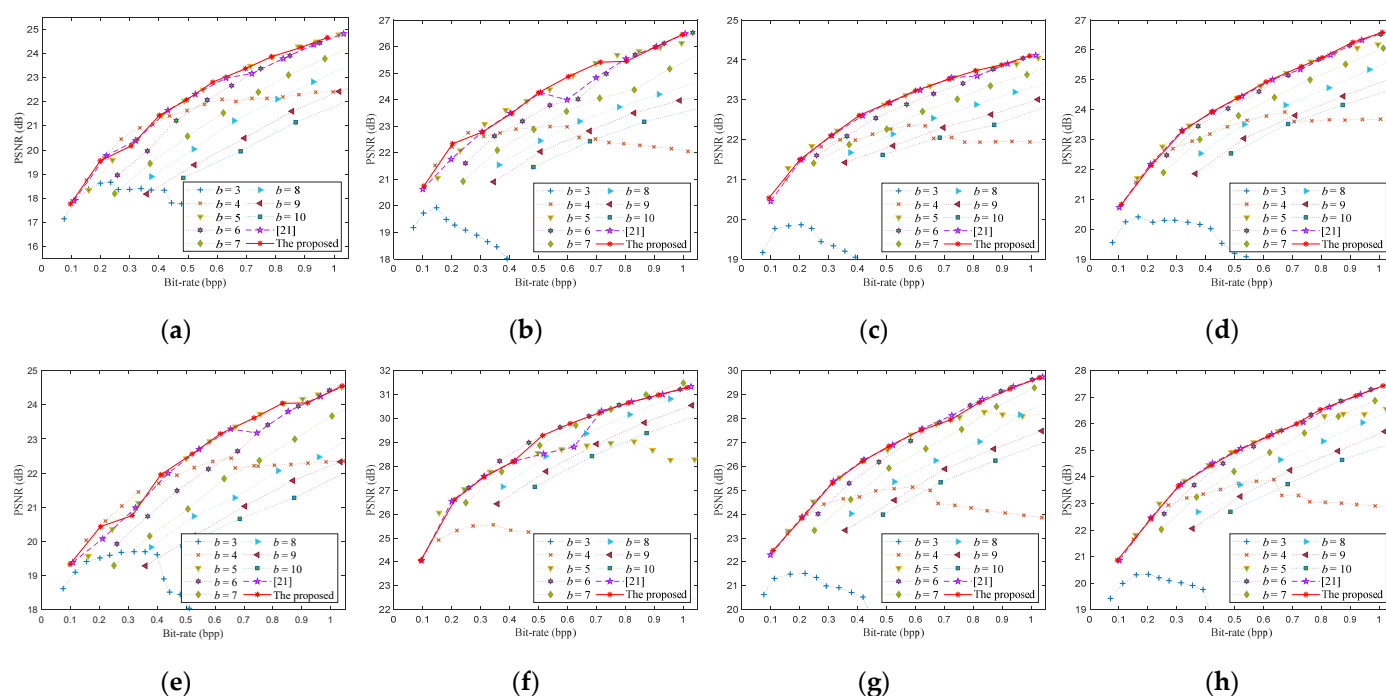


Figure 9. Rate-distortion performance of the proposed algorithm for the SQ framework. (a) Monarch; (b) Parrots; (c) Barbara; (d) Boats; (e) Cameraman; (f) Foreman; (g) House; (h) Lena.

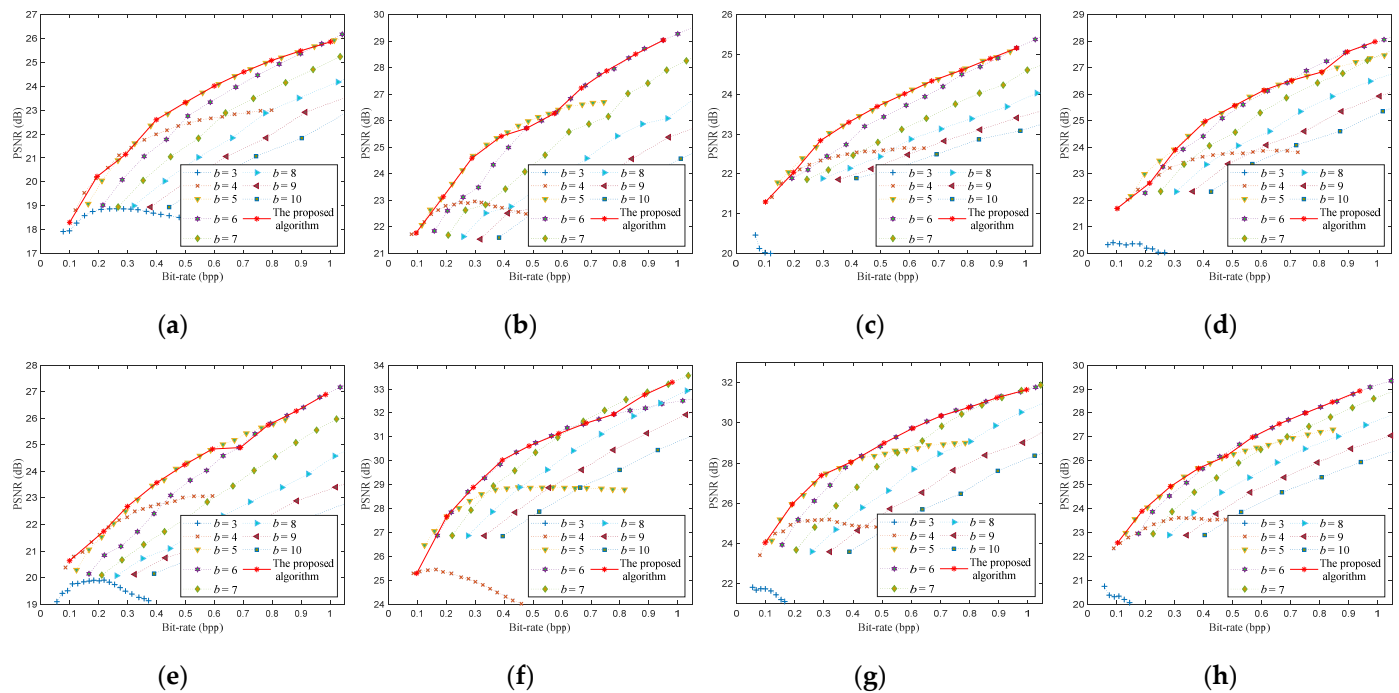


Figure 10. Rate-distortion performance of the proposed algorithm for the DPCM-plus-SQ framework. (a) Monarch; (b) Parrots; (c) Barbara; (d) Boats; (e) Cameraman; (f) Foreman; (g) House; (h) Lena.

Figure 9 shows the proposed algorithm's rate-distortion curves of the eight test images encoded by the CS-based coding system with uniform SQ. The rate-distortion curve of the proposed algorithm is very close to the optimal rate-distortion curve. The PSNRs of the proposed algorithm are slightly worse than the optimal PSNRs only at a few bit-rates. When the bit-rate is 0.3 bpp for Monarch, the predicted optimal bit-depth is 5 bit, while the actual optimal bit-depth is 4 bit. The PSNR of the proposed algorithm is 0.52 dB less than the optimal PSNR at bit-rate of 0.3 bpp. When the bit-rate is 0.3 and 0.8 bpp for Parrots, the predicted optimal bit-depth is 4 and 6, while the actual optimal bit-depth is 5 bit. The PSNRs of the proposed algorithm are about 0.25 and 0.3 dB less than the optimal PSNR at bit-rate of 0.3 and 0.8 bpp. When the bit-rate is 0.3 and 0.9 bpp for Cameraman, the predicted optimal bit-depth is 5 and 6, while the actual optimal bit-depth is 4 and 5. The PSNRs of the proposed algorithm are about 0.48 and 0.12 dB less than the optimal PSNR at bit-rate of 0.3 and 0.9 bpp. When the bit-rate is 0.3 and 0.9 bpp for Foreman, the predicted optimal bit-depth is 5 bit, while the actual optimal bit-depth is 6 bit. The PSNR of the proposed algorithm is 0.41 dB less than the optimal PSNR at bit-rate of 0.4 bpp. The optimal bit-depth model mainly causes these deviations, whereas the maximum deviation is 1 bit.

The proposed algorithm's rate-distortion curves are very close to the results of [21] on Barbara, Boats, House, Lena, and are better than [21] on Monarch, Parrots, Cameraman, Foreman. It can be seen from Figure 9a that the optimal bit-depth is 7 at the bit-rate of 0.7 bpp or 0.8 bpp, the proposed algorithm can accurately predict the optimal bit-depth. However, the bit-depth predicted by [21] is 6, which is one bit less than the optimal bit-depth. In Figure 9b, the optimal bit-depth is 4 at the bit-rate of 0.2 bpp, and the optimal bit-depth is 5 at the bit-rate of 0.6 bpp and 0.7 bpp. Compared with [21], the predicted bit-depths of the proposed algorithm are more accurate. Some similar situations occur in Figure 9e,f.

Figure 10 shows the proposed algorithm's rate-distortion curves of the eight test images encoded by the CS-based coding system with DPCM-plus-SQ. The rate-distortion curve of the proposed algorithm is very close to the optimal rate-distortion curve. The PSNRs of the proposed algorithm are slightly worse than the optimal PSNRs only at a few bit-rates. When the bit-rate is 0.5 bpp for Parrots, the predicted optimal bit-depth is 6 bit,

while the actual optimal bit-depth is 5 bit. The PSNR of the proposed algorithm is 0.25 dB less than the optimal PSNR at bit-rate of 0.5 bpp. When the bit-rate is 0.2 and 0.8 bpp for the image boats, the predicted optimal bit-depth is 4 and 5 bit, while the actual optimal bit-depth is 5 and 6. The PSNRs of the proposed algorithm are about 0.5 dB and 0.4 dB less than the optimal PSNR at bit-rates of 0.2 and 0.8 bpp. When the bit-rate is 0.7 bpp for Cameraman, the predicted optimal bit-depth is 6 bit, while the actual optimal bit-depth is 5 bit. The PSNR of the proposed algorithm is 0.45 dB less than the optimal PSNR at bit-rate of 0.7 bpp. When the bit-rate is 0.8 bpp for Foreman, the predicted optimal bit-depth is 6 bit, while the actual optimal bit-depth is 7 bit. The PSNR of the proposed algorithm is about 0.29 dB less than the optimal PSNR at bit-rate of 0.8 bpp.

From Figures 9 and 10, the prediction deviation of the optimal bit-depth is at most 1 bit, which mainly occurs at the junction between the two optimal bit-depths and has little effect on PSNR. The rate-distortion curves of the proposed algorithm are almost the optimal curve for the DPCM-plus-SQ framework and SQ framework. Although the proposed algorithm's rate-distortion performance is not optimal at some bit-rates, the gap is small.

7. Conclusions

The CS-based coding system needs to assign sampling rate and quantization bit-depth for a given bit-rate before encoding an image. In this work, we first propose a bit-rate model and an optimal bit-depth model for the CS-based coding system. The proposed bit-rate model and optimal bit-depth model have simple mathematical forms, and they have effective parameters based on training off-line data. Then, we propose a general rate-distortion optimization method to assign sampling rate and quantization bit-depth based on the bit-rate model and optimal bit-depth model. The proposed method only needs to extract some features of a small number of measurements, so the computational cost is low. Compared with the first sampling calculation of the CS measurements (blocks' size is 16×16), the addition and multiplication of the optimization process are about 5.94% and 1.17% of the sampling process, respectively, and the percentage decrease as the block size increases. The disadvantage of the proposed method is that a large amount of offline data needs to be collected to train the model parameters, which is usually acceptable. We test the uniform SQ framework and DPCM-plus-SQ framework, respectively. Experimental results show that the optimized rate-distortion performance and bit-rate of the proposed algorithm are very close to the optimal rate-distortion performance and the target bit-rate.

Author Contributions: Conceptualization, Q.C., D.C., and J.G.; data curation, Q.C.; formal analysis, Q.C., D.C., and J.G.; methodology, Q.C., D.C., and J.G.; writing—original draft, Q.C. writing—review and editing, J.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Candès, E.J.; Romberg, J.; Tao, T. Robust Uncertainty Principles : Exact Signal Frequency Information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509.
2. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306.
3. Baraniuk, R.G. Compressive sensing. *IEEE Signal Process. Mag.* **2007**, *24*, 118–121.
4. 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 International Conference on Image Processing, Atlanta, GA, USA, 8–11 October 2006.
5. Romberg, J. Imaging via compressive sampling. *IEEE Signal Process. Mag.* **2008**, *25*, 14–20.

6. Goyal, V.K.; Fletcher, A.K.; Rangan, S. Compressive sampling and lossy compression: Do random measurements provide an efficient method of representing sparse signals? *IEEE Signal Process. Mag.* **2008**, *25*, 48–56.
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.
8. Gan, L. Block compressed sensing of natural images. In Proceedings of the 15th International Conference on Digital Signal Processing, Wales, UK, 1–4 July 2007; pp. 403–406.
9. Unde, A.S.; Deepthi, P.P. Block compressive sensing: Individual and joint reconstruction of correlated images. *J. Vis. Commun. Image Represent.* **2017**, *44*, 187–197.
10. 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.
11. Fletcher, A.K.; Rangan, S.; Goyal, V.K. On the rate-distortion performance of compressed sensing. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Honolulu, HI, USA, 15–20 April 2007; Volume 3, pp. 885–888.
12. Mun, S.; Fowler, J.E. DPCM for quantized block-based compressed sensing of images. In Proceedings of the 20th European Signal Processing Conference; Bucharest, Romania, 27–31 August 2012; pp. 1424–1428.
13. Wang, L.; Wu, X.; Shi, G. Binned progressive quantization for compressive sensing. *IEEE Trans. Image Process.* **2012**, *21*, 2980–2990.
14. Chen, Z.; Hou, X.; Qian, X.; Gong, C. Efficient and Robust Image Coding and Transmission Based on Scrambled Block Compressive Sensing. *IEEE Trans. Multimed.* **2018**, *20*, 1610–1621.
15. 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.* **2020**, *30*, 1109–1120.
16. Li, X.; Lan, X.; Yang, M.; Xue, J.; Zheng, N. Universal and low-complexity quantizer design for compressive sensing image coding. In Proceedings of the 2013 Visual Communications and Image Processing, Kuching, Malaysia, 17–20 November 2013; pp. 1–5.
17. Zhang, J.; Zhao, D.; Jiang, F. Spatially directional predictive coding for block-based compressive sensing of natural images. In Proceedings of the 2013 IEEE International Conference on Image Processing, Melbourne, Australia, 15–18 September 2013; pp. 1021–1025.
18. Shirazinia, A.; Chatterjee, S.; Skoglund, M. Analysis-by-synthesis quantization for compressed sensing measurements. *IEEE Trans. Signal Process.* **2013**, *61*, 5789–5800.
19. Gu, X.; Tu, S.; Shi, H.J.M.; Case, M.; Needell, D.; Plan, Y. Optimizing Quantization for Lasso Recovery. *IEEE Signal Process. Lett.* **2018**, *25*, 45–49.
20. Laska, J.N.; Baraniuk, R.G. Regime change: Bit-depth versus measurement-rate in compressive sensing. *IEEE Trans. Signal Process.* **2012**, *60*, 3496–3505.
21. Chen, Q.; Chen, D.; Gong, J.; Ruan, J. Low-complexity rate-distortion optimization of sampling rate and bit-depth for compressed sensing of images. *Entropy* **2020**, *22*, 125.
22. Jiang, W.; Yang, J. The Rate-Distortion Optimized Compressive Sensing for Image Coding. *J. Signal Process. Syst.* **2017**, *86*, 85–97.
23. Liu, H.; Song, B.; Tian, F.; Qin, H. Joint sampling rate and bit-depth optimization in compressive video sampling. *IEEE Trans. Multimed.* **2014**, *16*, 1549–1562.
24. Huang, J.; Mumford, D. Statistics of natural images and models. In Proceedings of the 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Fort Collins, CO, USA, 23–25 June 1999; Volume 1.
25. Lam, E.Y.; Goodman, J.W. A mathematical analysis of the DCT coefficient distributions for images. *IEEE Trans. Image Process.* **2000**, *9*, 1661–1666.
26. Pudi, V.; Chattopadhyay, A.; Lam, K.Y. Efficient and Lightweight Quantized Compressive Sensing using μ -Law. In Proceedings of the IEEE International Symposium on Circuits and Systems, Florence, Italy, 27–30 May 2018.
27. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612.
28. Jordi Ribas-Corbera, S.L. Rate Control in DCT Video Coding for Low-Delay Communications. *IEEE Trans. Circuits Syst. Video Technol.* **1997**, *9*, 172–185.
29. Wu, X. Lossless compression of continuous-tone images via context selection, quantization, and modeling. *IEEE Trans. Image Process.* **1997**, *6*, 656–664.
30. Qian, C.; Zheng, B.; Lin, B. Nonuniform quantization for block-based compressed sensing of images in differential pulse-code modulation framework. In Proceedings of the 2014 2nd International Conference on Systems and Informatics, Shanghai, China 15–17 November 2015; pp. 791–795.
31. Aiazzi, B.; Alparone, L.; Baronti, S. Estimation based on entropy matching for generalized Gaussian PDF modeling. *IEEE Signal Process. Lett.* **1999**, *6*, 138–140.
32. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
33. Wang, D.; Tan, D.; Liu, L. Particle swarm optimization algorithm: An overview. *Soft Comput.* **2018**, *22*, 387–408.

34. Svozil, D.; Kvasnicka, V.; Pospichal, J. Introduction to multi-layer feed-forward neural networks. *Chemom. Intell. Lab. Syst.* **1997**, *39*, 43–62.
35. Tamura, S.; Tateishi, M. Capabilities of a four-layered feedforward neural network: Four layers versus three. *IEEE Trans. Neural Networks* **1997**, *8*, 251–255.
36. Sadeghi, B.H.M. BP-neural network predictor model for plastic injection molding process. *J. Mater. Process. Technol.* **2000**, *103*, 411–416.
37. Ayoobkhan, M.U.A.; Chikkannan, E.; Ramakrishnan, K. Lossy image compression based on prediction error and vector quantisation. *Eurasip J. Image Video Process.* **2017**, *2017*, 35.
38. Arbeláez, P.; Maire, M.; Fowlkes, C.; Malik, J. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 898–916.
39. Roth, S.; Black, M.J. Fields of experts. *Int. J. Comput. Vis.* **2009**, *82*, 205–229.
40. Skretting, K. Huffman Coding and Arithmetic Coding. Available online: <https://www.mathworks.com/matlabcentral/fileexchange/2818-huffman-coding-and-arithmetic-coding> (accessed on 13 October 2020).
41. Mun, S.; Fowler, J.E. Block compressed sensing of images using directional transforms. In Proceedings of the 16th IEEE International Conference on Image Processing, Cairo, Egypt, 7–10 November 2009; pp. 3021–3024.
42. Lee Rodgers, J.; Alan Nice Wander, W. Thirteen ways to look at the correlation coefficient. *Am. Stat.* **1988**, *42*, 59–66.