

Research Article

JPEG Lifting Algorithm Based on Adaptive Block Compressed Sensing

Yongjun Zhu ^{1,2}, Wenbo Liu ¹, Qian Shen ¹, Yin Wu ³ and Han Bao ¹

¹College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

²School of Electronic and Information Engineering, Suzhou University of Science and Technology, Suzhou 215009, China

³College of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China

Correspondence should be addressed to Yongjun Zhu; zyj@mail.usts.edu.cn

Received 25 May 2020; Accepted 12 June 2020; Published 11 July 2020

Guest Editor: Liangtian Wan

Copyright © 2020 Yongjun Zhu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a JPEG lifting algorithm based on adaptive block compressed sensing (ABCS), which solves the fusion between the ABCS algorithm for 1-dimension vector data processing and the JPEG compression algorithm for 2-dimension image data processing and improves the compression rate of the same quality image in comparison with the existing JPEG-like image compression algorithms. Specifically, mean information entropy and multifeature saliency indexes are used to provide a basis for adaptive blocking and observing, respectively, joint model and curve fitting are adopted for bit rate control, and a noise analysis model is introduced to improve the antinoise capability of the current JPEG decoding algorithm. Experimental results show that the proposed method has good performance of fidelity and antinoise, especially at a medium compression ratio.

1. Introduction

Image processing technology has always been a research hotspot in the field of computer science. Especially, in the recent years, under the emergence of high-definition and large-scale images and the impact of massive video information, image compression processing technology has become particularly noticeable. Image compression technology can use limited storage space to save a larger proportion of image data; at the same time, it can also reduce the data size of images of the same quality, which can effectively improve the efficiency of network data transmission. The traditional image compression technology includes two independent parts, image acquisition and image compression, which limit the fusion improvement method of the two correlated compression technology parts. The emergence of compressed sensing (CS) theory breaks the above frame of image compression, and it completes the image acquisition and compression in the step of sparse observation synchronously; on the one hand, it simplifies the image processing process, and on the other hand, it also provides new research areas for image fusion compression.

There are many types of images processed in image compression technology, and this article selects a still image as the research object. The common still image compression formats include JPEG, JPEG2000, JPEG-XR, TIFF, GIF, and PCX. This paper focuses on the research of image compression algorithms with a JPEG similar structure and improves them with the combination of CS technology. In addition, the algorithms with a similar principle architecture to JPEG are collectively referred to as JPEG-like algorithms, including traditional JPEG, JPEG-LS, JPEG2000, and JPEG-XR. Data redundancy is essential to the compression of a still image. JPEG-like algorithms use time-frequency transform algorithms and entropy coding as main methods to eliminate data redundancy [1–3]. Although having achieved certain effects of still image compression, these algorithms have insufficient considerations on three types of data redundancy (coding redundancy, interpixel redundancy, and psychological visual redundancy) [4]. Firstly, the simple image blocking without guidance could not support the effective coding efficiency to eliminate redundancy in the existing JPEG-like algorithms. Secondly, the uniform time-frequency transform of the same dimension cannot

reasonably use the a priori information between pixels of different subimage blocks to reduce interpixel redundancy. In the end, the former JPEG-like algorithms fail to eliminate psychological visual redundancy by considering overall and local saliency. CS technology breaks through the limitations of the Nyquist sampling theorem to provide innovative ideas for sparse reconstruction of signals [5]. In particular, the adaptive block compressed sensing (ABCS) combined with adaptive partitioning and sampling provides a feasible solution for the optimization of JPEG-like algorithms [6, 7]. That is, the block compression measurement matrix could be used as the forward discrete cosine transform (FDCT) matrix in the JPEG coding, and the inverse discrete cosine transform (IDCT) process is replaced by sparse reconstruction. In addition, multiple feature saliency and noise analysis are introduced to implement adaptive control of the observation matrix and minimal error iterative reconstruction [8, 9].

In this article, we proposed a JPEG lifting algorithm based on the ABCS, and named it as JPEG-ABCS. This proposed algorithm focuses on the following aspects: (1) guiding best morphological blocking by minimizing mean information entropy (MIE); (2) generating an element vector of subimage pixels using the texture feature and 2-dimensional direction DCT; (3) selecting the dimension of the measurement matrix by variance and local significance factors; (4) rate control by matching the overall sampling rate and the quantization matrix; (5) realizing iterative reconstruction of a minimum error under noise condition by using noise influence model analysis.

The remainder of this paper is organized as follows. In Section 2, the basic theories of JPEG-like algorithms and the ABCS algorithm are illustrated. In Section 3, we focus on the introduction of the JPEG-ABCS algorithm. Then, the implementation of the proposed JPEG-ABCS algorithm is analyzed in Section 4. In Section 5, the experiment and result analysis shows the benefit of the JPEG-ABCS. The paper concludes in Section 6.

2. Preliminary Knowledge

2.1. Background of the Existing JPEG-Like Algorithms. The existing JPEG-like algorithms are similar in structure, mainly including blocking, forward time-frequency transform, quantization, entropy coding, and the inverse operation of the above four processes. As the basic one of JPEG-like algorithms, the structure of the JPEG model is shown in Figure 1.

It can be seen from Figure 1 that in the entire JPEG model, the original image I is treated as two-dimensional data, and its key link is adopting the 2-dimensional DCT. Generally, the block size is square, such as 8×8 , and the recommended quantization matrix (light-table) is given in equation (1) [10]. Based on the Hoffman coding, the encoding part adopts differential pulse code modulation (DPCM) for DC coefficients and run length coding (RLC) for AC coefficients:

$$\text{light-table} = \mathbf{Q}_0 = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 84 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}. \quad (1)$$

Compared with the fixed bit rate of the JPEG algorithm, the JPEG-LS algorithm adds the function of rate control by using a quality factor. JPEG2000 adopts nonfixed square blocking (tile) and discrete wavelet transform (DWT) to improve the quality of the restored image. JPEG-XR introduces the lapped orthogonal transform (LOT) to reduce the blocking artifact at low bit rates.

2.2. Basic Theory of CS Algorithm. CS theory was originally proposed by Candès et al. in 2006, which proved that the original signal can be accurately reconstructed by partial Fourier transform coefficients. The advent of CS technology solves the problem that image sampling and compression cannot be performed simultaneously. In general, the main contents of the research on CS theory include sparse representation, compression observation, and optimization reconstruction [11]. Firstly, the main task of sparse representation is to find a set of bases that can make the signal sparse representation, which is the premise and foundation of the entire CS theory. Secondly, the primary task of compression observation is to design a linear measurement matrix uncorrelated with the basis vector to obtain dimensionality reduction observation data, which is the key content of CS theory. Lastly, optimization reconstruction is a difficult problem in CS theory, and its main goal is to solve the original signal through the reverse optimization problem of the sparse vector. The specific solution method of this process is the constrained optimization method.

CS mathematical model is based on the assumption of signal sparsity. Let $\mathbf{x} \in R^N$ be the original signal with n dimension. Suppose that the sparse matrix $\Psi \in R^{N \times N}$ makes the sparse representation coefficient of x as $\mathbf{s} = \Psi^{-1}\mathbf{x}$, where $\mathbf{s} \in R^N$ contains only K ($K \ll N$) nonzero elements. The original signal x is called the K sparse signal under sparse basis Ψ . The number of nonzero elements in the coefficient vector \mathbf{s} can be calculated by $K = \|\mathbf{s}\|_0$, where $\|\cdot\|_0$ denotes l_0 norm.

CS theory states that the information content in sparse signals can be effectively captured by a smaller number of observations. Let $\Phi \in R^{M \times N}$ be the measurement matrix, where $M < N$. The linear dimension-reduction acquisition vector of the original signal \mathbf{x} is given as $\mathbf{y} = \Phi\mathbf{x}$, where $\mathbf{y} \in R^M$ represents the CS observation signal. In addition, the CS theory points out that to accurately recover the original

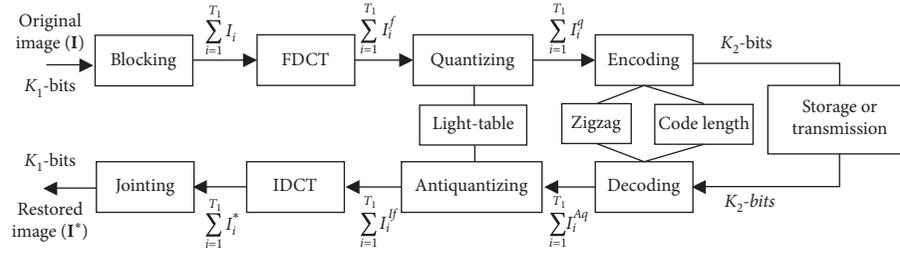


FIGURE 1: Structure of the JPEG model.

signal by the observation signal, its dimensions must obey the following condition: $M \geq cK \log(N)$, where c is an adjustment constant.

Since $M < N$, the reconstruction of the sparse signal \mathbf{x} from the measurement vector \mathbf{y} is ill-posed which requires us to solve the underdetermined system of equations. There are many solutions for such a system. It is common practice to achieve effective signal reconstruction by using signal sparsity as an additional constraint. The accurate signal reconstruction is accomplished through solving the following optimization problem:

$$\min_{\mathbf{s}} \|\mathbf{s}\|_p \quad \text{s.t. } \mathbf{y} = \Phi \mathbf{x} = \Phi \Psi \mathbf{s} = \Omega \mathbf{s}, \quad (2)$$

where Ω is the sensing matrix, $\|\cdot\|_p$ denotes the L_p norm, and the value of p is usually 0, 1, and 2 according to different optimization goals. This is a NP-hard problem, and in order to ensure the stability and robustness of the reconstruction process, the measurement matrix Φ must satisfy the restricted isometric property (RIP).

The above is the description of the three important problems of the CS algorithm, which solves the separation problem of traditional image acquisition and compression. However, when CS is applied to large-scale high-definition images and video processing, because the 2-dimensional image contains a lot of information, the overall projection requires a large-scale measurement matrix, which will inevitably lead to two major problems: excessive storage and reconstruction algorithm complexity. The above problems limit the application of CS in image processing. The emergence of block compressed sensing (BCS) theory solves this problem well. The solution is to cut the whole image into several small unit blocks, reconstruct after independent observation, and then perform stitching to restore and reconstruct the original image.

Traditional block compressed sensing (BCS) technology introduces the idea of blocks in CS theory to solve the dimensional disaster of data processing, and then improves the processing speed of the algorithm [12]. Its basic model is shown in the following equation:

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_{T_1} \end{bmatrix} = \Phi \mathbf{x} = \begin{bmatrix} \Phi_B & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \Phi_B \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{T_1} \end{bmatrix}, \quad (3)$$

where $\mathbf{x}_i \in R^n$, $\mathbf{y}_i \in R^m$, and $\Phi_B \in R^{m \times n}$ are the i -th subblocks of the original signal, observation signal, and block measurement matrix and $T_1 = (N/n) = (M/m)$ is the number of blocks. In addition, a coefficient $\eta = (M/N)$ is often defined in the BCS, which is called the mean sampling rate. In the analysis of the above BCS algorithm model, although the blocking strategy solves the problems of dimensional disaster and computational complexity, the model uses a unified measurement matrix which can neither reflect the inherent differences between each subimage, nor can it achieve differentiated blocking.

In order to overcome the above shortcomings, the nonuniform blocking and observing are introduced into BCS, and combined with the idea of the adaptive algorithm, the ABCS algorithm is generated. The ABCS algorithm mentioned in this article is the introduction of the adaptive strategy into BCS, which is mainly reflected in adaptive blocking and observation [13, 14]. The ABCS algorithm model is as follows:

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_{T_1} \end{bmatrix} = \Phi \mathbf{x} = \begin{bmatrix} \Phi_1 & \cdots & 0 \\ \vdots & \Phi_i & \vdots \\ 0 & \cdots & \Phi_{T_1} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{T_1} \end{bmatrix}, \quad (4)$$

where $\mathbf{x}_i \in R^{n_i}$, $\mathbf{y}_i \in R^{m_i}$, and $\Phi_i \in R^{m_i \times n_i}$ are the i -th subblocks of the original signal, observation signal, and measurement matrix. The difference between the ABCS algorithm and the BCS algorithm is that it gives the dimensional freedom of subblock and measurement matrix, which provides conditions for the reasonable use of the correlation of the internal elements of the original signal.

3. Fusion of JPEG Model and ABCS Algorithm

3.1. Workflow of JPEG Lifting Algorithm. According to the above section, the JPEG image compression model mainly includes blocking, FDCT, quantization, coding, and the inverse process of the above four parts. The focus of this section is to do the research about the method on how to embed the advantages of the ABCS algorithm into the JPEG model. The basis for the fusion of the JPEG model and the ABCS algorithm is that the consistent purpose is for image compression. The former mainly compresses the image by reducing the number of bits occupied by each pixel, and the

latter mainly compresses the data by reducing the amount of sampled data, so the ABCS algorithm is suitable to be embedded to the data acquisition stage of the JPEG model; that is, the ABCS algorithm is fused in the blocking and FDCT processes to reduce the amount of input data in the quantization process. In addition, the distinction of the data processing method in JPEG and ABCS is noticed. The image data in the JPEG algorithm are processed in the form of two-dimensional data, which is conducive to saving the two-dimensional structural characteristics of the image, while the input signal in the ABCS algorithm is a simple one-dimensional vector form and does not have two-dimensional characteristics. The proposed algorithm JPEG-ABCS mainly includes the solution of two main problems: (a) the conversion problem between two-dimensional time-frequency transformation of JPEG and one-dimensional measurement model of ABCS; (b) the specific method of applying ABCS to the JPEG compression algorithm.

In typical JPEG image compression, after the pre-processing stage, an $R \times C$ input image \mathbf{I} is divided into 8×8 size subimages \mathbf{I}_i , $i = 1, 2, \dots, T_1$, $T_1 = ((R \times C)/(8 \times 8))$. Each subimage is transmitted to a 2D DCT transform, and the 2D DCT can be completed using two one-dimensional DCTs according to the separability of the DCT. In addition, the blocking method designed in this paper adopts the variable shape blocking method under a unified dimension ($n = r \times c$), so the FDCT process can be described as follows:

$$\mathbf{I}_i^f = \text{dct2}(\mathbf{I}_i) = \mathbf{D}_r \mathbf{I}_i \mathbf{D}_c^T, \quad (5)$$

where \mathbf{I}_i^f is the subimage in the DCT domain, $\mathbf{D}_r \in R^{r \times r}$ and $\mathbf{D}_c \in R^{c \times c}$ are the 1D vertical and horizontal DCT orthogonal matrices, respectively, and r and c are the number of rows and columns of each subimage [15].

The block sparse representation and the flexible uniform-dimension blocking are introduced into the ABCS algorithm. Equation (4) can be rewritten as follows:

$$\{\mathbf{y}_i\}_{i=1}^{T_1} = \{\Phi_i \mathbf{x}_i\}_{i=1}^{T_1} = \{\Phi_i \Psi_i \mathbf{s}_i\}_{i=1}^{T_1} = \{\Omega_i \mathbf{s}_i\}_{i=1}^{T_1}, \quad (6)$$

where $\mathbf{x}_i \in R^n$, $\mathbf{y}_i \in R^{m_i}$, and $\mathbf{s}_i \in R^n$ are the i -th subblocks of the original signal, observation signal, and sparse signal; $\Phi_i \in R^{m_i \times n}$, $\Psi_i \in R^{n \times n}$, and $\Omega_i \in R^{m_i \times n}$ are the i -th subblocks of the measurement matrix, sparse matrix, and sensing matrix; $\eta_i = (m_i/n)$ is the subsampling rate of the i -th subblock.

For retaining the two-dimensional characteristics of the image signal in the application of JPEG-ABCS, it is necessary to analyze the two-dimensional DCT transform in JPEG and the compression observation in ABCS. It is impossible that the 1-dimension vector \mathbf{x}_i generated directly from the subimage \mathbf{I}_i through column/row scanning has two-dimensional structural characteristics. The inverse solution of the reconstructed signal $\hat{\mathbf{x}}_i$ in the ABCS algorithm is generally denoted as $\hat{\mathbf{x}}_i = \Psi_i \hat{\mathbf{s}}_i = \Psi_i \Omega_i^+ \mathbf{y}_i = \Psi_i \Omega_i^+ \Phi_i \mathbf{x}_i = \Psi_i \Omega_i^+ \Omega_i \mathbf{s}_i$; that is, the reconstruction of the original signal is only related to the sparse representation coefficient \mathbf{s}_i . If the sparse representation coefficient \mathbf{s}_i has two-dimensional structure information, it is equivalent to the original signal \mathbf{x}_i

with two-dimensional structure information. Therefore, the equivalent two-dimensional block vector generation can be achieved by taking the sparse matrix of ABCS as the corresponding matrix under the two-dimensional DCT transform:

$$\mathbf{D}_r \mathbf{I}_i \mathbf{D}_c^T = \mathbf{I}_i^f \Leftrightarrow \Psi_i^+ \mathbf{x}_i = \mathbf{s}_i. \quad (7)$$

Analyzing equation (7), the function between the sparse matrix and the DCT orthogonal matrices is established as follows:

$$\Psi_i = \text{kron}(\mathbf{D}_c^T, \mathbf{D}_r^T), \quad (8)$$

where $\text{kron}(\ast)$ represents the Kronecker product function. Original signal vector \mathbf{x}_i is obtained by scanning the pixel value of the subimage \mathbf{I}_i vertically. In addition, if the texture of the image is not in the vertical and horizontal directions, the directional DCT is used instead of the horizontal and vertical DCT orthogonal matrix.

Replacing FDCT and IDCT in the JPEG with adaptive sparse observing and sparse restoring, respectively, replacing blocking in the JPEG with adaptive blocking and vectorization, adding noise to the data storage or data transmission are done, and then the workflow of the proposed JPEG-ABCS algorithm is shown in Figure 2. Comparing the two image compression models shown in Figures 1 and 2, the key points of the JPEG-ABCS model are (1) adaptive blocking, that is, replacing a fixed block with a variable block; (2) adaptive vectorization, that is, providing a matching vector generation method based on image orientation characteristics; (3) adaptive observing, that is, replacing uniform observing with nonuniform observing; (4) adding a controllable variable in the rate control process to improve the JPEG algorithm; (5) designing a denoising method in adaptive restoring to reduce the noise impact on restored data.

3.2. Innovation of JPEG Lifting Algorithm. The innovations of the above JPEG lifting algorithm are as follows:

- (1) Adding the mean sampling rate to overcome the deficiency of the traditional JPEG-like algorithm that can only use the time-frequency transform and the quantization matrix to eliminate redundant information in image compression.
- (2) By analyzing the correlation between sparseness and error, the optimal OMP iterative algorithm is established to enhance the JPEG-like algorithm's noise immunity performance.
- (3) In the adaptive block observation, the MIE-based adaptive block reduces the information entropy of the subimage set to lower the bpp, the ASM-based adaptive vectorization can ensure the maximum avoidance of image information loss, and the adaptive observation based on multifeature saliency ensures a reasonable distribution of the total measurement number.

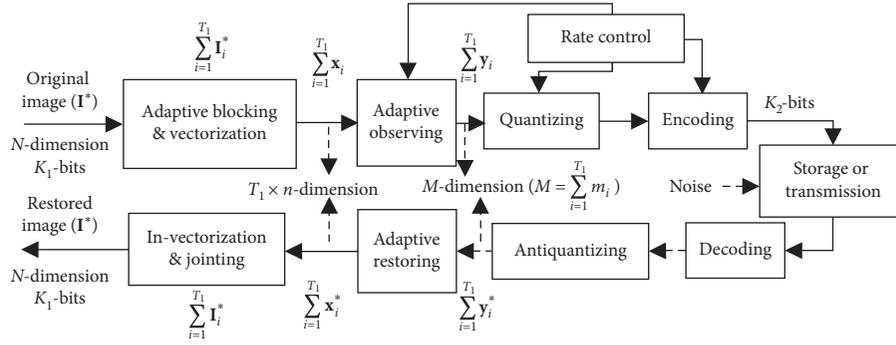


FIGURE 2: Workflow of JPEG-ABCS.

4. Implementation of JPEG-ABCS

This section mainly describes the implementation of the JPEG-ABCS algorithm mentioned in the previous section. The specific implementation is discussed from four aspects: adaptive blocking, adaptive vectorization, adaptive observing, and denoising by optimizing the number of iterations.

4.1. Adaptive Blocking Method Based on MIE. The adaptive blocking method proposed in this paper is a variable partitioning in the same dimension, that is, $n = r \times c$, where n is a fixed value, typical value is 64, and r and c are the number of rows and columns of the variable block, typical value $\{r \times c\}_j = 2^j \times 2^{6-j}$, $j = 0, 1, 2, \dots, 6$. Specifically, the optimized block $n_{\text{opt}} = r_{\text{opt}} \times c_{\text{opt}}$ is based on minimizing the mean information entropy (MIE) of the block observation signal set. Since blocking process needs to be completed before observation, it is impossible to use an ungenerated observation set $\{\mathbf{y}_i\}_{i=1}^{T_1}$ for guiding the blocking optimization. Therefore, an alternative method is introduced to guide the reasonable blocking by minimizing the MIE of the original signal's block set:

$$\begin{cases} r_{\text{opt}} \times c_{\text{opt}} = \arg \min_{\{r \times c\}_j} \left\{ \text{MIE}(\{\mathbf{x}_i\}_{i=1}^{T_1}) \right\}_{j=0}^{T_2-1} \\ \text{MIE}(\{\mathbf{x}_i\}_{i=1}^{T_1}) = \frac{1}{T_1} \sum_{i=1}^{T_1} \text{IE}(\mathbf{x}_i) = \frac{1}{T_1} \sum_{i=1}^{T_1} \left(- \sum_{j=h_{\min}}^{h_{\max}} p_{i,j} \log_2 p_{i,j} \right), \end{cases} \quad (9)$$

where $\text{MIE}(\cdot)$ represents the MIE function of the pixel set, $\text{IE}(\mathbf{x}_i)$ represents the information entropy of the i -th sub-image in the pixel domain, $p_{i,j}$ represents the proportion of elements with the pixel gray value j in the i -th subimage, T_2 is the number of blocking ways, and h_{\min} and h_{\max} are the minimum and maximum values of the pixel gray in the original signal, respectively. However, the effectiveness of the above method lies on consistency between the observation signal's MIE and the original signal's MIE at the same partitioning. In order to verify the above consistency problem, this paper conducted a test experiment using multiple standard images, and its experimental results are shown in Figure 3. The experimental data show that, under the constraint of minimizing MIE, the optimal block of the

original signal and the observed signal is consistent, which verifies the feasibility and rationality of the proposed block optimization method. Specifically, it can be seen from Figure 3 that under the constraint of minimum MIE, the optimal block shape is only related to the test image itself, not to the sampling rate. In addition, it has been verified by a large number of other standard test images that the method of finding the best block has the same trend whenever applied to the observation signal and the original signal, and there must be an extreme point.

4.2. Adaptive Vectorization Based on ASM. The basis of adaptive vectorization is how to identify the directional characteristics of the image. There are many methods for identifying direction features in the field of array signal processing, especially in DOA estimation research, such as the Capon algorithm, MUSIC algorithm, maximum likelihood algorithm, subspace fitting algorithm, and ESPRIT algorithm [16, 17]. In this article, the angular second-order moment (ASM) value under the gray-level cooccurrence matrix (GLCM) is used to characterize the saliency of the direction [18]:

$$\begin{cases} g_{\text{ASM}}(i, j | d, \theta) = \sum_{i=0}^{N_h-1} \sum_{j=0}^{N_h-1} \{p(i, j | d, \theta)\}^2 = \sum_{i=0}^{N_h-1} \sum_{j=0}^{N_h-1} \left\{ \frac{\mathbf{P}(i, j | d, \theta)}{R} \right\}^2, \\ \mathbf{P} = \text{gray comatrix}(\mathbf{I} | d, \theta), \end{cases} \quad (10)$$

where gray comatrix(*) is the GLCM function, $\mathbf{P}(i, j | d, \theta)$ is the (i, j) term of \mathbf{P} , $p(i, j | d, \theta)$ is the normalized form of $\mathbf{P}(i, j | d, \theta)$, $\mathbf{I}(i, j | d, \theta)$ is the adjacent pixel pairs in the image with distance d , direction θ , and gray values (i, j) , and R is the ideal maximum number of pixel pairs under the selected conditions.

Combined with the rectangular shape of adaptive blocking, the ASM values in four directions are defined for adaptive vectorization, namely, g_{ASM}^0 , g_{ASM}^{90} , g_{ASM}^{45} , and g_{ASM}^{135} [19]. In addition, the maximum value of these four values is defined as $g_{\text{MAX}} = \max\{g_{\text{ASM}}^0, g_{\text{ASM}}^{90}, g_{\text{ASM}}^{45}, g_{\text{ASM}}^{135}\}$.

$$\begin{cases} g_{\text{ASM}}^0 = g_{\text{ASM}}(i = j | 1, 0^\circ), & g_{\text{ASM}}^{90} = g_{\text{ASM}}(i = j | 1, 90^\circ), \\ g_{\text{ASM}}^{45} = g_{\text{ASM}}(i = j | 1, 45^\circ), & g_{\text{ASM}}^{135} = g_{\text{ASM}}(i = j | 1, 135^\circ). \end{cases} \quad (11)$$

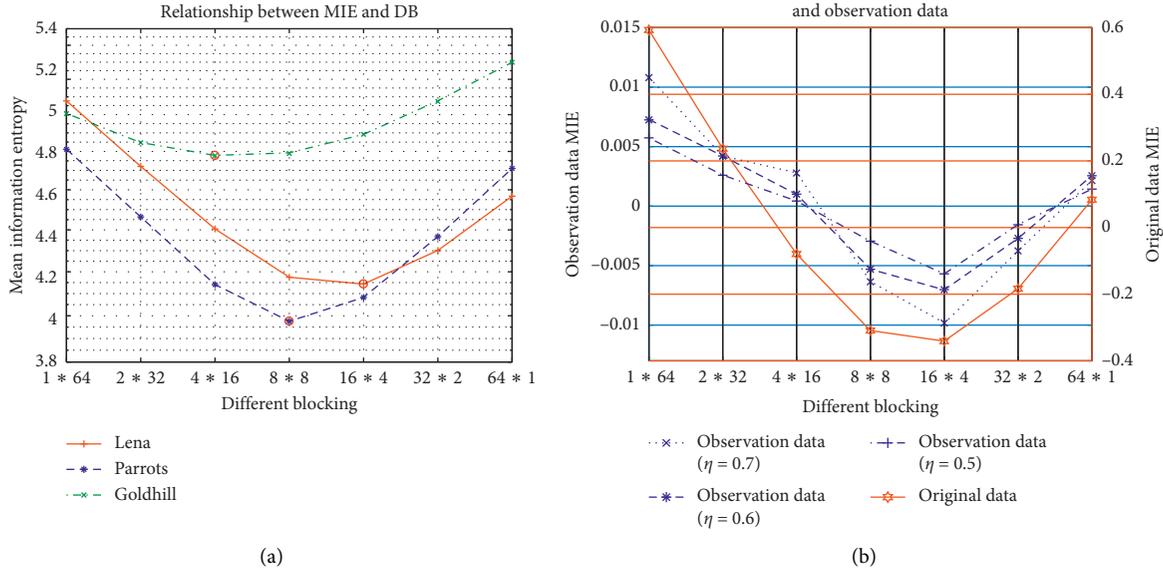


FIGURE 3: MIE changes in terms of different blocking. (a) MIE comparison of different images; (b) MIE consistency between original and observation data.

The specific method of adaptive vectorization is based on the relationship between the four ASM values. If $g_{ASM}^0 \times g_{ASM}^{90} \geq g_{ASM}^{45} \times g_{ASM}^{135}$ and $g_{MAX} = g_{ASM}^0$, then the vectorization set $\sum_{i=1}^{T_1} \mathbf{x}_i$ of the original subimage set $\sum_{i=1}^{T_1} \mathbf{I}_i$ is generated using horizontal scanning and vertical linking; if $g_{ASM}^0 \times g_{ASM}^{90} \geq g_{ASM}^{45} \times g_{ASM}^{135}$ and $g_{MAX} = g_{ASM}^{90}$, then the vectorization set $\sum_{i=1}^{T_1} \mathbf{x}_i$ of the original subimage set $\sum_{i=1}^{T_1} \mathbf{I}_i$ is generated using vertical scanning and horizontal linking; if $g_{ASM}^0 \times g_{ASM}^{90} < g_{ASM}^{45} \times g_{ASM}^{135}$ and $g_{MAX} = g_{ASM}^{45}$, then the vectorization set $\sum_{i=1}^{T_1} \mathbf{x}_i$ of the original subimage set $\sum_{i=1}^{T_1} \mathbf{I}_i$ uses zigzag generation along the main diagonal direction; if $g_{ASM}^0 \times g_{ASM}^{90} < g_{ASM}^{45} \times g_{ASM}^{135}$ and $g_{MAX} = g_{ASM}^{135}$, then the vectorization set $\sum_{i=1}^{T_1} \mathbf{x}_i$ of the original subimage set $\sum_{i=1}^{T_1} \mathbf{I}_i$ uses zigzag generation along the counter-diagonal direction. It should be noted that the adaptive vectorization of each subimage must be related to the design of the sparse matrix to jointly realize the one-dimensional vectorization that preserves the two-dimensional structural characteristics of the subimage data.

4.3. Adaptive Observing Based on Multifeature Saliency and Bit Rate Control. The key point of the nonuniform measurement matrix $\Phi_i \in R^{m_i \times n}$ is the determination of m_i . Considering that different subimages contain different amounts of information and the sensitivity of the human eye's attention mechanism to different images is different, this paper proposes an adaptive measurement matrix $\Phi_i = \sqrt{n/m_i} \Gamma_{m_i}$ based on multifeature saliency $J(\mathbf{x}_i)$ and the orthogonal symmetric Toeplitz matrix (OSTM):

$$\left\{ \begin{array}{l} \Gamma_{m_i} = \text{OSTM}(m_i, n), \\ m_i = \eta_i \times n = \frac{J(\mathbf{x}_i)}{\left(\sum_{i=1}^{T_1} J(\mathbf{x}_i)/T_1\right)} \times \eta \times n, \\ J(\mathbf{x}_i) = \gamma \log_2(J_1(\mathbf{x}_i)^\alpha \times J_2(\mathbf{x}_i)^\beta), \\ J_1(\mathbf{x}_i) = \frac{1}{n} \sum_{j=1}^n (\mathbf{x}_{ij} - \mu_i)^2, J_2(\mathbf{x}_i) = \frac{1}{n} \sum_{j=1}^n \frac{\sum_{k=1}^q |\mathbf{x}_{ij}^k - \mathbf{x}_{ij}|}{\mathbf{x}_{ij}}, \end{array} \right. \quad (12)$$

where γ is the adjustment factor, $J_1(*)$ stands for the overall variance function, $J_2(*)$ is the local saliency function according to Weber's theorem [20], q is the number of elements in the salient domain determined by the optimal bounding box, OSTM(m_i, n) is formed by randomly taking m_i rows of $n \times n$ -dimensional OSTM [21], and $\alpha = 2$ and $\beta = 1$ are the recommended values. The purpose of designing the adaptive measurement matrix in this way is to rationalize the sampling process and to achieve more sampling of detail blocks and less sampling of smooth blocks.

The traditional JPEG-like algorithms control the bit rate (bits per pixel, bpp) through the quantization matrix, encoding, and bit-stream organization [22]. In this paper, the mean sampling rate η has been used to improve the compression performance of the JPEG-ABCS algorithm. The bit rate control for an 8-bit 256-level grayscale image is as follows:

$$\text{bpp} = 8 \times \frac{\eta}{\mu \times \varepsilon}, \quad (13)$$

where η is the mean sampling rate and also corresponds to information decay ratio caused by sparse measurement, μ ($\mu \geq 1$) represents information decay ratio caused by the quantization, and ε ($\varepsilon \geq 1$) is the bit compression ratio of the entropy encoding and bit-stream organization.

Analysis of the above three factors that affect the rate control in the JPEG-ABCS model shows that once the encoding method is determined, the only factors that can be optimized are η and μ , while ε is a fixed number. In order to reduce the bit rate of restored images at the same quality, the value of these two factors must be set reasonably. This article focuses on the analysis of image performance impact in terms of different η under the same bpp and matching design of the quantization matrix that can determine the value of μ . In the process of analyzing the impact of η on the performance of compressed images, the synthetic indicator composed of the peak signal-to-noise rate (PSNR) and structural similarity (SSIM) is used as evaluation criteria to find the best η under different bpp. At the same time, in order to complete the comparison experiment of different η under the same bpp, it is necessary to set different quantization matrices. This article uses the quality factor (QF) to design different quantization matrices [23]. Because the data ($\sum_{i=1}^{T_1} \mathbf{y}_i$) quantized in the JPEG-ABCS algorithm are a one-dimensional vector and are also a normalized measurement of the frequency domain sparse coefficients of the original signal ($\sum_{i=1}^{T_1} \mathbf{x}_i$), the quantization matrix is weakened into a quantization vector whose elements no longer characterize frequency domain property and have the same importance. Therefore, the elements of the quantization matrix for the subimage designed in this paper have the same value ($\mathbf{Q}_0^* = \text{ones}(m_i)$). The goal of the quantization matrix matching design is only to find a fitting function to approximate the relationship between bpp and QF. Figure 4 shows the experimental data of the above test process using Lena. According to Figure 4(a), it can be seen that under the constraint of maximizing synthetic features, the optimal mean sampling rate (η) increases with the increase in bpp. Meanwhile, an η obtaining function can be summarized, as shown in equation (14), and the typical values of B_{th}^1 and B_{th}^2 in the equation are 0.15 and 0.3. In addition, it should be noted that equation (14) can only be directly applied to images with a similar MIE of Lena. For other images, the threshold determination condition in the equation should be corrected according to the MIE of the image block set. Specifically, the coefficient ξ is introduced for the correction of the above two threshold conditions (that is, $B_{\text{th}}^1 = 0.15\xi$ and $B_{\text{th}}^2 = 0.3\xi$). The coefficient ξ can be defined as the MIE ratio of other images to the Lena image. The design of the fitting function ($\text{QF} = f(\text{bpp})$) adopts the cubic curve fitting method whose data are derived from the actual measurement value of QF and bpp. Figure 4(b) shows the comparison of consistency between the actual light-table's QF and the design value obtained from equation (15). From the results, the QF obtained by equation (15) satisfies the actual requirements well:

$$\eta = \begin{cases} 0.3, & \text{bpp} < B_{\text{th}}^1, \\ 0.4 + \left[\frac{\text{bpp} - 0.15}{0.05} \right] \times 0.1, & B_{\text{th}}^1 \leq \text{bpp} \leq B_{\text{th}}^2, \\ 0.7, & \text{bpp} > B_{\text{th}}^2, \end{cases} \quad (14)$$

$$\begin{cases} \text{light-table} = \left\lfloor \frac{\text{QFS} \times \mathbf{Q}_0^* + 50}{100} \right\rfloor, \\ \mathbf{Q}_0^* = \text{ones}(m_i), \\ \text{QFS} = \begin{cases} \frac{5000}{\text{QF}}, & 0 < \text{QF} \leq 50, \\ 200 - 2\text{QF}, & 100 \geq \text{QF} > 50, \end{cases} \\ \text{QF} = f(\text{bpp}) = \lambda_2 \times \text{bpp}^2 + \lambda_1 \times \text{bpp} + \lambda_0, \end{cases} \quad (15)$$

where $\lfloor \cdot \rfloor$ is the floor function and $\lambda_2 = 38.5972$, $\lambda_1 = 6.1241$, and $\lambda_0 = -0.0938$ are obtained by quadratic curve fitting.

4.4. Denoising by Optimizing the Number of Iterations. Consider the noise observation model as follows:

$$\tilde{\mathbf{y}}_i = \Phi_i \mathbf{x}_i + \mathbf{w}_i = \Phi_i (\mathbf{x}_i + \Phi_i^+ \mathbf{w}_i) = \Phi_i \tilde{\mathbf{x}}_i, \quad (16)$$

where \mathbf{w}_i is the additive white Gaussian noise with zero-mean and standard deviation σ_w and $\tilde{\mathbf{x}}_i$ is the equivalent noisy original signal.

Since the reconstructed original signal (\mathbf{x}_i^*) is recovered from the noisy observation signal ($\tilde{\mathbf{y}}_i$), the reconstruction error ($e_{\mathbf{x}_i}$) is mainly caused by the noise and reconstruction algorithm, and its mathematical expression can be defined as the following equation using the l_2 norm:

$$e_{\mathbf{x}_i} = \frac{1}{n} \|\mathbf{x}_i - \mathbf{x}_i^*\|_2^2, \quad (17)$$

where \mathbf{x}_i^* is restored by the pseudoinverse operation, that is, $\mathbf{x}_i^* = \Psi_i \mathbf{s}_i^* = \Psi_i \Omega_i^+ \tilde{\mathbf{y}}_i$, and \mathbf{s}_i^* is the reconstructed sparse signal. The abovementioned Ω_i^+ is the pseudoinverse of Ω_i , usually $\Omega_i^+ = \Omega_i^T (\Omega_i \Omega_i^T)^{-1}$, and also represents the reconstruction algorithm in CS. The reconstruction algorithm of CS is based on the sparse representation of the signal [24], that is, the reconstruction sparsity (v_i) of \mathbf{s}_i^* satisfies the inequality ($v_i \ll m < n$), so the pseudoinverse operation to get \mathbf{x}_i^* can be rewritten as follows:

$$\begin{aligned} \mathbf{x}_i^* &= \Psi_{i,v} \Omega_{i,v}^+ \tilde{\mathbf{y}}_i = \Psi_{i,v} (\Omega_{i,v}^T \Omega_{i,v})^{-1} \Omega_{i,v}^T \tilde{\mathbf{y}}_i \\ &= \Psi_{i,v} (\Omega_{i,v}^T \Omega_{i,v})^{-1} \Omega_{i,v}^T \Phi_i \tilde{\mathbf{x}}_i, \end{aligned} \quad (18)$$

where $\Psi_{i,v}$ is the $n \times v_i$ matrix generated of v_i column vectors in $\Psi_i = [\psi_{i,1} \ \psi_{i,2} \ \dots \ \psi_{i,m}]$ and $\Omega_{i,v}$ is the $m \times v_i$ matrix generated of v_i column vectors in $\Omega_i = [\omega_{i,1} \ \omega_{i,2} \ \dots \ \omega_{i,m}]$ that has the greatest correlation with $\tilde{\mathbf{y}}_i$.

Because equation (17) cannot be calculated directly, we add $\tilde{\mathbf{x}}_i$ to help in analysis and calculation:

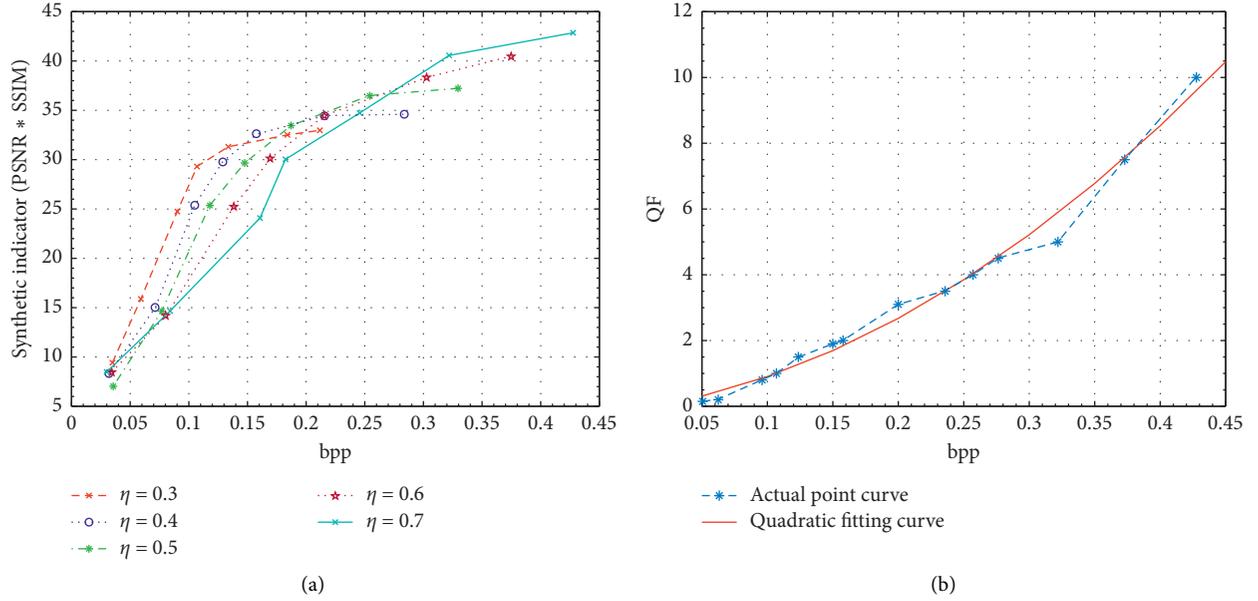


FIGURE 4: Optimization of mean sampling rate (η) and quality factor (QF) in terms of bpp. (a) Impact of mean sampling rate (η) on image quality. (b) Comparison of the actual optimal value of QF with the value calculated by equation (15).

$$\begin{aligned}
 e_{\mathbf{x}_i} &= \frac{1}{n} \left\| (\mathbf{x}_i - \tilde{\mathbf{x}}_i) + (\tilde{\mathbf{x}}_i - \mathbf{x}_i^*) \right\|_2^2 \\
 &= \frac{1}{n} \left\| -\Phi_i^+ \mathbf{w}_i + (\mathbf{I} - \Psi_{i,v} \Omega_{i,v}^+ \Phi_i) \tilde{\mathbf{x}}_i \right\|_2^2 \\
 &= \frac{1}{n} \left\| -\Psi_{i,v} \Omega_{i,v}^+ \mathbf{w}_i + (\mathbf{I} - \Psi_{i,v} \Omega_{i,v}^+ \Phi_i) \mathbf{x}_i \right\|_2^2 \\
 &= \frac{1}{n} \left\| \mathbf{G}_{\mathbf{x}_i} \mathbf{x}_i - \mathbf{G}_{\mathbf{w}_i} \mathbf{w}_i \right\|_2^2,
 \end{aligned} \tag{19}$$

$$\begin{cases} \mathbf{G}_{\mathbf{x}_i} = \mathbf{I} - \Psi_{i,v} \Omega_{i,v}^+ \Phi_i, \\ \mathbf{G}_{\mathbf{w}_i} = \Psi_{i,v} \Omega_{i,v}^+, \end{cases} \tag{20}$$

where $\mathbf{G}_{\mathbf{x}_i}$ is a projection matrix of rank $n - v_i$ and $\mathbf{G}_{\mathbf{w}_i}$ is a projection matrix of rank v_i . Since $\mathbf{G}_{\mathbf{x}_i}$ and $\mathbf{G}_{\mathbf{w}_i}$ satisfy orthogonality, the inner product of $\mathbf{G}_{\mathbf{x}_i} \mathbf{x}_i$ and $\mathbf{G}_{\mathbf{w}_i} \mathbf{w}_i$ is equal to zero. Therefore, equation (19) can be transformed into the following form:

$$e_{\mathbf{x}_i} = \frac{1}{n} \left\| \mathbf{G}_{\mathbf{x}_i} \mathbf{x}_i - \mathbf{G}_{\mathbf{w}_i} \mathbf{w}_i \right\|_2^2 = \frac{1}{n} \left\| \mathbf{G}_{\mathbf{x}_i} \mathbf{x}_i \right\|_2^2 + \frac{1}{n} \left\| \mathbf{G}_{\mathbf{w}_i} \mathbf{w}_i \right\|_2^2 = e_{\mathbf{x}_i}^a + e_{\mathbf{x}_i}^b. \tag{21}$$

Equation (21) reveals that the reconstruction error ($e_{\mathbf{x}_i}$) is composed of the algorithm error ($e_{\mathbf{x}_i}^a$) and the noise error ($e_{\mathbf{x}_i}^b$). $e_{\mathbf{x}_i}^a$ decreases as the reconstruction sparsity (v_i) increases, and $e_{\mathbf{x}_i}^b$ increases with the reconstruction sparsity (v_i) [25, 26]. Therefore, reconstruction error and reconstruction sparsity are a bias-variance trade-off, and there must be an optimal reconstruction sparsity (v_i^{opt}) that minimizes the reconstruction error ($e_{\mathbf{x}_i}$):

$$v_i^{\text{opt}} = \arg \min_{v_i} \{e_{\mathbf{x}_i}\} = \arg \min_{v_i} \left\{ \frac{1}{n} \left\| \mathbf{G}_{\mathbf{x}_i} \mathbf{x}_i \right\|_2^2 + \frac{1}{n} \left\| \mathbf{G}_{\mathbf{w}_i} \mathbf{w}_i \right\|_2^2 \right\}. \tag{22}$$

Figure 5 shows the relationship between reconstruction sparsity and reconstruction error under different noise conditions by using a modified Lena test image. The modified Lena image is generated by intercepting 60 sparse coefficients under a discrete cosine basis; that is, its original sparsity (K) is 60. The noise added in the test is zero-mean Gaussian white noise, and its standard deviation ($\sigma_w = \text{noise} - \text{std}$) also represents the intensity of the noise. The indicator PSNR is used to characterize the size of the reconstruction error. It can be seen from Figure 5 that the optimal reconstruction sparsity decreases as the noise intensity increases and is less than the original sparsity ($v_i^{\text{opt}} \leq K$).

From the verification experiment shown in Figure 5, we can see that there is indeed an optimal reconstruction sparsity in the reconstruction process under the noise background. However, equation (22) is not a feasible solution that can be directly used to optimize the reconstruction process. In the actual reconstruction process, only the observation data at the receiving end can be used for the optimization algorithm. Therefore, this paper designs a solution that uses observation data to optimize the reconstruction sparsity.

According to the definition of CS, the measurement matrix (Φ_i) obeys the RIP criteria, and therefore

$$\begin{cases} (1 - \delta_K) e_{\mathbf{x}_i} \leq e_{\mathbf{y}_i} \leq (1 + \delta_K) e_{\mathbf{x}_i}, \\ e_{\mathbf{y}_i} = \frac{1}{m_i} \left\| \mathbf{y}_i - \mathbf{y}_i^* \right\|_2^2 = \frac{1}{m_i} \left\| \Phi_i (\mathbf{x}_i - \mathbf{x}_i^*) \right\|_2^2, \end{cases} \tag{23}$$

where δ_K is a coefficient related to Φ_i and K , and $e_{\mathbf{y}_i}$ is the reconstruction error of observation data. The transformation of formula (23) can get the boundaries of the original data reconstruction error as follows:

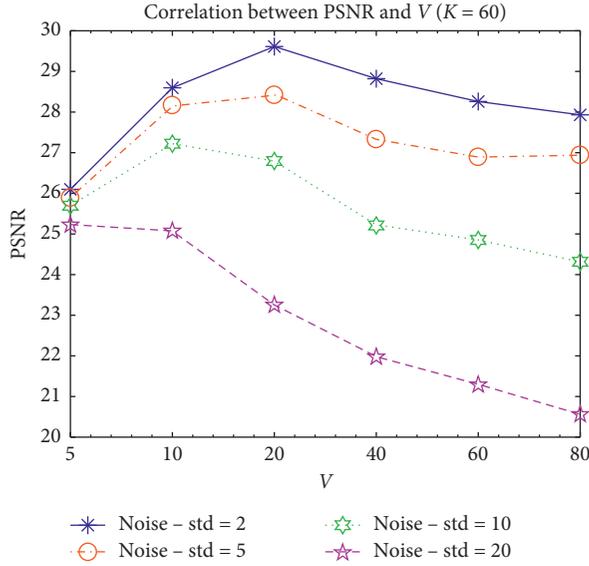


FIGURE 5: Testing the correlation between PSNR and v_i using modified Lena as a test image.

$$\frac{e_{y_i}}{(1 + \delta_K)} \leq e_{x_i} \leq \frac{e_{y_i}}{(1 - \delta_K)}. \quad (24)$$

It can be seen from the above two equations that the reconstruction errors of the original data and the observation data are consistent, so the reconstruction sparsity can be optimized by minimizing the errors of the observation data:

$$\begin{aligned} v_i^{\text{opt}} &= \arg \min_{v_i} \{e_{y_i}\} = \arg \min_{v_i} \left\{ \frac{1}{m_i} \|\mathbf{y}_i - \mathbf{y}_i^*\|_2^2 \right\} \\ &= \arg \min_{v_i} \left\{ \frac{1}{m_i} \|C_{y_i} \mathbf{y}_i\|_2^2 + \frac{1}{m_i} \|C_{w_i} \mathbf{w}_i\|_2^2 \right\}, \\ &\begin{cases} C_{y_i} = \mathbf{I} - \mathbf{\Omega}_{i,v} \mathbf{\Omega}_{i,v}^+, \\ C_{w_i} = \mathbf{\Omega}_{i,v} \mathbf{\Omega}_{i,v}^+, \end{cases} \end{aligned} \quad (25)$$

$$\frac{m_i}{\sigma_w^2} e_{y_i} \sim \chi_{v_i}^2.$$

It is known from the above equation that the reconstruction errors of the observation signal satisfies the chi-square distribution, so the upper and lower boundary of e_{y_i} can be derived from the chi-square distribution probability. In addition, when calculating the minimum value of e_{y_i} , the worst condition is considered, that is, by calculating the minimum value of the upper bound of e_{y_i} .

In the l_0 norm reconstruction algorithm of CS, the reconstruction sparsity is equal to the number of iterations. Therefore, optimizing the number of iterations can reduce the noise impact on image quality in using orthogonal matching pursuit (OMP) as the signal recovery algorithm [27, 28]. According to the Bayesian information criterion (the tuning parameters of confidence probability and effective probability are taken as $\sqrt{v_i} \log m$ and 0, respectively)

[29], the optimal value of iteration number v_i^{opt} can be achieved by minimizing the noise influence:

$$v_i^{\text{opt}} = \arg \min_{v_i} \left\{ \left(\frac{(2 + \sqrt{2} \log_2 m_i) \times v_i}{m_i} - 1 \right) \sigma_w^2 + e_{y_i} \right\}, \quad (26)$$

where $e_{y_i} = (1/m_i) \|\tilde{\mathbf{y}}_i - \mathbf{y}_i^*\|_2^2 = (1/m_i) \|\tilde{\mathbf{y}}_i - \Phi_i \mathbf{x}_i^*\|_2^2$ is the noise error of observation data.

4.5. Pseudocode of JPEG-ABCS. The JPEG lifting algorithm (JPEG-ABCS) described in this article mainly consists of the above four sections, except for the entropy codec, and its full pseudocode is shown in Algorithm 1.

5. Experiment and Result Analysis

In order to verify the superiority of the JPEG-ABCS algorithm, experiments were conducted in two cases: noiseless and noisy. Standard JPEG and JPEG2000 algorithms were used as comparison algorithms, and multiple grayscale standard images with 256×256 resolution were used in the following experiments which were conducted in the simulation software environment of Matlab2016b. In order to objectively evaluate the performance of the algorithm, peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) were introduced as image reconstruction evaluation indexes.

The PSNR index is the most widely used objective standard for characterizing the quality of reconstructed images:

$$\text{PSNR} = 20 \times \log_{10} \left(\frac{255}{\sqrt{(1/N) \sum_{i=1}^N [\mathbf{x}(i) - \mathbf{x}^*(i)]^2}} \right), \quad (27)$$

where $\mathbf{x}(i)$ and $\mathbf{x}^*(i)$ are the i -th element of the original image signal and the reconstructed original image signal, respectively.

The SSIM is another common signal reconstruction quality evaluation index used to describe the similarity between the original image signal and the reconstructed image signal:

$$\text{SSIM} = \frac{(2\mu_{\mathbf{x}}\mu_{\mathbf{x}^*} + c_1)(2\sigma_{\mathbf{x}\mathbf{x}^*} + c_2)}{(\mu_{\mathbf{x}}^2 + \mu_{\mathbf{x}^*}^2 + c_1)(\sigma_{\mathbf{x}}^2 + \sigma_{\mathbf{x}^*}^2 + c_2)}, \quad (28)$$

where $\mu_{\mathbf{x}}$ and $\mu_{\mathbf{x}^*}$ are the average gray value of all elements in \mathbf{x} and \mathbf{x}^* , $\sigma_{\mathbf{x}}$ and $\sigma_{\mathbf{x}^*}$ are the standard deviation of all elements in \mathbf{x} and \mathbf{x}^* , $\sigma_{\mathbf{x}\mathbf{x}^*}$ is the covariance of \mathbf{x} and \mathbf{x}^* , $c_1 = 0.01 \times H^2$ and $c_2 = 0.03 \times H^2$ are the constants, and $H = h_{\text{max}} - h_{\text{min}}$ is the range of pixel gray values.

5.1. Experiments and Analysis without Noise. The experiments under noiseless condition is mainly divided into three parts.

5.1.1. Effectiveness Experiment about Reducing bpp of JPEG Image by MIE Minimization Adaptive Blocking. In order to verify the effectiveness of the proposed MIE

minimization adaptive blocking method, this paper has conducted experiments on two standard images (Lena and Parrots), whose MIE under different blocks is known in the above section, and the experimental data are recorded in Table 1. The basic JPEG algorithm is used in the experiment, and the quantization matrix (light-table) is generated using formula (15), where QF is 50 and 25, respectively.

From the data in Table 1, it can be seen that the Lena and Parrots recovery images have the minimum bpp in the block shape of 16×4 and 8×8 , which just coincides with the minimum MIE block in the above section.

5.1.2. Verification Experiment of ASM-Based Adaptive Vectorization to Improve the 2D Image Reconstruction Performance in BCS Algorithm. The application of CS to image signal processing requires vectorization of two-dimensional images. Therefore, the proposed JPEG-ABCS algorithm also needs to vectorize the subimages. Different vectorization methods directly affect the reconstruction quality of the image.

In this article, three common vectorization methods (vertical scanning, 2D scanning, and zigzag scanning) are compared and analyzed. The verification experiment is carried out in the CS algorithm, using two kinds of images (standard test image and texture test image) to test under different blocking and sampling rates. The experimental data are recorded in Tables 2 and 3, respectively.

It can be seen from Table 2 that for standard test images that meet ASM conditions 1 or 2, vectorization using 2D scanning is optimal, and its PSNR and SSIM of the reconstructed image have a relative advantage than the other two methods.

Table 3 shows the results of texture test image reconstruction using three vectorization methods. Obviously, the vector generation using zigzag scanning has better performance at this case.

It can be seen from the above tables that for different types of test images, the use of a single mode of the vector generation method cannot always effectively improve the quality of image reconstruction, and the multimode method combined with image texture direction feature detection is recommended. Therefore, this paper proposes an adaptive vectorization method based on ASM to maximize the performance of 2D image reconstruction in the BCS algorithm.

5.1.3. Performance Comparison of Various JPEG-Like Algorithms. The image reconstruction quality of the three algorithms was compared with each other under noiseless condition to verify the benefit and universality of the proposed JPEG-ABCS. The experiment includes two parts: verification under different bpp and different test images. Figure 6 shows the experimental results of the Lena test image under three JPEG-like algorithms. Figures 6(a) and 6(b) show that compared with the JPEG and JPEG2000 algorithms, the proposed algorithm has advantages in PSNR and SSIM indicators under different

bpp conditions. Furthermore, from the transformation trend of the curve, it can be seen that the JPEG-ABCS algorithm has good performance at medium and high bit rates, but as the bit rate decreases, the performance of the algorithm proposed in this article has declined, the main reason is that as the dimension of the measurement matrix decreases with bpp, the observation process cannot cover all the information of the image. Figure 6(c) is the restored grayscale images of Lena using the three algorithms under 0.25 bpp condition. It can be seen from a subjective vision that JPEG-ABCS's performance is better than the other two algorithms.

In addition, Table 4 records the experimental results of the three algorithms for different images under the condition of $\text{bpp} = 0.25, 0.3, \text{ and } 0.4$. The data in the table show that the improvement effect of the proposed algorithm is universal for different images. For instance, at the condition of $\text{bpp} = 0.3$, the PSNR index of the four standard test images under the JPEG-ABCS algorithm is improved by 8.34%, 15.09%, 4.46%, and 8.13% compared to the JPEG algorithm and 6.19%, 12.98%, 3.39%, and 6.39% compared to the JPEG2000 algorithm, respectively; the SSIM index has also been improved, compared to JPEG it increased by 0.96%, 0.88%, 1.22%, and 0.59% and compared to JPEG2000 it increased by 0.62%, 0.64%, 0.84%, and 0.30%.

As can be seen from Figure 6 and Table 4, compared to the JPEG and JPEG2000 algorithms, the proposed JPEG-ABCS algorithm has a large improvement on PSNR and SSIM, mainly due to the adaptive blocking and adaptive sampling reducing MIE of image blocks under the same conditions, while sparse restoration guarantees image restoration quality.

5.2. Experiments and Analysis under Gaussian Noisy Conditions. In the Gaussian noise condition, it is verified that the JPEG-ABCS algorithm has improved antinoise performance compared to the standard JPEG algorithm. Figure 7 shows the experimental results which are obtained by using the monarch, peppers, and cameraman as test images. It can be clearly seen from Figure 7 that the test images reconstructed using JPEG-ABCS are superior to the test images reconstructed by JPEG in different noise intensities (here, the noise standard deviation is used as the noise intensity), especially the more noise intensity, the more obvious the superiority.

Table 5 shows the PSNR and SSIM comparison records of the noisy monarch test image under two different algorithms. As can be seen from the data in Table 5, the noisy image reconstruction performance of the JPEG-ABCS algorithm under different bpp conditions is better than that of the JPEG algorithm. For example, at the condition of $\text{bpp} = 0.25$, the PSNR index of the JPEG-ABCS algorithm under the four noise intensities is improved by 9.68%, 4.25%, 0.74%, and 3.06% compared to the JPEG algorithm; the SSIM index has also been improved, compared to JPEG by 4.54%, 3.69%, 1.53%, and 7.71%. In other words, JPEG-ABCS adds antinoise capability that the JPEG algorithm does not have.

```

(1) Input:
    Original image  $I$ ; rate control bpp;
    Subimage dimension  $n = 64$ ;
(2) Initialization:
     $(R, C) = \text{size of } (I); N = R \times C$ ;
     $T_1 = (N/n)$ ; //quantity of subimages
     $T_2 = 1 + \log_2 n$ ; //type number of adaptive blocking
    //step1: adaptive blocking and vectorization
(3) for  $j = 1, \dots, T_2$  do
(4)    $r_j \times c_j = 2^{j-1} \times 2^{T_2-j}$ ;
(5)    $\{\mathbf{I}_i \mid \text{size of } (\mathbf{I}_i) = r_j \times c_j, i = 1, \dots, T_1\} \leftarrow I$ 
(6)    $\{\mathbf{x}_i, i = 1, \dots, T_1\}_j \leftarrow \{\mathbf{I}_i, i = 1, \dots, T_1\}_j$ ;
(7)    $g_{\text{MIE}}^j = \text{MIE}(\{\mathbf{x}_i, i = 1, \dots, T_1\}_j)$ ;
(8) end for
(9)  $j_{\text{opt}} = \arg \min_j (\{g_{\text{MIE}}^j, j = 1, \dots, T_2\})$ ;
(10)  $r \times c = 2^{j_{\text{opt}}-1} \times 2^{T_2-j_{\text{opt}}}$ ;
(11)  $\{\mathbf{I}_i \mid \text{size of } (\mathbf{I}_i) = r \times c, i = 1, \dots, T_1\} \leftarrow I$ ;
(12)  $\{g_{\text{ASM}}^0, g_{\text{ASM}}^{45}, g_{\text{ASM}}^{90}, g_{\text{ASM}}^{135}\} \leftarrow I$ 
(13) if  $(g_{\text{ASM}}^0 \times g_{\text{ASM}}^{90} \geq g_{\text{ASM}}^{45} \times g_{\text{ASM}}^{135})$ ;
(14)   if  $(g_{\text{MAX}} = g_{\text{ASM}}^0)$  //Condition = 1
(15)      $\{\mathbf{x}_i, i = 1, \dots, T_1\}^1 \leftarrow \{\mathbf{I}_i, i = 1, \dots, T_1\}$ ;
     //1  $\leftarrow$  : horizontal scanning and vertical linking
(16)   else if  $(g_{\text{MAX}} = g_{\text{ASM}}^{90})$  //Condition = 2
(17)      $\{\mathbf{x}_i, i = 1, \dots, T_1\}^2 \leftarrow \{\mathbf{I}_i, i = 1, \dots, T_1\}$ ;
     //2  $\leftarrow$  : vertical scanning and horizontal linking
(18)   end if
(19) else if  $(g_{\text{ASM}}^0 \times g_{\text{ASM}}^{45} < g_{\text{ASM}}^{90} \times g_{\text{ASM}}^{135})$ 
(20)   if  $(g_{\text{MAX}} = g_{\text{ASM}}^{45})$  //Condition = 3
(21)      $\{\mathbf{x}_i, i = 1, \dots, T_1\}^3 \leftarrow \{\mathbf{I}_i, i = 1, \dots, T_1\}$ ;
     //3  $\leftarrow$  : zigzag along the main diagonal direction
(22)   else if  $(g_{\text{MAX}} = g_{\text{ASM}}^{135})$  //Condition = 4
(23)      $\{\mathbf{x}_i, i = 1, \dots, T_1\}^4 \leftarrow \{\mathbf{I}_i, i = 1, \dots, T_1\}$ ;
     //4  $\leftarrow$  : zigzag along the counter-diagonal direction
(24)   end if
(25) end if
    //step2: adaptive observing and bit rate control
(26) for  $i = 1, \dots, T_1$  do
(27)    $J_1(\mathbf{x}_i) \leftarrow \mathbf{x}_i; J_2(\mathbf{x}_i) \leftarrow \mathbf{I}_i$ ;
(28)    $J(\mathbf{x}_i) = \gamma \log_2 (J_1(\mathbf{x}_i)^\alpha \cdot J_2(\mathbf{x}_i)^\beta)$ ;
     //synthetic feature ( $J$ )
(29) end for
(30)  $\{\lambda_i, i = 1, \dots, T_1\} = (\log_2 J(\{\mathbf{x}_i, i = 1, \dots, T_1\}) / (1/T_1) \sum_{i=1}^{T_1} \log_2 J(\mathbf{x}_i))$ ;
(31)  $\eta \leftarrow \text{equation (14)} \text{bpp}$ ;
(32)  $\{\eta_i, i = 1, \dots, T_1\} = \{\lambda_i, i = 1, \dots, T_1\} \times \eta$ ;
     //  $\eta_i$  -- sampling ratio of subimages
(33)  $\{m_i, i = 1, \dots, T_1\} = \{\eta_i, i = 1, \dots, T_1\} \times n$ ;
(34)  $\{\widehat{m}_i, i = 1, \dots, T_1\} \leftarrow \{m_i, i = 1, \dots, T_1\} \& m_{\min} \& m_{\max}$ 
     //prevent undersampling and oversampling
(35)  $\Gamma = \text{OSTM}(n), \chi_i = \text{randperm}(n), \Gamma_{\chi_i} = \Gamma(\chi_i, :)$ ;
(36)  $\{\Phi_i, i = 1, \dots, T_1\} = \left\{ \sqrt{n/\widehat{m}_i} \Phi_{\chi_i}([1, \dots, \widehat{m}_i], :), i = 1, \dots, T_1 \right\}$ ;
(37)  $\{\mathbf{y}_i, i = 1, \dots, T_1\} \leftarrow \text{equation (16)} \{\Phi_i \mathbf{x}_i, i = 1, \dots, T_1\} \& \sigma_w$ ;
     //  $\sigma_w = 0$ : noiseless;  $\sigma_w \neq 0$ : noise
(38) light - table  $\leftarrow \text{equation (15)} \text{QF} \leftarrow \text{equation (15)} \text{bpp}$ ;
(39)  $\{\mathbf{y}_i^q, i = 1, \dots, T_1\} \leftarrow \text{light - table} \& \{\mathbf{y}_i, i = 1, \dots, T_1\}$ 
    //step3: codec and anti-quantization
(40) Bit Stream  $\leftarrow \{\mathbf{y}_i^q, i = 1, \dots, T_1\} \& \text{RLC encoding}$ 
(41)  $\{\widehat{\mathbf{y}}_i^q, i = 1, \dots, T_1\} \leftarrow \text{Bit Stream} \& \text{RLC decoding}$ 
(42)  $\{\widehat{\mathbf{y}}_i, i = 1, \dots, T_1\} \leftarrow \text{light - table} \& \{\widehat{\mathbf{y}}_i^q, i = 1, \dots, T_1\}$ 

```

ALGORITHM 1: Continued.

```

//step4: reconstruction and denoising
(43) if Condition == 1
(44)    $\Psi = \text{kron}(\mathbf{D}_c^T, \mathbf{D}_r^T)$ 
(45) else if Condition == 2
(46)    $\Psi = \text{kron}(\mathbf{D}_r^T, \mathbf{D}_c^T)$ 
(47) else  $\Psi = \mathbf{D}_n^T$ 
(48) end if
(49)  $\{\Omega_i, i = 1, \dots, T_1\} = \{\Phi_i \Psi, i = 1, \dots, T_1\}$ 
(50) for  $i = 1, \dots, T_1$  do
(51)    $\Omega_i = \{\omega_{i1}, \dots, \omega_{in}\}, r = \hat{y}_i, A = \emptyset, \mathbf{s}^* = \mathbf{0}^n;$ 
// $\{\omega_{ij}, j = 1, \dots, n\}$ -- column vector of  $\Omega_i$ 
(52)    $v_i^{\text{opt}} \leftarrow \text{equation(28)}\{v_i | \text{argmin}_{v_i} \overline{e_{y_i}}\};$ 
//calculate optimal iterative of subimages
(53)   for  $j = 1, \dots, v_i^{\text{opt}}$  do
(54)      $\wedge = \text{argmin}_j |\langle r, \omega_{ij} \rangle|;$ 
(55)      $A = AU\{\wedge\};$ 
(56)      $r = \hat{y}_i - \Omega_i(:, A) [\Omega_i(:, A)]^+ \hat{y}_i;$ 
(57)   end for
(58)    $\mathbf{s}_i^* = [\Omega_i(:, A)]^+ \hat{y}_i;$ 
// $\mathbf{s}_i^*$ : reconstruction sparse representation
(59)    $\mathbf{x}_i^* = \Psi \mathbf{s}_i^*;$ 
// $\mathbf{x}_i^*$ : reconstruction original signal of subimages
(60) end for
(61)  $\mathbf{x}^* = \{\mathbf{x}_i^*, i = 1, \dots, T_1\};$ 
//step5: antivectorization and jointing
(62)  $\mathbf{I}_i^* = \{\mathbf{x}_i^*, r, c\}; \mathbf{I}^* = \{\mathbf{I}_i^* | i = 1, \dots, T_1, \text{size of } (\mathbf{I}^*) = R \times C\}$ 
// $\mathbf{I}^*$ : recovered image with JPEG-ABCS

```

ALGORITHM 1: JPEG-ABCS algorithm based on OMP (orthogonal matching pursuit).

TABLE 1: Effecton of different block shapes on bpp of the JPEG image.

Test image	QF	Correlation between bpp and block shape ($r \times c$)						
		1×64	2×32	4×16	8×8	16×4	32×2	64×1
Lena	50	1.3235	1.2179	0.9351	0.7430	0.7290	0.8324	0.9068
	25	0.8776	0.8304	0.6227	0.4838	0.4733	0.5462	0.5892
Parrots	50	1.1168	0.9629	0.7463	0.6161	0.6498	0.7674	0.8308
	25	0.7104	0.6353	0.4824	0.3988	0.4246	0.4972	0.5403

TABLE 2: Comparison experiments of different vectorization methods on image reconstruction performance using standard test images.

Test image	Verification method		Vertical scanning	2D scanning PSNR/SSIM	Zigzag scanning
	ASM condition	$r \times c$			
Lena	Condition = 2	64×4	30.69/0.8907	32.11/0.9402	30.86/0.8984
		32×8	31.39/0.9150	32.74/0.9477	31.31/0.9138
		16×16	31.72/0.9227	32.87/0.9470	31.73/0.9261
		8×32	31.68/0.9248	32.02/0.9325	31.72/0.9257
		4×64	31.07/0.9221	31.64/0.9321	30.78/0.9175
		64×4	29.12/0.8783	29.71/0.9211	29.14/0.8787
Goldhill	Condition = 1	32×8	29.45/0.8977	30.21/0.9251	29.38/0.8923
		16×16	29.83/0.9134	30.35/0.9335	29.95/0.9146
		8×32	30.02/0.9113	30.28/0.9288	30.13/0.9192
		4×64	29.98/0.9102	30.17/0.9191	29.87/0.9112
		64×4	29.73/0.9031	30.90/0.9547	29.70/0.9034
		32×8	30.09/0.9068	31.67/0.9585	30.20/0.9158
Peppers	Condition = 2	16×16	30.77/0.9309	32.02/0.9638	30.76/0.9293
		8×32	30.91/0.9413	31.36/0.9565	30.60/0.9325
		4×64	30.38/0.9421	30.63/0.9506	30.38/0.9418

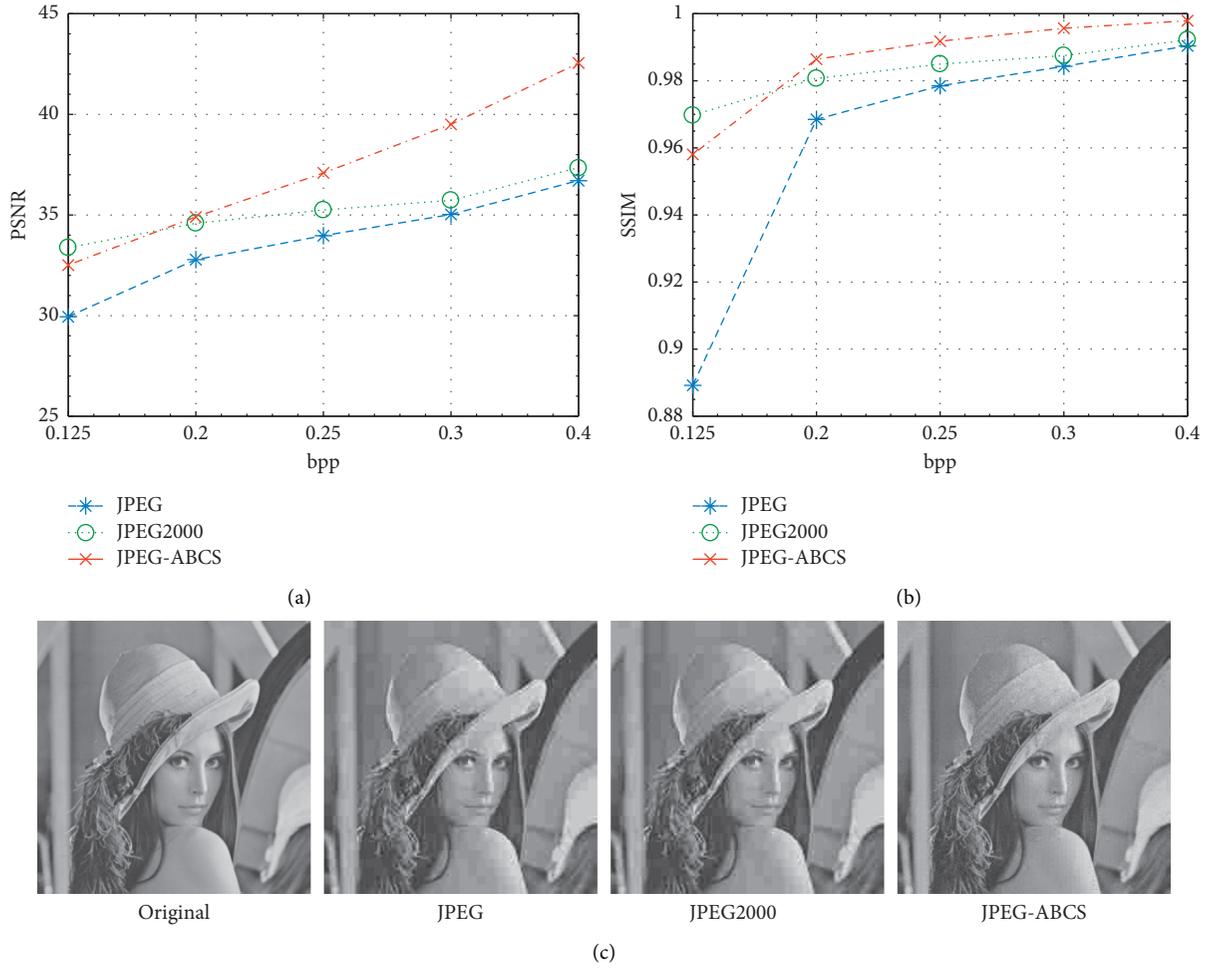
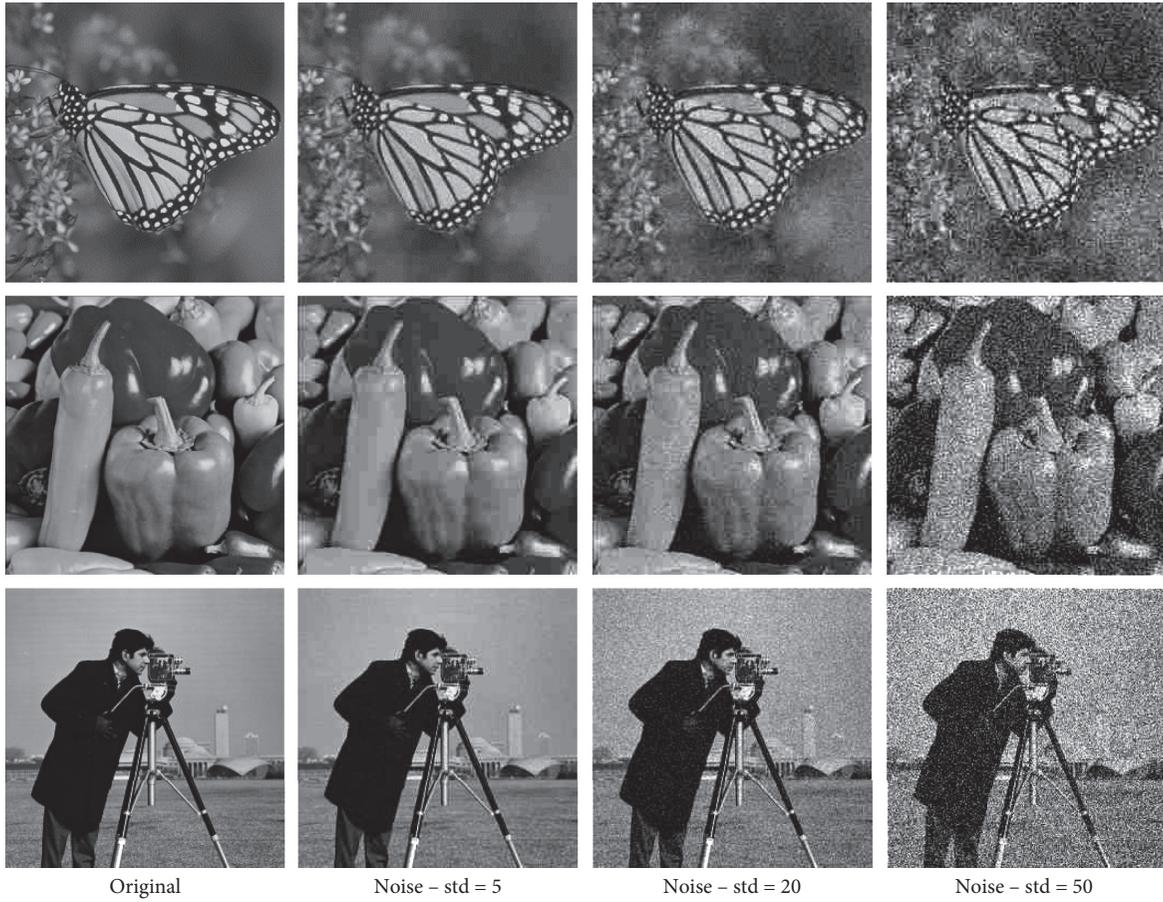


FIGURE 6: Comparative experiment of the three JPEG-like algorithms using Lena as a test image. (a) PSNR of the three algorithms in terms of different bpp. (b) SSIM of the three algorithms in terms of different bpp. (c) Restored images generated by three JPEG-like algorithms (bpp = 0.25).

TABLE 3: Comparison experiments of different vectorization methods on image reconstruction performance using texture test images.

Verification method			Vertical scanning	2D scanning	Zigzag scanning
Test image	η	$r \times c$		PSNR/SSIM	
	0.3	32 × 8	30.92/0.7101	31.52/0.7461	32.43/0.8244
		16 × 16	31.50/0.7620	32.04/0.7878	33.06/0.8407
		8 × 32	31.39/0.7433	31.54/0.7439	32.55/0.8291
	0.4	32 × 8	32.05/0.8195	32.51/0.8276	33.97/0.8846
		16 × 16	32.53/0.8430	32.91/0.8497	34.01/0.8823
		8 × 32	32.39/0.8377	32.27/0.8256	33.87/0.8851
ASM condition = 4					
	0.3	32 × 8	28.47/0.8046	28.84/0.8715	29.14/0.9033
		16 × 16	28.62/0.8179	29.01/0.8834	29.17/0.8967
		8 × 32	28.53/0.8113	28.90/0.8740	29.25/0.9119
	0.4	32 × 8	28.84/0.8659	29.44/0.9261	29.69/0.9365
		16 × 16	29.04/0.8817	29.65/0.9329	29.74/0.9353
		8 × 32	29.08/0.8941	29.34/0.9220	29.67/0.9354
ASM Condition = 3					



(a)

FIGURE 7: Continued.



FIGURE 7: Monarch, peppers, and cameraman restored images under different noise intensities (bpp = 0.4). (a) Using the JPEG algorithm. (b) Using the JPEG-ABCS algorithm.

TABLE 4: Performance comparison of the three algorithms in terms of different test images.

Test image		Lena	Peppers	PSNR/SSIM		Parrots
				Goldhill		
bpp = 0.25	JPEG	33.97/0.9785	33.17/0.9845	31.62/0.9685		35.80/0.9882
	JPEG2000	35.25/0.9850	34.32/0.9887	32.05/0.9718		36.52/0.9923
	JPEG-ABCS	37.09/0.9918	37.42/0.9962	32.29/0.9767		39.38/0.9966
bpp = 0.3	JPEG	35.02/0.9842	34.25/0.9890	32.09/0.9729		36.79/0.9912
	JPEG2000	35.73/0.9875	34.89/0.9914	32.42/0.9766		37.39/0.9940
	JPEG-ABCS	37.94/0.9936	39.42/0.9977	33.52/0.9848		39.78/0.9970
bpp = 0.4	JPEG	36.63/0.9902	35.76/0.9934	32.72/0.9789		38.27/0.9944
	JPEG2000	37.33/0.9922	36.33/0.9945	32.84/0.9814		39.11/0.9963
	JPEG-ABCS	42.01/0.9976	41.36/0.9985	33.84/0.9868		42.78/0.9985

TABLE 5: PSNR and SSIM of the monarch restored image using JPEG and JPEG-ABCS under different noise intensities.

Method	Condition	Noise – std = 5	Noise – std = 10	Noise – std = 20	Noise – std = 50
JPEG	bpp = 0.25	32.14/0.9452	32.03/0.9432	31.05/0.9309	28.39/0.7677
	bpp = 0.3	32.73/0.9601	32.45/0.9576	30.85/0.9387	28.27/0.7459
	bpp = 0.4	33.69/0.9726	32.94/0.9671	30.84/0.9373	28.44/0.7820
JPEG-ABCS	bpp = 0.25	35.25/0.9881	33.39/0.9780	31.28/0.9451	29.26/0.8269
	bpp = 0.3	36.48/0.9921	33.68/0.9803	31.46/0.9488	29.27/0.8245
	bpp = 0.4	36.74/0.9925	33.82/0.9809	31.43/0.9488	29.31/0.8290

The main reason for the above improvement is that the algorithm itself considers the noise model in the reconstruction algorithm and adds the idea of iterative optimization.

6. Conclusions

In this paper, a JPEG lifting algorithm based on ABCS was proposed, and its structure and implementation method were specifically introduced. At the same time, the improvements of the algorithm were described, and the feasibility and rationality of the above improvements were demonstrated by experiments. Finally, through comparison experiments with similar algorithms, the contribution of this lifting algorithm to JPEG-like algorithms, that is, to improve the quality of image reconstruction, reduce bit rate (bpp), and add the antinoise function has been evaluated

Data Availability

The simulation results used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the National Key Research and Development Program of the Ministry of Science and Technology of China (2018YFB2003304), the National Natural Science Foundation of China under Grant (61471191), and the Youth Program of the National Natural Science Foundation of China (31700478).

References

- [1] K. W. Gregory, "The JPEG still picture compression standard," *Communications of the ACM*, vol. 38, no. 1, pp. 18–34, 1992.
- [2] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: an overview," *IEEE Transactions on Consumer Electronics*, vol. 46, no. 4, pp. 1103–1127, 2000.
- [3] F. Dufaux, G. J. Sullivan, and T. Ebrahimi, "The JPEG XR image coding standard [standards in a nutshell]," *IEEE Signal Processing Magazine*, vol. 26, no. 6, pp. 195–204, 2009.
- [4] S. Yuan and J. Hu, "Research on image compression technology based on Huffman coding," *Journal of Visual Communication and Image Representation*, vol. 59, pp. 33–38, 2019.
- [5] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [6] Y. Zhou and H. Guo, "Collaborative block compressed sensing reconstruction with dual-domain sparse representation," *Information Sciences*, vol. 472, pp. 77–93, 2019.
- [7] S. H. Liu, J. B. Jia, and Y. J. Yang, "Image reconstruction algorithm for electrical impedance tomography based on block sparse Bayesian learning," in *Proceedings of the 2017 IEEE International Conference on Imaging Systems and Techniques*, IEEE, Beijing, China, pp. 267–271, October 2017.
- [8] P. Mazumdar, K. Lamichhane, M. Carli, and F. Battisti, "A feature integrated saliency estimation model for omnidirectional immersive images," *Electronics*, vol. 8, no. 12, p. 1538, 2019.
- [9] S. Mahdi, Y. R. Tohid, A. T. Mohammad et al., "Block sparse signal recovery in compressed sensing: optimum active block selection and within-block sparsity order estimation," *Theory Circuits System Signal Process*, vol. 37, no. 4, pp. 1649–1668, 2018.
- [10] Y. Niu, B. Tondi, Y. Zhao, and M. Barni, "Primary quantization matrix estimation of double compressed JPEG images via CNN," *IEEE Signal Processing Letters*, vol. 27, pp. 191–195, 2020.
- [11] I. Orovic, V. Papic, C. Ioana et al., "Compressive sensing in signal processing: algorithms and transform domain formulations," *Mathematical Problems in Engineering*, vol. 2016, Article ID 7616393, 16 pages, 2016.
- [12] L. Gan, "Block compressed sensing of natural images," in *Proceedings of the 15th Proceedings of the International Conference on Digital Signal Processing*, pp. 403–406, IEEE, Cardiff, UK, July 2007.
- [13] S. Zhou, Z. Chen, Q. Zhong, and H. Li, "Block compressed sampling of image signals by saliency based adaptive partitioning," *Multimedia Tools and Applications*, vol. 78, no. 1, pp. 537–553, 2019.
- [14] W. Rui, Z. X. Yu, L. F. Du et al., "Saliency-based adaptive block compressive sampling for image signals," *Journal of Image and Graphics*, vol. 18, no. 10, pp. 1255–1260, 2013.
- [15] Z. Zhang, Y. Shi, D. Kong, W. Ding, and B. Yin, "Image decoding optimization based on compressive sensing," *Journal of Computational and Applied Mathematics*, vol. 236, no. 5, pp. 812–818, 2011.
- [16] F. Wen, J. Shi, and Z. Zhang, "Joint 2D-DOD, 2D-DOA, and polarization angles estimation for bistatic EMVS-MIMO radar via PARAFAC analysis," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 2, pp. 1626–1638, 2020.
- [17] F. Q. Wen and J. P. Shi, "Fast direction finding for bistatic EMVS-MIMO radar without pairing," *Signal Processing*, vol. 173, pp. 34–36, 2020.
- [18] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, 1973.
- [19] H. Xu, J. Z. Xu, and F. Wu, "Lifting-based directional DCT-Like transform for image coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 10, pp. 1325–1335, 2007.
- [20] J. Shen, "Weber's law and weberized TV restoration," *Physica D: Nonlinear Phenomena*, vol. 175, no. 3–4, pp. 241–251, 2003.
- [21] A. Böttcher, "Orthogonal symmetric Toeplitz matrices," *Complex Analysis and Operator Theory*, vol. 2, no. 2, pp. 285–298, 2008.
- [22] Y. Z. Li, S. J. Yao, K. Yang et al., "A high-imperceptibility and histogram-shifting data hiding scheme for JPEG images," *IEEE Access*, vol. 7, pp. 73573–73582, 2003.
- [23] P. A. M. Oliveira, R. S. Oliveira, R. J. Cintra, F. M. Bayer, and A. Madanayake, "JPEG quantisation requires bit-shifts only," *Electronics Letters*, vol. 53, no. 9, pp. 588–590, 2017.
- [24] D. Meng, X. Wang, M. Huang, L. Wan, and B. Zhang, "Robust weighted subspace fitting for DOA estimation via block sparse recovery," *IEEE Communications Letters*, vol. 24, no. 3, pp. 563–567, 2020.

- [25] S. Beheshti and M. A. Dahleh, "Noisy data and impulse response estimation," *IEEE Transactions on Signal Processing*, vol. 58, no. 2, pp. 510–521, 2010.
- [26] M. Brajovic, I. Stankovic, M. Dakovic et al., "Error in the reconstruction of nonsparse images," *Mathematical Problems in Engineering*, vol. 2018, Article ID 4314527, 10 pages, 2018.
- [27] Y. Zhu, W. Liu, and Q. Shen, "Adaptive algorithm on block-compressive sensing and noisy data estimation," *Electronics*, vol. 8, no. 7, p. 753, 2019.
- [28] R. Qi, D. Yang, Y. Zhang, and H. Li, "On recovery of block sparse signals via block generalized orthogonal matching pursuit," *Signal Processing*, vol. 153, pp. 34–46, 2018.
- [29] T. Liu, F. Wen, L. Zhang, and K. Wang, "Off-grid DOA estimation for colocated MIMO radar via reduced-complexity sparse Bayesian learning," *IEEE Access*, vol. 7, pp. 99907–99916, 2019.