Article

Enhancing Performance of Lossy Compression on Encrypted Gray Images through Heuristic Optimization of Bitplane Allocation

Chuntao Wang 1,2,*, Renxin Liang 1, Shancheng Zhao 3, Shan Bian 1,2 and Zhimao Lai 4

1 College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510642, China; renxin123@stu.scau.edu.cn (R.L.); bianshan@scau.edu.cn (S.B.)
2 Guangzhou Key Laboratory of Intelligent Agriculture, Guangzhou 510642, China
3 College of Information Science and Technology, Jinan University, Guangzhou 510632, China; zhaoday@mail2.sysu.edu.cn
4 Immigration Management College (Guangzhou), China People’s Police University, Guangzhou 510663, China; laizhimao@cppu.edu.cn
* Correspondence: wangct@scau.edu.cn

Abstract: Nowadays, it remains a major challenge to efficiently compress encrypted images. In this paper, we propose a novel encryption-then-compression (ETC) scheme to enhance the performance of lossy compression on encrypted gray images through heuristic optimization of bitplane allocation. Specifically, in compressing an encrypted image, we take a bitplane as a basic compression unit and formulate the lossy compression task as an optimization problem that maximizes the peak signal-to-noise ratio (PSNR) subject to a given compression ratio. We then develop a heuristic strategy of bitplane allocation to approximately solve this optimization problem, which leverages the asymmetric characteristics of different bitplanes. In particular, an encrypted image is divided into four sub-images. Among them, one sub-image is reserved, while the most significant bitplanes (MSBs) of the other sub-images are selected successively, and so are the second, third, etc., MSBs until a given compression ratio is met. As there exist clear statistical correlations within a bitplane and between adjacent bitplanes, where bitplane denotes those belonging to the first three MSBs, we further use the low-density parity-check (LDPC) code to compress these bitplanes according to the ETC framework. In reconstructing the original image, we first deploy the joint LDPC decoding, decryption, and Markov random field (MRF) exploitation to recover the chosen bitplanes belonging to the first three MSBs in a lossless way, and then apply content-adaptive interpolation to further obtain missing bitplanes and thus discarded pixels, which is symmetric to the encrypted image compression process. Experimental simulation results show that the proposed scheme achieves desirable visual quality of reconstructed images and remarkably outperforms the state-of-the-art ETC methods, which indicates the feasibility and effectiveness of the proposed scheme.

Keywords: encrypted image compression; lossy compression; Markov random field; content-adaptive interpolation

1. Introduction

The encryption-then-compression (ETC) technique is a kind of technique that performs compression on encrypted digital media such as images and videos. It adapts to scenarios of cloud computing, distributed processing, etc., and is in contrast to the conventional compression-then-encryption (ETC) system that conducts compression prior to encryption. In an ETC system, the content owner only encrypts digital media for secrecy but does not compress them because of computing resource limitations. After the encrypted digital media have been uploaded to the cloud side, the cloud side compresses, without access to the decryption key, the encrypted digital media in order to save storage space and
communication bandwidth. After the compressed encrypted sequence is received, the receiver finally performs joint decompression and decryption to recover the original image in a lossless or lossy way.

With the content of digital media masked by encryption, it seems to be counter-intuitive to compress the encrypted digital media. By formulating this problem as distributed coding with side information at the decoder, however, Johnson et al. [1] demonstrated via the information theory that, compared with the conventional CTE system, the lossless ETC system neither degrades security nor sacrifices compression efficiency, while the lossy ETC system only approximates to a CTE system. In addition, they designed two toy examples to illustrate the feasibility of an ETC system.

Since then, a number of ETC schemes have been developed in the literature, most of which mainly focus on the compression and reconstruction of encrypted images and a few concentrate on that of encrypted videos. According to the reconstruction quality, they can be categorized into lossless [2–9] and lossy [10–26] approaches. The lossless approaches reconstruct the original image or video perfectly, whereas the lossy ones recover a degraded version of the original image or video.

As the lossy ETC method achieves larger compression ratios at the cost of tolerable reconstruction distortion, it has been well investigated in the literature. According to the compression method, lossy compression approaches for encrypted images and videos can be roughly classified into three types, i.e., the compressive sensing (CS)- [10–13], scalar quantization-[14–23], and uniform downsampling-based ones [24–26]. In implementing the encrypted image compression, they use a compressive sensing matrix [27], a scalar quantizer with a fixed or adaptive quantization step, and a uniform downsampling technique, respectively.

As is well known, the key to improving the compression ratio of unencrypted images and videos is to remove redundant content by exploiting statistical, human visual, or audio characteristics, etc. This is similar for the compression of encrypted images and videos. As analyzed in [20], in the ETC scenario, it is preferable to fully leverage the statistical characteristics of images and videos at the receiver. This is because the receiver has both the decryption key and sufficient computing resources, while the content owner has limited computing resources, and the cloud side cannot access the decryption key or may exploit statistical characteristics at the risk of statistical information leakage. For example, the two-dimensional Markovian property [2,3], resolution progressive strategy [6], and Markov random field (MRF) [9] are integrated in the reconstruction process of a lossless ETC system to improve the compression efficiency remarkably. In the situation of a lossy ETC system, the technique of content-adaptive interpolation (CAI) is generally incorporated at the receiver side to facilitate the reconstruction of missing pixels discarded in the compression process [24–26].

Guided by these notions and analysis, in this paper, we develop a novel ETC scheme for the lossy compression of gray images using CAI and MRF. Specifically, to better exploit statistical correlations between neighboring pixels, we adopt both pixel- and bitplane-level downsampling for encrypted image compression, and deploy MRF and CAI to recover missing bitplanes and thus pixels. At the pixel level, we perform uniform downsampling with a scaling factor of two in the horizontal and vertical directions on a stream-ciphered gray image, yielding four sub-images, namely $S_{00}$, $S_{01}$, $S_{10}$, and $S_{11}$, each of which is a quarter of the size of the original image. If the given compression ratio is smaller than $1/4$, some of the bitplanes of $S_{00}$ are selected to meet the given compression ratio and the compression process is finished; otherwise, $S_{00}$ is reserved and the bitplane-level downsampling is further conducted. At the bitplane level, partial bitplanes of these sub-images are preserved while the others are discarded. To achieve better reconstruction quality, we formulate the bitplane-level downsampling as an optimal problem, maximizing the peak signal-to-noise ratio (PSNR) on the condition of a given compression rate. As higher bitplanes (e.g., MSB) are more important than lower bitplanes (e.g., LSB) in image reconstruction, we are motivated to solve this optimization problem through a heuristic
strategy, e.g., maintaining the first, most significant bitplane (MSB) of these three sub-images successively, then preserving the second MSBs, and so on. In other words, we exploit the asymmetric characteristics of different bitplanes in optimal bitplane selection.

As there exist clear statistical correlations both within any bitplane of the first three MSBs and between adjacent bitplanes, we further leverage an MRF to characterize these statistical correlations [8,9]. Based on this, we then use the low-density parity-check (LDPC) code according to the framework of Johnson et al. [1] to compress the first three MSBs of sub-images $S_{00}$, $S_{01}$, $S_{10}$, and $S_{11}$. This would further enhance the compression efficiency of the encrypted images.

After obtaining the downsampled encrypted image, the receiver conducts de-compression and decryption to recover the original gray image in a lossy way, which is actually symmetric to the process of encrypted image compression. In particular, LDPC-coded bitplanes are first reconstructed perfectly via the joint LDPC decoding, decryption, and MRF exploitation [8,9], and CAI [24] is then deployed to generate missing bitplanes and thus pixels discarded in the compression process.

With the stream cipher for encryption, the pixel- and bitplane-level downsampling for compression, and the MRF and CAI for lossy reconstruction, we thus propose a novel ETC scheme for the lossy compression of encrypted gray images. Experimental results show that the proposed scheme results in desirable visual quality for a reconstructed image and achieves a significant improvement over the state-of-the-art, indicating the feasibility and effectiveness of the proposed scheme.

In summary, the contributions of the paper are two-fold: (1) We formulate the task that maximizes the PSNR of an image reconstructed by a downsampling-based ETC system on the condition of a given compression ratio as an optimization problem, and develop a heuristic bitplane allocation strategy to solve this optimization problem; (2) We propose a novel ETC scheme for lossy compression on encrypted gray images through heuristic optimization of bitplane allocation, achieving a remarkable improvement over the state-of-the-art.

The remainder of the paper is arranged as follows. Related works are introduced in Section 2. The proposed ETC scheme is described in Section 3. Experimental results and conclusions are given in Sections 4 and 5, respectively.

2. Related Works

2.1. Lossy Compression on Encrypted Images

As this paper mainly focuses on the lossy compression of encrypted images, lossy compression methods for encrypted images and videos in the literature are briefly described below.

As mentioned in Section 1, the lossy ETC methods in the literature can be roughly classified into three categories, namely the compressive sensing (CS)- [10–13], scalar quantization- [14–25], and uniformly downsampling-based [24–26] ones. For the CS-based ETC method, the conventional measurement matrix of CS [10,13], gradient projection matrix [11], and learned dictionary [12] is adopted to compress encrypted images, and the modified basis pursuit is developed to reconstruct the original gray image.

For the quantization-based method, a scalar quantizer is applied to quantize an encrypted image, aiming to reduce the data amount, and the successive de-quantization and content-adaptive interpolation (CAI) are generally used to reconstruct the original image in a lossy way. For example, Gschwandtner et al. [14] employed quantizers with different steps and the Huffman coding to implement the compression of encrypted gray images. Zhang et al. [15] conducted compression by discarding rough and fine details of transformed coefficients and quantizing reserved coefficients. Zhang et al. [16] formed based and enhanced layers of an encrypted gray image, followed by quantizing the enhanced layer to implement compression. Later, Zhang et al. [17] decomposed a gray image into multi-layers and quantized the permuted prediction errors of different layers. Wang et al. quantized lifting wavelet-transformed coefficients via scalar quantizers that are
constructed in a heuristic \[20\], rate-distortion optimization \[21\], or weighted rate-distortion optimization \[22\] way.

For the uniform downsampling-based category \[24–26\], an encrypted gray image is uniformly downsampled to achieve compression. At the receiver, the CAI \[24,25\] or MRF \[26\] is incorporated to reconstruct the original image in a lossy way.

2.2. MRF-Based ETC Method

As aforementioned in Section 1, we employ an MRF to characterize statistical correlations both within a bitplane \[8\] and between adjacent bitplanes \[9\], and apply it to improve the compression efficiency. Thus, in this subsection, we mainly introduce MRF and MRF-based ETC methods \[8,9\].

2.2.1. MRF Introduction

MRF is a kind of statistical model characterizing the spatial statistics of a natural image. It has been widely applied in many research fields, such as image denoising, segmentation, and computer stereo vision.

Suppose that \(I\) is a \(Q\)-bit image of size \(m \times n\) and \(L = \{(x, y), x \in [1, n], y \in [1, m]\}\) denotes the corresponding coordinate set, where \(Q\) is the bit depth. If each pixel \(I(x, y)\) is represented with a random variable, \(F_s(s \in L)\), that takes on values in the state space, \(\Phi = \{0, 1, \cdots, 2^{Q-1}\}\), then all \(F_s\)s form a random field, \(F = \{F_s|s \in \Phi, s \in L\}\). If the random field \(F\) further meets positivity and Markovian, i.e., \(p(F = F) > 0\) for \(\forall F \in F\) and \(p(F_s|F_{L-s}) = p(F_s|F_{\delta(s)})\), respectively, then \(F\) becomes an MRF, where \(L - s\) and \(\delta(s)\) denote the coordinate set excluding \(s\) and the neighborhood of \(s\), respectively.

With the Hammersley–Clifford theorem, MRF is equivalent to the Gibbs random field and thus can be equivalently calculated as \[8\]:

\[
p(F = F) = \frac{1}{Z} \exp\left(-\frac{U(F)}{T}\right)
\]

where \(Z\), \(U(F)\), and \(T\) are a normalizing constant, an energy function equal to \(U(F) = \sum_{c \in C} V_c(F)\), and a temperature constant, respectively. The \(C\) denotes a set of cliques formed by the neighborhood system \(\delta(s)\), and \(V_c(\cdot)\) is a potential function defined on a given clique \(c \in C\).

To compute \(p(F = F)\) in an efficient way, MRF is first represented with a factor graph and the sum-product algorithm is then applied on the represented factor graph to obtain \(p(F = F)\) \[21,28\]. When the first-order neighboring system is considered, the factor graph for MRF can be constructed as follows \[21\]. Specifically, take a variable node (VN) to represent each pixel of a given image, and use a factor node (FN) to characterize the statistical correlations between neighboring VNIs. Figure 1 illustrates the factor graph for a \(3 \times 3\) image, where VN and FN are denoted with a circle and square, respectively. A VN takes on values in \(\Phi = \{0, 1, \cdots, 2^Q - 1\}\) and an FN stands for the potential function \(U(F)\) in Equation (1).

Figure 1. Illustration of a factor graph for the MRF of a \(3 \times 3\) image.
If a gray image is considered as a combination of eight bitplanes, each bitplane can then be taken as a binary image and thus characterized with a factor graph for a binary image (i.e., \( Q = 1 \)). As investigated in [9], there exist clear statistical correlations within each bitplane and between adjacent bitplanes, where bitplanes denote those belonging to the first three MSBs. Therefore, MRF can be further applied to characterize statistical correlations between adjacent bitplanes, and the corresponding factor graph is illustrated in Figure 2.

\[ \text{Figure 2. Illustration of a factor graph for the MRF between adjacent bitplanes.} \]

\( B^k \) and \( B^{k-1} \) denote two adjacent bitplanes, \( F^k(x, y) \) and \( F^{k-1}(x, y) \) are two VNs at the same coordinate of \( B^k \) and \( B^{k-1} \), and \( D^k(x, y) \) is an FN representing the statistical correlation between \( F^k(x, y) \) and \( F^{k-1}(x, y) \).

2.2.2. Joint Factor Graph for Reconstruction of Bitplanes

In the ETC framework developed by Johnson et al. [1], a stream cipher is taken for encryption, a channel code such as LDPC can be used for the compression of encrypted data, and a joint decoding and decryption process is conducted to recover the original data. Under this framework, Wang et al. exploited an MRF to characterize statistical correlations within a bitplane [8] and between adjacent bitplanes [9]. They then constructed factor graphs for stream cipher-based encryption, LDPC-based compression, and MRF exploitation, and combined these three factor graphs seamlessly to form a joint factor graph for the reconstruction of a binary image (BIRFG) [8]. Based on BIRFG, they further incorporated the factor graph for statistical correlations between adjacent bitplanes to form a joint factor graph for gray image reconstruction (JFGIR) [9]. Both BIRFG and JFGIR are briefly described as follows.

Figure 3 illustrates BIRFG. It consists of three factor graphs, i.e., the factor graphs for LDPC decoding, decryption, and MRF exploitation, which are plotted in boxes with solid, dot, and dot-and-dash lines, respectively. The factor graph of LDPC decoding is used to decompress the received encrypted and compressed sequence, the graph of decryption is deployed to decrypt the decompressed but encrypted signal, and that of MRF exploitation is incorporated to exploit the statistical correlations of natural images.
Figure 3. BIRFG for binary image $I$ of size $m \times n$. $S_j (j = 1, \ldots, q)$ are LDPC syndrome bits, which is taken as the encrypted and decompressed bit sequence; $\hat{Y}_i (i = 1, \ldots, mn)$ is the decompressed but encrypted sequence, $K_i$ is the encryption key sequence, $\hat{F}_i$ is a 1-D bit sequence converted from a given binary image, and $\hat{F}_{y,x}$ denotes pixels of 2-D binary image. $M_{y,x}$, $N_{y,x}$, $P_{y,x}$, $t$, and $g_j$ represent the constraints imposed by the potential function, image source prior, decryption, and LDPC code, respectively.

By using BIRFG to characterize bitplanes $B_k$ and $B_{k-1}$ in Figure 2 and deploying $D_k(x, y)$ to represent statistical correlations between bitplanes, JFGIR is thus formed accordingly. For compactness, it is omitted here.

With the constructed joint factor graph, the sum-product algorithm is then applied iteratively. After the iteration converges, the original image can thus be reconstructed. For details of the sum-product algorithm on BIRFG and JFGIR, readers are referred to [8] and [9], respectively.

3. Proposed ETC Scheme

In this section, we present the proposed ETC scheme through heuristic optimization of bitplane allocation, as illustrated in Figure 4. It includes gray image encryption via a stream cipher, encrypted image compression with heuristic bitplane allocation optimization, and lossy reconstruction using MRF and content-adaptive interpolation (CAI). Below are details for these parts.

Figure 4. Flowchart of the proposed ETC scheme, where $\oplus$ denotes an exclusive or operation.

3.1. Stream Cipher-Based Encryption

Suppose that $I$ is a gray image of size $m \times n$. Then, the $k$th bitplane of image $I$, namely $B_k$, is obtained as

$$B_k(x, y) = \left\lfloor \frac{I(x, y)}{2^{k-1}} \right\rfloor \% 2, \quad k = 1, \ldots, 8 \quad (2)$$

It is clear that $B_8$ and $B_1$ denote the most significant bitplane (MSB) and least significant bitplane (LSB), respectively.

Next, generate a pseudorandom bit sequence of length $m \times n$, say $K_k = \{K_k(i) | K_k(i) \in 0, 1 | i = 1, \ldots, m \times n \}$, via the $k$th secret key $KEY_c + 2^k$, where $KEY_c$ is a one-time-pad initial secret key. Afterwards, encrypt each bitplane with $K_k$ as

$$Y_k(x, y) = I_k(x, y) \oplus K_k(i), \quad i = (y-1)n + x. \quad (3)$$

where $I_k$ denotes the $k$th bitplane of image $I$ and $\oplus$ is an exclusive or operation.
After encryption bitplane-by-bitplane, all encrypted bitplanes are merged in the following way to yield the stream ciphered gray image \( Y \), i.e.,

\[
Y(x, y) = \sum_{k=1}^{8} Y^k(x, y) \times 2^{k-1}.
\]

(4)

where \( Y^k \) represents the \( k \)th bitplane of image \( Y \). The encrypted \( Y \) is sent through a public channel to the cloud side, and the secret key \( KEY_c \) is transmitted via a secure channel to the receiver.

### 3.2. Compression via Heuristic Optimization of Bitplane Allocation

To save storage space and communication bandwidth, the cloud side chooses to compress the encrypted gray image \( Y \) on the condition of no decryption key. As the encryption has masked the image content, the cloud side without access to the decryption key cannot exploit the statistical characteristics of the original unencrypted image to conduct compression. As introduced in Section 2, to achieve feasible compression, the LDPC syndrome can be used for the lossless compression of a stream ciphered binary image, and the compressive sensing matrix, scalar quantizer, or uniform downsampling can be applied for the lossy compression of a stream ciphered binary or gray image.

As uniform downsampling generally achieves relatively large compression ratios at tolerable distortions and is rather suitable for the cloud side without the decryption key to conduct compression on images encrypted by random permutation, stream cipher, cryptography, etc., it is adopted in our work as the compression method. To facilitate CAI at the receiver, we employ the uniform downsampling with a scaling factor of two. In particular, for a stream ciphered image \( Y \), we divide it into four sub-images, namely \( B_{00}, B_{01}, B_{10}, \) and \( B_{11} \), according to Figure 5, i.e.,

\[
B_{ij}(x, y) = Y(2x + 1 + i, 2y + 1 + j), x \in [0, n - 1], y \in [0, m - 1], i, j \in \{0, 1\}
\]

(5)

Without loss of generality, we take \( B_{00} \) as the uniformly downsampled result. This leads to an initial compression ratio of \( R = 0.25 \).

Figure 5. Illustration of sub-image extraction, where pixels with the same symbol are uniformly downsampled as a sub-image.

If the target compression ratio \( R_0 \) is larger than 0.25, then more bitplanes are needed to choose from \( B_{01}, B_{10}, B_{11} \) and sent to the receiver. If \( R_0 \) is less than 0.25, then more bitplanes are required to discard from \( B_{00} \). This thus gives rise to a key problem, i.e., how to select suitable bitplanes from sub-images so that the reconstruction performance in terms of PSNR can be maximized on the condition of a given compression ratio \( R_0 \).

This problem is essentially a rate-distortion optimization problem. As the statistical characteristics of the original image generally cannot be exploited at the cloud side, this rate-distortion optimization cannot be directly implemented at the cloud side. In an alternative way, the cloud side can choose to design a preferable bitplane selection method, with which the receiver can well exploit the selected bitplanes to recover discarded parts. As a result, the reconstruction quality may be maximized at a given compression ratio.
For a given $R_0$ larger than 0.25, it is clear that the more important the bitplane is for reconstruction, the higher the priority in bitplane selection should be. As MSB is the most important bitplane in image reconstruction and LSB is the least important one, we choose to optimize bitplane allocation in a heuristic way. Specifically, we conduct the bitplane selection as follows.

1. Select the MSB of $B_{11}$ and compute the compression ratio as $R = 0.25 + 1/32 = 0.28125$. If $R > R_0$ holds, then $(R - R_0) \times 8 \times m \times n$ bits are randomly discarded from the MSB of $B_{11}$, where discarding locations can be generated through a secret key $KEY_s$. Otherwise, go to Step 2.

2. Choose the MSB of $B_{01}$ and calculate $R = 0.28125 + 1/32 = 0.3125$. Then, process it in a way similar to Step 1 to either discard some bits or go to Step 3.

3. Extract the MSB of $B_{10}$ and obtain $R = 0.3125 + 1/32 = 0.34375$. Next, remove some of the bits or go to Step 4.

4. Sample the second MSBs of $B_{11}$, $B_{01}$, and $B_{10}$ sequentially in a manner similar to Steps 1–3. Even sample their third MSBs, fourth MSBs, and so on. Repeat this process until $R_0$ is met.

For a given $R_0$ smaller than 0.25, bitplanes in $B_{00}$ should be discarded. In this situation, the LSB bitplane of $B_{00}$ is first removed, and the compression ratio is updated as $R = 0.25 - 1/32 = 0.21875$. If $R < R_0$ holds, then $(R - R_0) \times 8 \times m \times n$ bits in LSB are randomly retained. Otherwise, continue to remove the second MSB, the third one, and so on until $R_0$ is achieved.

Although this heuristic optimization of bitplane allocation can probably maximize PSNR at a given compression ratio, we may further improve the compression efficiency by exploiting the statistical correlations that exist within a bitplane and between adjacent bitplanes, where bitplanes belong to the first three MSBs. Specifically, by incorporating uniform downsampling and statistical correlation exploitation, we conduct the compression of an encrypted gray image as follows.

1. Divide the stream ciphered gray image $Y$ into four sub-images $B_{00}$, $B_{01}$, $B_{10}$, and $B_{11}$ via the method in Figure 5.

2. Compress the MSB of $B_{00}$ through LDPC coding [8], i.e.,

   \[
   S^8_{00} = HB^{1D}_{00}.
   \]

   where $H$ is a parity-check matrix, $B^{1D}_{00}$ is a one-dimensional ($D$) vector converted from the 2-D matrix $B_{00}$, and $S^8_{00}$ is the syndrome used as the compressed sequence for $B_{00}$. If the second and third MSBs of $B_{00}$ are selected, they are then compressed similarly, yielding Syndrome $S^5_{00}$ and $S^4_{00}$, respectively.

   As there exist weak statistical correlations within and between the other five bitplanes, these bitplanes are not compressed via the LDPC coding and directly sent to the receiver. For notational convenience, these five bitplanes are also denoted $S^k_{00}$ ($k = 5, \ldots, 1$).

3. Compute the compression ratio, $R_{acc}$, for sub-image $B_{00}$ as

   \[
   R_{acc} = \frac{\sum_{k=1}^{8} \text{length}(S^k_{00})}{8 \times m \times n},
   \]

   where $\text{length}(\cdot)$ denotes a function counting the bit number. If the given compression ratio $R_0$ is smaller than $R_{acc}$, then determine the minimum $k$ via the following optimization, i.e.,

   \[
   \arg \min_{k \in \{1, \ldots, 8\}} \frac{\sum_{i=k}^{8} \text{length}(S^i_{00})}{8 \times m \times n} > R_0.
   \]
Next, transmit the syndrome sequence of $S_{00}^0 (i = k, \ldots, 8)$ to the receiver and terminate the compression process. Otherwise, send the syndrome sequence of $S_{00}^k (k = 1, \ldots, 8)$ to the receiver and go to the next step.

(4) Compress the MSB of $B_{11}$ by LDPC coding and generate syndrome $S_{11}^8$. If $R_{acc} = R_{acc} + \frac{\text{length}(S_{11}^8)}{8 \times m \times n} > R_0$ holds, then further send $S_{11}^8$ to the receiver and end the compression. Otherwise, go to the next step.

(5) Impose the LDPC coding on the MSB of $B_{01}$ and yield syndrome $S_{01}^1$. If $R_{acc} = R_{acc} + \frac{\text{length}(S_{01}^1)}{8 \times m \times n} > R_0$ holds, then transmit $S_{01}^1$ to the receiver and complete the compression. Otherwise, go to the next step.

(6) Conduct similar processing on the MSB of $B_{10}$. If $R_{acc} = R_{acc} + \frac{\text{length}(S_{10}^8)}{8 \times m \times n} > R_0$ holds, then send $S_{10}^8$ to the receiver and finish the compression. Otherwise, go to the next step.

(7) Similar to Steps 4–7, condense the second and third MSBs of $B_{11}, B_{01}$, and $B_{10}$ successively until the given compression ratio $R_0$ is approximately satisfied.

(8) Similar to the compression of bitplanes $B_{0i}^k (k = 1, \ldots, 5)$, the other five bitplanes of $B_{11}, B_{01}$, and $B_{10}$ are not LDPC-coded and denoted $S_{11}^k, S_{01}^k$, and $S_{10}^k$ for notational convenience, respectively. If $R_0$ is sufficiently large, $S_{11}^k, S_{01}^k$, and $S_{10}^k$ are chosen sequentially from $k = 5$ to $k = 1$ until $R_0$ is nearly achieved.

It is noted that this compression procedure leads to a compression ratio approximating but not exactly the same as $R_0$, aiming to trade-off the complexity in bitplane selection and the exact compression ratio. If an exact compression ratio is required, optimal bit selection is required, which needs much more investigation and thus can be considered an interesting topic for future work.

After the compression is completed, the selected syndrome sequence, say $S = \{S_{00}^0, S_{00}^7, \ldots\}$, is transmitted through a public channel to the receiver, while the secret key $KEY_s$ is sent via a secret channel. It is worth pointing out that the $KEY_s$ can be used to generate all related coordinates of randomly discarded or supplemented bits by yielding sub-keys such as $KEY_s + 2^s + 2^k$, where $s = 0, 1, 2, 3$ denotes the label of a sub-image and $k = 1, \ldots, 8$ stands for the number of bitplanes.

In addition, to facilitate LDPC decoding at the receiver, we need to tell the receiver about the number of sent bitplanes in each sub-image, and the number of syndrome bits for the three MSBs of each sub-image. As each sub-image contains eight bitplanes, three bits are enough to represent the bitplane number. As the size of each bitplane of a sub-image is $m \times n/4$, a total of $\text{len} = \lceil \log_2 (m \times n/4) \rceil$ bits is sufficient to record the number of syndrome bits of each bitplane of a sub-image. Therefore, by incorporating this auxiliary information, the final compression ratio $R$ is calculated as

$$R = \frac{\text{length}(S) + 3 \times 4 + \text{len} \times t}{8 \times m \times n}$$

where $t (0 \leq t \leq 12)$ is the number of LDPC-coded MSBs of all four sub-images. Moreover, as bitplanes other than the three MSBs are not compressed, their sizes are fixed to $m \times n/4$ and thus do not need to be recorded.

3.3. Reconstruction via MRF Exploitation and Interpolation

After receiving the compressed encrypted sequence $S$, the auxiliary information indicating the transmitted bitplane number and the LDPC-coded sequence length, and secret keys $KEY_c$ and $KEY_s$, the receiver performs decompression and decryption to reconstruct the original image in a lossy way.

First of all, from the received auxiliary information, the receiver determines the number of received bitplanes of each sub-image and the length of LDPC-coded bitplanes. Via these parameters, the syndrome sequence of each LDPC-coded bitplane is extracted from $S$, and the corresponding bitplane is then recovered in a lossless way via the joint
LDPC decoding, decryption, and MRF exploitation (see also Section 2.2.2) [9]. Similarly, the non-compressed bitplanes (i.e., the first to fifth LSBs) are obtained from S and then recovered accordingly. For the other bitplanes that are not sent from the cloud side, all are set with zeros. After bitplanes have been recovered, they are merged to yield four reconstructed sub-images: \( B'_{00}, B'_{01}, B'_{10}, \) and \( B'_{11} \).

As there exist clear statistical correlations between pixels in the neighborhood, we further employ CAI [29] to reconstruct missing parts. In more detail, we use \( B_0 \) to interpolate \( B'_{11}, B'_{01}, \) and \( B'_{10} \) successively. Figure 6 illustrates the interpolation in our work. Specifically, at the first stage, we deploy \( B_0 \) to interpolate \( B'_{11} \). Suppose that \( t = [t_1, t_2, t_3, t_4] \) are neighboring pixels in the diagonal directions of pixel \( B_{11}'(x, y) \), where \( t \) belongs to pixels in \( B'_{00} \) whereas \( t \) is located in \( B'_{11} \). Then, we use \( t \) to predict \( B_{11}'(x, y) \) as

\[
v_0 = \begin{cases} \text{mean}(t), & \text{if } \max(t) - \min(t) < 20 \\ \frac{t_1 + t_2}{2}, & \text{if } |t_3 - t_4| - |t_1 - t_2| > 20 \\ \frac{t_1 + t_4}{2}, & \text{if } |t_1 - t_2| - |t_3 - t_4| > 20 \\ \text{median}(t), & \text{otherwise} \end{cases}
\]  

where \( \text{mean}(\cdot) \), \( \max(\cdot) \), \( \min(\cdot) \), and \( \text{median}(\cdot) \) are functions generating the mean, maximum, minimum, and median values of a given vector.

![Figure 6. Interpolation illustration, where 00, 01, 10, and 11 denote \( B'_{00}, B'_{01}, B'_{10}, \) and \( B'_{11} \), respectively. The circle, pentagram, triangle, and circular polygon stand for pixels in sub-image \( B_{00}' \), \( B_{01}' \), \( B_{10}' \), and \( B_{11}' \), respectively.](image)

We proceed to improve the prediction accuracy for \( B'_{11}(x, y) \). Assume that \( N(N > 0) \) MSBs of \( B'_{11} \) are recovered in a lossless way from the received sequence \( S \). Then, they can be exploited to improve the prediction result \( v_0 \). In more detail, first, obtain the first \( N \) MSB bits from pixel magnitude \( B_{11}'(x, y) \) and \( v_0 \), and then converge them to decimal values \( d_0' \) and \( d_0 \), respectively, where \( d_0', d_0 \in [0, M - 1] \) with \( M = 2^N \). If \( d_0 \) meets \( M/2 < |d_0| < 3M/4 \), then the prediction value \( v_0 \) deviates from the original value significantly.

To tackle this issue, another two prediction values are generated as

\[
\begin{align*}
v_1 &= \frac{\text{sum}(t) - \max(t)}{3} \\
v_2 &= \frac{\text{sum}(t) - \min(t)}{3}
\end{align*}
\]  

where \( \text{sum}(\cdot) \) sums up all elements of a given vector. The \( N \) MSB bits of \( v_1 \) and \( v_2 \) are also extracted to yield the corresponding decimal values \( d_1 \) and \( d_2 \), respectively. Value \( d_j (j = 0, 1, 2) \) leading to the minimum difference is taken as the optimum one, \( d_{\text{opt}} \), i.e.

\[
d_{\text{opt}} = \arg\min_{d_j \in \{0, 1, 2\}} \min(|d_j - d_0'|, M - |d_j - d_0|)
\]  

The value yielding \( v_{\text{opt}} \) is thus considered the practically optimum interpolation value.
Via the $d_{\text{opt}}$, the difference with respect to $d'_0$ is calculated as $\gamma_{\text{opt}} = d_{\text{opt}} - d'_0$. Thus, the final interpolation result for $B'_{11}(x,y)$ is obtained as

$$B'_{11}(x,y) = \begin{cases} d_{\text{opt}} \times \Delta + \frac{d_{\text{opt}} \times \Delta}{M} - \Delta + \frac{\Delta}{M} - 1, & \text{if } \gamma_{\text{opt}} < -\frac{M}{2} \\ d_{\text{opt}} \times \Delta + \frac{d_{\text{opt}} \times \Delta}{M} + \Delta, & \text{if } \gamma_{\text{opt}} > \frac{M}{2} \\ d_{\text{opt}} \times \Delta + \frac{d_{\text{opt}} \times \Delta}{M} + \text{mod}(\nu_{\text{opt}}, \frac{\Delta}{M}), & \text{otherwise} \end{cases}$$

(13)

where $\Delta = 2^{8-N}$ and $\text{mod}(\cdot, \cdot)$ is a modulo function.

After generating $B'_{11}$, we then use it as well as $B'_{00}$ to interpolate $B'_{01}$ and $B'_{10}$ successively. This is rather similar to the interpolation of $B'_{11}$, as shown in Figure 6.

When sub-images $B'_{11}$, $B'_{01}$, and $B'_{10}$ are reconstructed, they are merged according to Figure 6 to yield the reconstructed gray image $I'$, which is an inverse process of sub-image extraction, as illustrated in Figure 5.

4. Experiments and Analysis

In this section, we assess the proposed scheme for the lossy compression and reconstruction of gray images. We first give suitable parameters for the proposed scheme; then, we evaluate the visual quality of the reconstructed images and, finally, examine the rate-PSNR (peak signal-to-noise ratio) performance. Below are details of the experimental simulations and results.

4.1. Experimental Settings

The proposed scheme involves an MRF in bitplane reconstruction. As adopted in [8], the discontinuity-adaptive model [30] is also applied in our scheme. It includes three model parameters, i.e., the sharpness-controlled model parameter of $\delta$, the temperature constant of $T$, and the source prior of $P$. As degrees of statistical correlations within a bitplane and between adjacent bitplanes are different, different mode parameters are set. Specifically, for MRF within a bitplane, we deploy the practically optimal model parameters explored in [8], i.e., $\delta = 45$, $T = 0.00049$, and $P = 0.5$. For MRF between adjacent bitplanes, we use the practically optimum model parameters set in [9], i.e., $\delta = 45$ and $P = 0.5$, and $T = 0.05$ and 0.05 for MRF between the 8th and 7th MSBs and that between the 7th and 6th MSBs, respectively.

In compressing the first three MSBs, the LDPC code is adopted. The length of the LDPC code is configured as 4096. The compression rate of the LDPC code is set as $R \in [0.03, 0.95]$ with step 0.025. The binary search is applied on this rate set, and the minimum compression rate leading to successful compression and lossless reconstruction is then taken as the optimal compression rate. It is noted that different MSBs would have different optimal compression rates.

In the simulation, we test eight $512 \times 512$ gray images of different texture and edges, namely Baboon, Barb, Boat, Hill, Lena, Man, Peppers, and Tank, as illustrated in Figure 7. Each test image is encrypted, compressed, and reconstructed via the proposed scheme in Section 3, in which the to-be-LDPC-coded bitplane of size $256 \times 256$ is divided into 16 non-overlapped blocks of size $64 \times 64$ and each block is compressed via Equation (6). The reconstruction performance is evaluated with PSNR and compression ratio (CR), where CR is calculated via Equation (9).
4.2. Visual Quality Evaluation of Reconstructed Images

As described in Section 3, by sending different numbers of bitplanes, we achieve different compression ratios and obtain reconstructed images with various PSNRs and thus different visual quality. To evaluate the visual quality of the reconstructed images, we take images “F16” and “Lena” as examples.

Figure 8 illustrates the reconstructed images at different compression ratios for images “F16” and “Lena”. It is shown that the reconstructed images have feasible visual quality even at a low compression ratio of 0.2. Moreover, by comparing with Figure 8a or c and b or d, one can find that with the increase in compression ratio, more details are saved in the compression process and thus better visual quality can be achieved for the reconstructed image. This is because the proposed scheme fully exploits the statistical correlations within a bitplane, between adjacent bitplanes, and among neighboring pixels, and thus well recovers missing parts during the compression of encrypted gray images.
4.3. Rate-PSNR Performance Assessment

To further assess the proposed scheme, we examine the rate-PSNR performance and compare it with the state-of-the-art ETC approaches presented in [23–25]. For notational convenience, these approaches are denoted KANG, ZHOU, and QIN, respectively. For fair comparison, parameter settings in Section 4.1 are adopted for the proposed scheme, and the optimal parameters given in [23–25] are applied for KANG, ZHOU, and QIN, respectively.

In the simulation, the test images in Figure 7 are used for all four compared methods. After each test image is processed with the corresponding encryption, compression, and reconstruction algorithms, the rate-PSNR performance is evaluated. Figure 9 plots the rate-PSNR performance for KANG, ZHOU, QIN, and the proposed scheme. It is found that the proposed scheme generally outperforms KANG, ZHOU, and QIN remarkably. Considering that KANG, ZHOU, and QIN all deploy content-adaptive interpolation (CAI) at the receiver, and ZHOU and QIN further optimize the encrypted image compression through context-adaptive sampling and reconstruction via the inpainting technique, respectively, these simulation results well demonstrate the feasibility and effectiveness of the proposed scheme.

In addition, when compared to KANG, the proposed scheme achieves a significant improvement. Although both of them apply the same CAI method, the proposed scheme conducts the heuristic optimization of bitplane allocation, as well as the exploitation of the
statistical correlations within a bitplane and between adjacent bitplanes, and consequently improves the reconstruction performance at the same compression ratio.

Figure 9. Rate-PSNR performance comparison for the proposed scheme, KANG, ZHOU, and QIN. (a–i) present the rate-PSNR performance of all compared methods for images Baboon, Bar, Boat, F16, Hill, Lena, Man, Peppers, and Tank, respectively.

5. Conclusions

In this paper, we present a lossy compression scheme on encrypted gray images through heuristic optimization of bitplane allocation. For a stream ciphered gray image, we develop a pixel- and bitplane-level downsampling method to perform the compression. At the pixel level, the uniform downsampling technique is employed, which yields four sub-images with a quarter of the size and may discard part of the sub-images according to the given compression ratio. At the bitplane level, the task of bitplane selection-based compression is first formulated as an optimization problem that chooses suitable bitplanes to maximize the PSNR of the reconstructed image subject to a given compression ratio, and then a heuristic strategy is developed to approximately solve this optimization problem. For the LDPC-coded bitplanes among the first three MSBs, we further exploit an LDPC code to compress them. In the reconstruction stage, bitplanes among the first three MSBs are recovered in a lossless way via the joint LDPC decoding, decryption, and MRF exploitation. They are then merged to form sub-images, and content-adaptive interpolation is deployed to reconstruct missing bitplanes and thus discarded pixels. Experimental simulation results show that the proposed scheme outperforms the state-of-the-art ETC methods, indicating the feasibility and effectiveness of the proposed scheme.

Although the proposed ETC scheme is promising, it could be further improved in future investigation. For example, each sub-image generated in this work may be further downsampling in a multi-scale way, aiming to better exploit the statistical correlations...
between neighboring pixels to improve the compression efficiency. In addition, as some of the MSBs of the sub-images are sent to the receiver, they could be fully used to improve the accuracy of content-adaptive interpolation. Furthermore, the proposed scheme may be extended to the efficient compression of encrypted color images.

Author Contributions: C.W.: Conceptualization, Methodology, Writing, Reviewing, Supervision. R.L.: Investigation, Software, Writing. S.Z.: Software, Discussion, Reviewing. S.B.: Discussion, Reviewing. Z.L.: Validation, Reviewing. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported in part by the National Natural Science Foundation of China under Grants 62172165, 61672242, 61702199, and 61871201, in part by the Science and Technology Program of Guangzhou under Contract 202102020582, in part by the Science and Technology Project of Hebei Education Department under Grant QN2021417, and the Opening Project of Guangdong Province Key Laboratory of Information Security Technology under Grants 2020B1212060078-05 and 2020B1212060078-07.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is contained within the article.

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

References
1. Johnson, M.; Ishwar, P.; Prabhakaran, V.; Schonberg, D.; Ramchandran, K. On compressing encrypted data. *IEEE Trans. Signal Process.* 2004, 52, 2992–3006. [CrossRef]
2. Schonberg, D.; Draper, S.; Ramchandran, K. On Compression of Encrypted Images. In Proceedings of the 2006 International Conference on Image Processing, Atlanta, GA, USA, 8–11 October 2006; pp. 269–272. [CrossRef]
3. Schonberg, D.; Draper, S.C.; Yeo, C.; Ramchandran, K. Toward Compression of Encrypted Images and Video Sequences. *IEEE Trans. Inf. Forensics Secur.* 2008, 3, 749–762. [CrossRef]
4. Lazzeretti, R.; Barni, M. Lossless compression of encrypted grey-level and color images. In Proceedings of the 2008 16th European Signal Processing Conference, Lausanne, Switzerland, 25–29 August 2008; pp. 1–5.
5. Anil Kumar, A.; Makur, A. Distributed source coding based encryption and lossless compression of gray scale and color images. In Proceedings of the 2008 IEEE 10th Workshop on Multimedia Signal Processing, Cairns, QLD, Australia, 8–10 October 2008; pp. 760–764. [CrossRef]
6. Liu, W.; Zeng, W.; Dong, L.; Yao, Q. Efficient Compression of Encrypted Grayscale Images. *IEEE Trans. Image Process.* 2010, 19, 1097–1102. [CrossRef] [PubMed]
7. Zhou, J.; Liu, X.; Au, O.C.; Tang, Y.Y. Designing an Efficient Image Encryption-Then-Compression System via Prediction Error Clustering and Random Permutation. *IEEE Trans. Inf. Forensics Secur.* 2014, 9, 39–50. [CrossRef]
8. Wang, C.; Ni, J.; Zhang, X.; Huang, Q. Efficient Compression of Encrypted Binary Images Using the Markov Random Field. *IEEE Trans. Inf. Forensics Secur.* 2017, 13, 1271–1285. [CrossRef]
9. Wang, C.; Feng, Y.; Li, T.; Xie, H.; Kwon, G.R. A new encryption-then-compression scheme on gray images using the markov random field. *Comput. Mater. Contin.* 2018, 56, 107–121. [CrossRef]
10. Kumar, A.A.; Makur, A. Lossy compression of encrypted image by compressive sensing technique. In Proceedings of the TENCON 2009—2009 IEEE Region 10 Conference, Singapore, 23–26 January 2009; pp. 1–5. [CrossRef]
11. Zhang, X.; Ren, Y.; Feng, G.; Qian, Z. Compressing Encrypted Image Using Compressive Sensing. In Proceedings of the 2011 Seventh International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Dalian, China, 14–16 October 2011; pp. 222–225. [CrossRef]
12. Song, C.; Lin, X.; Shen, X. Secure and effective image storage for cloud based e-healthcare systems. In Proceedings of the IEEE Global Communications Conference, Atlanta, GA, USA, 9–13 December 2013; pp. 653–658.
13. Zhang, B.; Yang, L.; Wang, K.; Cao, Y. Random Permutation-Based Block Compressed Sensing for Image Encryption-then-Compression Applications. In Proceedings of the 2018 IEEE 18th International Conference on Communication Technology (ICCT), Chongqing, China, 8–11 October 2018; pp. 1203–1207. [CrossRef]
14. Gschwandtner, M.; Uhl, A.; Wild, P. Compression of Encrypted Visual Data. In Proceedings of the IFIP International Conference on Communications and Multimedia Security, Crete, Greece, 19–21 October 2006; pp. 141–150.
15. Zhang, X. Lossy Compression and Iterative Reconstruction for Encrypted Image. *IEEE Trans. Inf. Forensics Secur.* 2011, 6, 53–58. [CrossRef]
16. Zhang, X.; Feng, G.; Ren, Y.; Qian, Z. Scalable Coding of Encrypted Images. *IEEE Trans. Image Process.* 2012, 21, 3108–3114. [CrossRef] [PubMed]
17. Zhang, X.; Sun, G.; Shen, L.; Qin, C. Compression of encrypted images with multi-layer decomposition. *Multimed. Tools Appl.* **2014**, *72*, 489–502. [CrossRef]

18. Zhang, X.; Ren, Y.; Shen, L.; Qian, Z.; Feng, G. Compressing Encrypted Images With Auxiliary Information. *IEEE Trans. Multimed.* **2014**, *16*, 1327–1336. [CrossRef]

19. Hu, R.; Li, X.; Yang, B. A new lossy compression scheme for encrypted gray-scale images. In Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 4–9 May 2014; pp. 7387–7390. [CrossRef]

20. Wang, C.; Ni, J. Compressing Encrypted Images Using the Integer Lifting Wavelet. In Proceedings of the 2015 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), Adelaide, Australia, 23–25 September 2015; pp. 409–412. [CrossRef]

21. Wang, C.; Ni, J.; Huang, Q. A New Encryption-Then-Compression Algorithm Using the Rate-Distortion Optimization. *Signal Process. Image Commun.* **2015**, *39*, 141–150. [CrossRef]

22. Wang, C.; Xiao, D.; Peng, H.; Zhang, R. A lossy compression scheme for encrypted images exploiting Cauchy distribution and weighted rate distortion optimization. *J. Vis. Commun. Image Represent.* **2018**, *51*, 122–130. [CrossRef]

23. Qin, C.; Zhou, Q.; Cao, F.; Dong, J.; Zhang, X. Flexible Lossy Compression for Selective Encrypted Image With Image Inpainting. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *29*, 3341–3355. [CrossRef]

24. Kang, X.; Peng, A.; Xu, X.; Cao, X. Performing scalable lossy compression on pixel encrypted images. *Eurasip J. Image Video Process.* **2013**, *2013*, 400. [CrossRef]

25. Zhou, J.; Au, O.C.; Zhai, G.; Tang, Y.Y.; Liu, X. Scalable Compression of Stream Cipher Encrypted Images Through Context-Adaptive Sampling. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 1857–1868. [CrossRef]

26. Wang, C.; Li, T.; Ni, J.; Huang, Q. A New MRF-Based Lossy Compression for Encrypted Binary Images. *IEEE Access* **2020**, *8*, 11328–11341. [CrossRef]

27. Donoho, D. Compressed Sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [CrossRef]

28. Kschischang, F.; Frey, B.; Loeliger, H.A. Factor graphs and the sum-product algorithm. *IEEE Trans. Inf. Theory* **2001**, *47*, 498–519. [CrossRef]

29. Kang, X.; Xu, X.; Peng, A.; Zeng, W. Scalable Lossy Compression for Pixel-Value Encrypted Images. In Proceedings of the 2012 Data Compression Conference, Snowbird, UT, USA, 10–12 April 2012; p. 400. [CrossRef]

30. Al-Shaykh, O.; Mersereau, R. Lossy compression of noisy images. *IEEE Trans. Image Process.* **1998**, *7*, 1641–1652. [CrossRef] [PubMed]