symmetry

MDPI

Article

Separable Data-Hiding Scheme for Encrypted Image to Protect Privacy of User in Cloud

Li Liu [1,2,3,*], Lifang Wang [2], Yun-Qing Shi [3] and Chin-Chen Chang [4]

[1] School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, China
[2] College of Electronic Information and Engineering, Taiyuan University of Science and Technology, Taiyuan 030024, China; wanglf@nwpu.edu.cn
[3] Department of Electronics and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102, USA; shi@njit.edu
[4] Department of Information Engineering and Computer Science, Feng Chia University, No. 100, Wenhwa Rd., Seatwen, Taichung 40724, Taiwan; alan3c@gmail.com
* Correspondence: skies5315@sina.com; Tel.: +1-862-215-3228

Received: 11 December 2018; Accepted: 8 January 2019; Published: 13 January 2019

Abstract: As cloud computing becomes popular, the security of users' data is faced with a great threat, i.e., how to protect users' privacy has become a pressing research topic. The combination of data hiding and encryption can provide dual protection for private data during cloud computing. In this paper, we propose a new separable data-hiding scheme for encrypted images based on block compressive sensing. First, the original uncompressed image is compressed and encrypted by block compressive sensing (BCS) using a measurement matrix, which is known as an encryption key. Then, some additional data can be hidden into the four least significant bits of measurement using the data-hiding key during the process of encoding. With an encrypted image that contains hidden data, the receiver can extract the hidden data or decrypt/reconstruct the protected private image, according to the key he/she possesses. This scheme has important features of flexible compression and anti-data-loss. The image reconstruction and data extraction are separate processes. Experimental results have proven the expected merits of the proposed scheme. Compared with the previous work, our proposed scheme reduces the complexity of the scheme and also achieves better performance in compression, anti-data-loss, and hiding capacity.

Keywords: separable; data hiding; compressive sensing; Cloud

1. Introduction

Cloud computing is an important pillar of the transformation of the new generation of information technology (IT) and its application mode. As such, it has changed and facilitated people's lives significantly because of its low energy consumption, fast storage, good expansibility, high reliability, and resource-sharing characteristics. It can provide users with limitless computing resources and limitless storage resources [1]. Users can have access to computer resources when it is needed without large capital costs for hardware and people to deploy and operate their service. Such elasticity of resources is unprecedented in the history of IT [2,3]. Gartner, an international research and consulting firm, said that the size of the global market for public cloud services was $260.2 billion in 2017, up 18.5% from $219.6 billion in 2016 [4]. By 2020, the global cloud computing market has been projected to reach $411 billion [5]. Gartner also predicts that cloud computing will become a $300-billion business by 2021 as companies increasingly use cloud services to implement their digital business activities. Therefore, cloud computing, as an indispensable infrastructure of the Internet society, has become a market opportunity in the next few years.

However, the services provided by third-party cloud service providers are subject to additional security threats [6]. In recent years, there has been significant concern about protecting the privacy of cloud computing. In May 2014, the network used by the American e-commerce giant, Yibei Company, was attacked, and 145 million pieces of customers' information were leaked worldwide. In October 2014, the computer system of JPMorgan Chase and Company, the largest bank in the United States based on its assets, was attacked, and information of 76 million households and 7 million small businesses was leaked. In 2016, Google's Project Zero found a serious vulnerability at Cloudflare, the web performance and security company, which resulted in the leakage of the data of millions of website customers served by the Cloudflare content delivery network. In July 2017, after Baidu Cloud, a comprehensive cloud service provider, was exposed to automatic sharing/disclosure of users' photos in 2016, the Baidu network disk once again identified that there were problems with product logic, and users' photos, documents and other private information were leaked. In August 2017, server storage files at Amazon Web Services (AWS) were leaked, exposing the information of 1.8 million voters. At this point, the security of the data in cloud computing has been pushed to the cusp of the storm, and protecting the privacy of users' data has become an extremely important and urgent task. Many research [7,8] efforts also are being devoted to protecting users' private data in cloud computing.

Data hiding [9–12] is a kind of technology that embeds secret information into a carrier dataset, such as text, audios, images, and videos, in an imperceptible manner. It ensures that the human eye cannot perceive any modification of the carrier data in order to avoid attracting the attention of malicious attackers. Encryption technology [13,14] coverts ordinary signals into unintelligible data, which is an effective and popular way to protect the confidentiality of private data. Therefore, the combination of data hiding and encryption [15,16] can provide two layers of protection for private data during cloud computing.

In 2008, Puech et al. [17] proposed the concept of data hiding for encrypted images (DH-EI). Later, many schemes [18–21] were developed to improve the performance of DH-EI. According to the smoothness of image blocks, Zhang's scheme [18] embedded additional data by flipping the three least significant bits (LSBs) of the specific pixels in each block, but the additional data cannot be extracted before image decryption. Li et al.'s scheme [19] introduced random diffusion and an accurate prediction strategy to efficiently use the correlation of pixels in order to obtain greater hiding capacity. By compressing the encrypted LSBs, Zhang's scheme [20] made space for data hiding in order to separate the extraction of data from image decryption. This separable data-hiding scheme provides more flexibility in protecting the privacy of users' data. The receiver can extract the hidden data or decrypt/reconstruct the protected private image, respectively, according to the key he/she permits. Figure 1 shows the three different cases for the receiver, i.e., (1) If a receiver has only the data-hiding key, he/she can extract the hidden data without seeing the content of the image; (2) If a receiver has only the encryption key, he/she can decrypt or reconstruct the protected private image, which is similar to the original image, but cannot extract the hidden data; (3) If the receiver has both the encryption key and the data-hiding key, he/she can extract the hidden data and reconstruct the protected private image.

Most of the existing separable data-hiding schemes for encrypted images (SDH-EI) hide data in the spatial domain [20–22], which makes them sensitive to the loss of data, and the protected image and the hidden data cannot be recovered if the encrypted image that contains the hidden data is damaged. Also, the hiding capacity is quite low. However, the separable data-hiding schemes for encrypted images based on compressive sensing (SDH-EI-CS) [23–25] inherited the performance of compressive sensing (CS) and overcame the disadvantages of lost data and low capacity. In 2014, Xiao and Chen [23] proposed a novel SDH-EI-CS scheme in which the protected image was encrypted with an encryption key and some extra space was made available to hide data. Then, the enlarged stego-image was compressed to the original size by compressive sensing (CS). Compared with the previous schemes, this scheme included the separable idea in [20], and achieved good performance of anti-data-loss. However, it did not have the compression property, and the hidden data had a small loss due to compression sensing. Later, Xiao et al. [24] improved the performance of this scheme by using the block discrete cosine transform (DCT) and obtained the merits of being able to compress the ciphertext. But the quality of the directly recovered image was decreased. In 2017, Liao et al. [25] presented as SDH-EI-CS scheme based on the discrete fourier transform (DFT) to achieve a better quality of the directly decrypted images, but with the compressibility of the cipher text sacrificed.

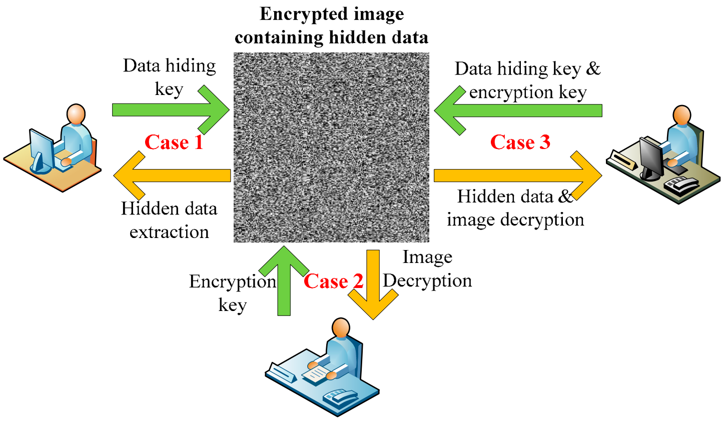

Encrypted image containing hidden data
Data hiding key
Case 1
Hidden data extraction
Data hiding key & encryption key
Case 3
Hidden data & image decryption
Encryption key
Case 2
Image Decryption

Figure 1. Three different cases for the receiver.

In this paper, we propose a new scheme of separable data hiding in encrypted images based on block compressive sensing (SDH-EI-BCS). This scheme fully includes the features of flexible compression and anti-data-loss in block compressive sensing (BCS), and its design provides a simple method of data hiding that is easy to implement as well as separate reconstruction and extraction approaches. Compared with the previous work of SDH-EI-CS, our proposed scheme reduces the complexity of the scheme and achieves better performance in compression, anti-data-loss and hiding capacity. So, the proposed scheme provides a convenient and secure approach for processing, storing and transmitting users' data in cloud computing.

The rest of this paper is organized as follows. Section 2 briefly describes three related works about block compressive sensing. In Section 3, we describe the proposed scheme. Section 4 presents the experimental results of the proposed scheme, and our conclusions are stated in Section 5.

2. Block Compressive Sensing (BCS)

In the compressive sensing (CS) theory, assume a real-valued signal $\boldsymbol{x}$ with length, N, is k-sparse in an orthonormal basis matrix $\boldsymbol{\Psi}$, sized $N \times N$, i.e., $\boldsymbol{x} = \boldsymbol{\Psi\theta}$, where the coefficient vector $\boldsymbol{\theta}$ only has k non-zero significant coefficients. The signal $\boldsymbol{x}$ can be reconstructed with certain accuracy by m measurement values:

$$\boldsymbol{y}{=}\boldsymbol{\Phi x}{=}\boldsymbol{\Phi\Psi\theta}, \tag{1}$$

where $\boldsymbol{\Phi}$ is an $m \times N$ random measurement matrix that is incoherent with $\boldsymbol{\Psi}$, and $\boldsymbol{y}$ is the measurement vector with length m, where $k < m << N$. The reconstructed process is used to solve the optimization problem [26] under the constraint of $\boldsymbol{y}{=}\boldsymbol{\Phi\Psi\theta}$:

$$\widehat{\theta} = \underset{\boldsymbol{\theta}}{\mathrm{argmin}}\|\boldsymbol{\theta}\|_{l_1} \; s.t. \; \boldsymbol{y}{=}\boldsymbol{\Phi\Psi\theta}. \tag{2}$$

Finally, $\boldsymbol{x}$ can be reconstructed by $\widehat{\boldsymbol{x}}{=}\boldsymbol{\Psi}\widehat{\boldsymbol{\theta}}$.

In order to save storage space and speed-up image processing, a block-based CS (BCS) is proposed and applied. First, an image is divided into non-overlapping $B \times B$ blocks, and then, each block $\boldsymbol{x_i}$ is individually sampled based on the same measurement matrix, $\boldsymbol{\Phi}_B$, with a constrained structure [27]. In other words, the measurement vector, $\boldsymbol{y_i}$, with the size of $m_B \times 1$ can be obtained from $\boldsymbol{y_i}{=}\boldsymbol{\Phi_B x_i}$, where $\boldsymbol{\Phi_B}$ is an $m_B \times B^2$ measurement matrix with $m_B = \lfloor m/N B^2 \rfloor$.

Compressive sensing has features of compression and encryption. In 2008, Rachlin et al. [28] discussed the security of the measurement values and had proved that compressed sensing-based encryption did not achieve Shannon's definition of perfect secrecy, but can provide a computational guarantee of secrecy. Hossein et al. [29] also proved that the measurement values can achieve perfect security under certain conditions, and it provided the theory base for the encryption application of CS.

3. Proposed Scheme

In cloud computing, a user usually hopes to encrypt the image before transmission to protect the privacy of the image. Figure 2 shows that user A encrypted the private image, saved the encrypted key into his/her private cloud storage, and then embedded the hidden bits into the encrypted image by using the data-hiding key. So, the stego-image, i.e., the encrypted image that contained the hidden data, was saved in the public cloud. When user B wants to download user A's image, he/she must provide authentication information to the public cloud. User B must pass the authentication before he/she can download the stego-image and obtain the corresponding information according to the key that he/she has.

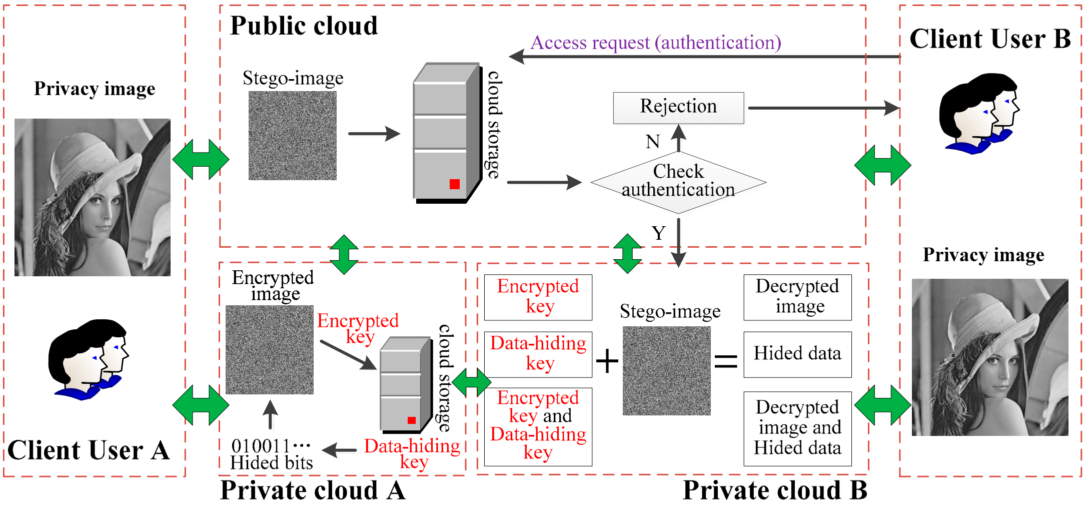


Figure 2. Example of the application of the proposed scheme in cloud computing.

Figure 3 shows the detailed process of the proposed system. First, a protected private image I is sampled by a data owner using BCS to obtain measurement values. Then, non-uniform quantization can be performed, and the data that are to be hidden are embedded into the coding of each measurement value. When the receiver possesses the encrypted image that contains hidden data, he/she can extract or decrypt the desired image in accordance with a different key.

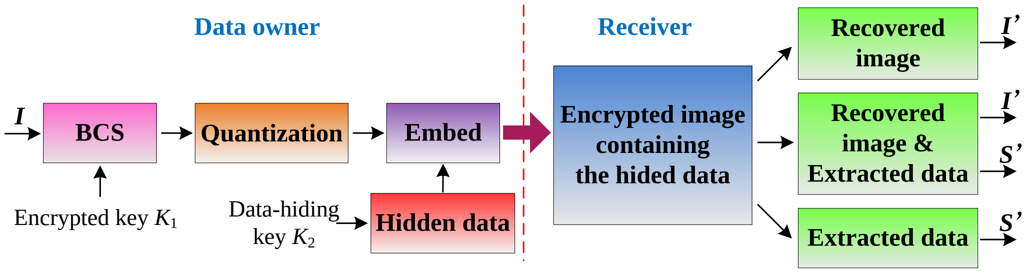


Figure 3. Detailed process of the proposed system.

3.1. Image Encryption Procedure

Assume that the protected image I, sized $M \times N$, is divided into non-overlapping blocks of size $B \times B$, and that each block $\boldsymbol{x}_i$ is measured by BCS to obtain measurement vector $\boldsymbol{y}_i$ using Equation (3), i.e., $\boldsymbol{y_i} = [y_i^1, y_i^2, \ldots, y_i^m]^{\mathrm{T}}$, where m is the number of measurements in one block. Take every $\boldsymbol{y_i}$ in order into the overall matrix $y = [y_1, y_2, \ldots, y_l]$, where $l = \frac{M \times N}{B^2}$ is the number of blocks.

$$\boldsymbol{y}_i = \boldsymbol{\Phi}_B \boldsymbol{x}_i, \tag{3}$$

where $\boldsymbol{\Phi_B}$ is a block measurement matrix with the size of $m \times B^2$. Then a random number R was used to rearrange each element of the measurement matrix in order to distribute its energy.

Equation (3) shows that the randomness of the measurement matrix destroys the correlation between the pixels of the image block, so all measurements obtained in vector $\boldsymbol{y_i}$ are completely independent of each other. Here, $\boldsymbol{\Phi_B}$ and R were used as the encrypted key K_1.

3.2. Data-Hiding Procedure

Each measurement value can be quantized into 1 of 256 levels by an 8-bit non-uniform quantization and then encoded into an eight-bit codeword adapted to the [0, 255] grayscale range for the image. Figure 4 shows an eight-bit codeword that represents the measurement, and the data hiding can be performed during the quantization and encoding procedure.

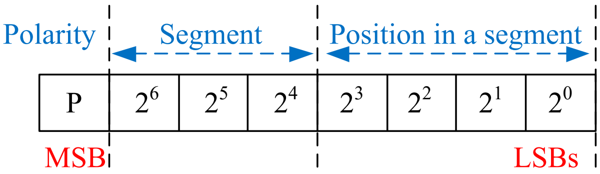


Figure 4. An eight-bit codeword that represents the measurements.

Assume $S = \{s_1, s_2, \ldots, s_r\}$, where $r = l \times m$, is a bit stream that is to be hidden. All of the measurements were normalized and converted into the sampling level q_i^j from 0 to 2048, according to Equation (4):

$$q_i^j = \frac{y_i^j}{\max(y)} \times 2048,\ i = 1, 2, \ldots, l; j = 1, 2, \ldots, m. \tag{4}$$

First, the polarity shown in Figure 4 must be obtained from Equation (5):

$$\mathrm{P} = \begin{cases} 1, & \text{if } q_i^j \text{ is positive} \\ 0, & \text{if } q_i^j \text{ is negative} \end{cases}. \tag{5}$$

Then, two steps, i.e., the segment and the position in the segment shown in Figure 4, must be finished to obtain an 8-bit codeword that has a hidden bit. Figure 5 shows the encoding and hiding processes.

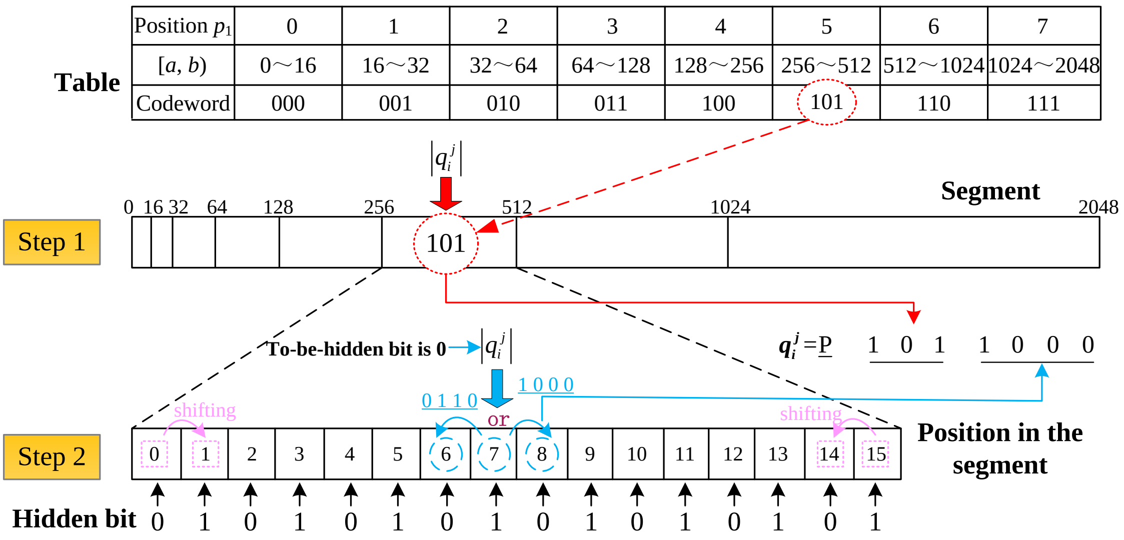
Step 1. Segment. Assume the segment where $\left|q_i^j\right|$ is located (i.e., position p_1) and determine the appropriate 3-bit code in the table in Figure 5.

$$p_1 = \begin{cases} 0, & if \left|q_i^j\right| < 16 \\ \left\lfloor \log_2 \left|q_i^j\right| \right\rfloor - 3, & if \left|q_i^j\right| \geq 16 \end{cases}, \tag{6}$$

where $\lfloor \cdot \rfloor$ indicates that the value will be rounded down to the nearest integer.

Then, recode the interval of the current segment into $[a, b]$:

$$[a, b) = \begin{cases} [2^0, 2^{p_1+4}), & if\ p_1 = 0 \\ [2^{p_1+3}, 2^{p_1+4}), & if\ p_1 \neq 0 \end{cases}. \tag{7}$$



Position p_1	0	1	2	3	4	5	6	7
$[a, b)$	0～16	16～32	32～64	64～128	128～256	256～512	512～1024	1024～2048
Codeword	000	001	010	011	100	101	110	111

Figure 5. Process of encoding and hiding.

Step 2. Position in the segment. The LSBs indicate the quantized value of the sampling level inside one of the segments. Thus, each segment is divided into a linear fashion into 16 quantum levels. The numbers from 0 to 15 are used to denote the position p_2 in a segment. So, the position p_2 of the sampling level $\left|q_i^j\right|$ in a segment can be computed using Equation (8):

$$p_2 = \left\lfloor \frac{16 \times (\left|q_i^j\right| - a)}{b - a} \right\rfloor. \tag{8}$$

Next, different rules must be used to achieve data hiding, as shown in Step 2 of Figure 5. And before hiding, the random seed is used as data-hiding key K_2 to rearrange bits in the bitstream.

Rule 1: When $p_2 = 0$ or $p_2 = 15$, if the bit to be hided $s_k \neq p_2 \text{mod} 2$, then move p_2 to the right or left by 1. Otherwise, there is nothing to do. The purpose of this operation is to ensure that the shifting of the quantized interval occurs within the segment in order to reduce the modification of the measurements.

Rule 2: When p_2 is in the range of 1 to 14, further judgement is needed if the bit that is to be hidden is $s_k \neq p_2 \text{mod} 2$. Otherwise, there is nothing to do. Through the shifting of position p_2, we can embed one bit into one measurement.

$$p_2 = \begin{cases} p_2 + 1, & \text{if } \left|q_i^j\right| >= a + \dfrac{(b-a)}{16}(p_2 + \dfrac{1}{2}) \\ p_2 - 1, & \text{if } \left|q_i^j\right| < a + \dfrac{(b-a)}{16}(p_2 + \dfrac{1}{2}) \end{cases}. \tag{9}$$

3.3. Formatting of Mathematical Components

Data extraction and image recovery are very easy in our scheme. When the receiver obtains the stego-image, if he/she has the data-hiding key K_2, the LSBs of the encrypted image can be obtained and converted into the decimal number d_t. Then the hidden bit can be computed by $s_t = d_t \text{mod} 2$. But if the receiver has an encrypted key K_1, he/she can recover the protected image directly by BCS reconstruction. Because the compressive sensing has a certain resilience to error, the small modification of the measurements will not have a significant effect on recovering the image. If the receiver has both an encrypted key K_1 and a data-hiding key K_2, he/she can obtain the hidden bits and the recovered image.

4. Experiments

Eight test images from the USC-SIPI image database, i.e., Lena, Goldhill, Airplane, Boat, Peppers, Tiffany, Man, and Sailboat, have been used as the original protected images as shown in Figure 6. They all have the size of 512 × 512. Two groups of experiments were performed to evaluate the performance of the proposed scheme, i.e., (1) performance analysis of the proposed scheme and (2) performance analysis against data-loss.

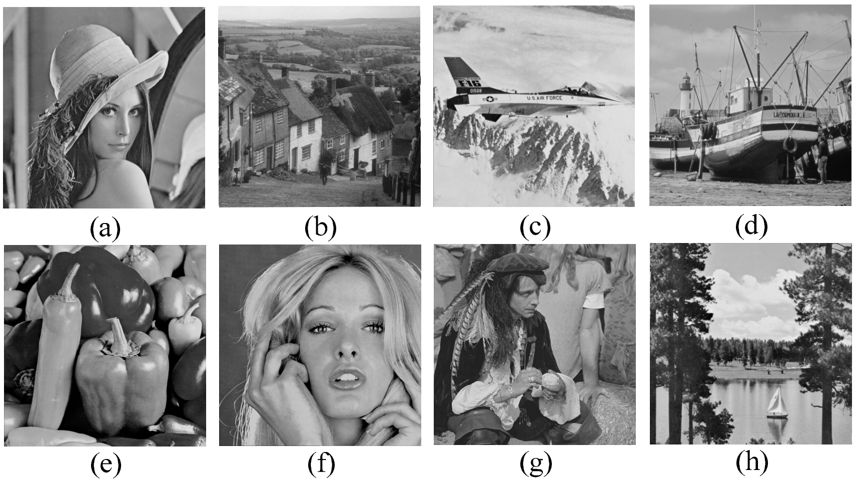


Figure 6. Eight original protected images in this experiment: (**a**) Lena, (**b**) Goldhill, (**c**) Airplane, (**d**) Boat, (**e**) Peppers, (**f**) Tiffany, (**g**) Man, and (**h**) Sailboat.

The peak signal-to-noise-ratio (PSNR), the structural similarity index measurement (SSIM), and the embedding capacity (EC) have been considered as the three primary factors to be evaluated in our proposed scheme. The PSNR defined in Equation (10) and the SSIM in reference [30] were used to evaluate the quality of the image. The larger the PSNR value is, the better the quality of the image is.

$$\begin{cases} PSNR = 10\log_{10}\dfrac{255^2}{MSE} \\ MSE = \dfrac{1}{M \times N}\sum\limits_{i=1}^{M}\sum\limits_{j=1}^{N}(D(x,y))^2 \end{cases}, \tag{10}$$

where $M \times N$ represents the size of the images, and $D(\cdot)$ is a difference matrix of the pixel values between the original image and the recovered image. SSIM simulates the human visual system (HVS) to assess the quality of images by comparing their luminance, contrast, and structure. Generally, the closer the value of SSIM is to 1, the better the visual quality of the stego-image is.

The embedding capacity in our scheme was expressed by bits and defined as follows:

$$EC = \left[\beta \times B^2\right]\frac{M \times N}{B^2}(bits), \tag{11}$$

where β is the compression ratio, B is the size of the block, and $[\cdot]$ denotes that the value will be rounded to the nearest integer. Usually, the embedding capacity is flexible and changeable depending on the compression ratio.

4.1. Performance Analysis of the Proposed Scheme

Figure 7 shows the stego-images and recovered images in this experiment when the compression ratio was 0.25 and the block size was 8. The relevant experimental data are provided in Table 1. In Figure 7, the fingerprint image with the size of 256 × 256 was treated as the image that was to be hidden by embedding it into the encrypted image, and then the stego-images were obtained as (a3), (b3), (c3), (d3), (e3), (f3), (g3), and (h3). Obviously, the image that is to be hidden and the stego-image are the same size. In cloud computing, this small size makes it easier to store and transmit the data, and this is especially important when storage and transmission resources are limited. When the receiver only has the encryption key, he/she can directly recover the original protected image from the stego-image. Moreover, the quality of the recovered image, which is evaluated by PSNR and SSIM in Table 1, is acceptable when the compression ratio is low. When the receiver only has the data-hiding key, he/she can extract the hidden data completely without seeing the content of the image as long as the stego-images are not distorted. Similarly, when the receiver has both the encryption key and the data-hiding key, he/she can extract the hidden data and recover the protected image.

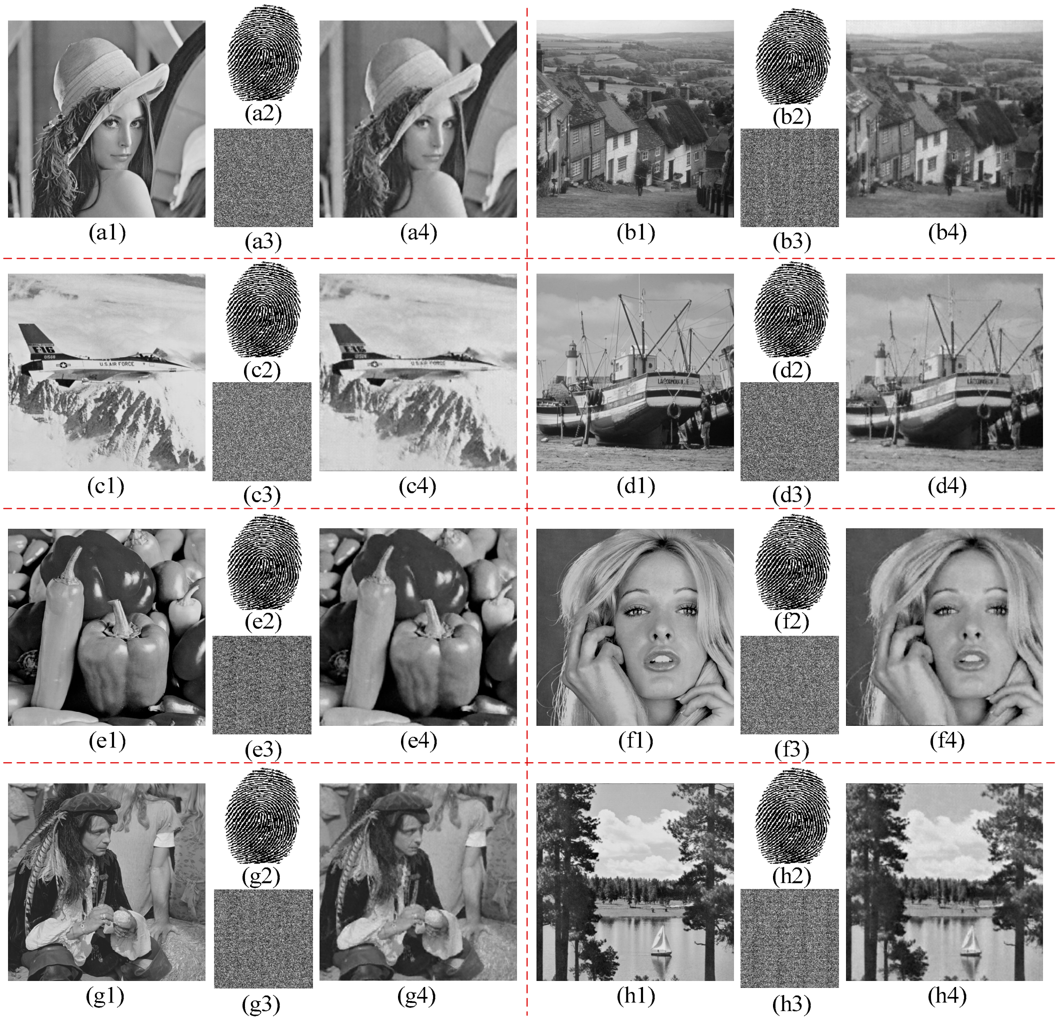


Figure 7. Demonstration of stego-images and recovered images in this experiment when the compression ratio was 0.25: (**a1**,**b1**,**c1**,**d1**,**e1**,**f1**,**g1**,**h1**) are the original protected images; (**a2**,**b2**,**c2**,**d2**,**e2**,**f2**,**g2**,**h2**) are the images that are to be hidden; (**a3**,**b3**,**c3**,**d3**,**e3**,**f3**,**g3**,**h3**) are the stego-images; (**a4**,**b4**,**c4**,**d4**,**e4**,**f4**,**g4**,**h4**) are the recovered images.

Table 1. Relevant experimental data in Figure 7.

Image	Size of Stego-Image	Peak Signal-To-Noise-Ratio (PSNR) (dB)	Structural Similarity Index Measurement (SSIM)	Embedding Capacity (EC) (Bit)
Lena	256 × 256	31.28	0.9360	65,536
Goldhill	256 × 256	29.28	0.9066	65,536
Airplane	256 × 256	29.45	0.9252	65,536
Boat	256 × 256	29.01	0.9215	65,536
Peppers	256 × 256	31.48	0.9469	65,536
Tiffany	256 × 256	28.81	0.9160	65,536
Man	256 × 256	28.80	0.9132	65,536
Sailboat	256 × 256	27.51	0.9166	65,536
Average	256 × 256	29.45	0.9228	65,536

Table 2 gives the PSNR, SSIM, and EC with different compression ratios for Lena when the block size was 8. Table 2 indicates that the change of the compression ratio gave our scheme great flexibility. The larger the compression ratio is, the better the quality of the recovered images will be and the higher the embedding capacity will be, but the increase in the size of image also will consume more storage space in cloud storage. Therefore, users can make their own decisions concerning processing based on their own needs.

Table 2. PSNR (dB), SSIM and EC (bits) with different compression ratios for Lena when the size of block was 8.

Compression Ratio β	Size of Stego-Image	PSNR	SSIM	EC
0.90	512 × 464	36.56	0.9757	237,568
0.80	512 × 408	35.94	0.9738	208,896
0.60	512 × 304	34.74	0.9655	155,648
0.50	521 × 256	33.79	0.9590	131,072
0.25	256 × 256	31.28	0.9360	65,536

Figure 8 shows an example of stego-image and the recovered image for Lena when the compression ratio was 0.9. Because of the larger compression ratio, more bits can be embedded into the encrypted images. So, size of the image that was to be hidden was chosen as 512 × 464, as shown in Figure 8b.

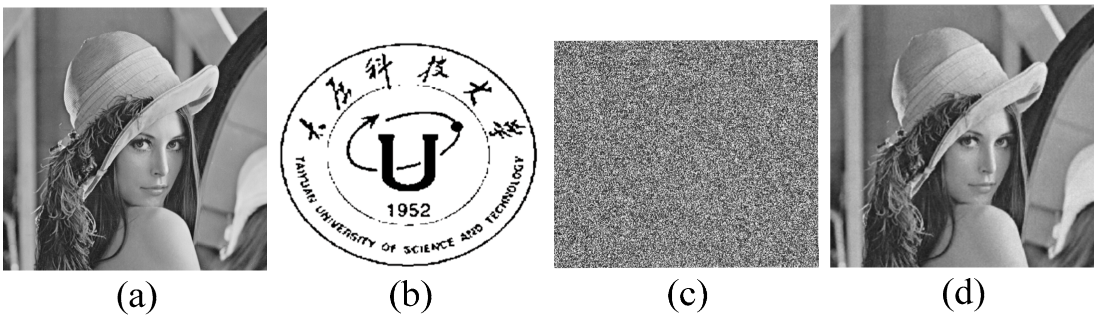


Figure 8. Example of stego-images and recovered images in this experiment, when the compression ratio was 0.9: (**a**) Original protected images; (**b**) the 512 × 464 image that was to be hidden; (**c**) Stego-image with the size of 512 × 464; (**d**) recovered images with PSNR = 36.56 dB and SSIM = 0.9757.

In addition, we also conducted additional experiments with different block sizes and compression ratios. Table 3 gives the PSNR, SSIM, and EC with different parameters for Lena. Table 3 indicates that the block size had little effect on either the embedding capacity or the quality of the image, but compression ratio affected their performance significantly.

Table 3. PSNR (dB) and EC (bits) with different compression ratios and block sizes for Lena.

Block Size $\boldsymbol{B \times B}$		**Compression Ratio** $\boldsymbol{\beta}$				
		0.9	**0.8**	**0.6**	**0.5**	**0.25**
8 × 8	PSNR	36.56	35.94	34.74	33.79	31.25
	SSIM	0.9757	0.9738	0.9655	0.9590	0.9353
	EC	237,568	208,896	155,648	131,072	65,536
16 × 16	PSNR	35.81	35.47	34.56	33.78	30.93
	SSIM	0.9712	0.9697	0.9632	0.9580	0.9267
	EC	235,520	209,920	157,696	131,072	65,536
32 × 32	PSNR	35.75	35.19	34.42	33.79	31.35
	SSIM	0.9708	0.9653	0.9600	0.9546	0.9272
	EC	236,032	209,664	157,184	131,072	65,536

To further illustrate the flexibility and effectiveness of our proposed scheme, some experimental comparisons of relevant schemes, including Xiao et al.'s scheme [24] and Liao et al.'s scheme [25], are given in Table 4. At the same compression ratio, the embedding capacity of our scheme absolutely was superior to those of the other schemes. Although the quality of the recovered image before extracting the hidden data in our scheme was slightly lower than that of Liao et al.'s scheme [25], it obviously was better than that of Xiao et al.'s scheme [24]. There was no difference in our scheme before or after the hidden data were extracted. So, the quality of the recovered image after extracting the hidden data in Xiao et al.'s scheme [24] and Liao et al.'s scheme [25] was better than that of our proposed scheme. In addition, in our scheme, the hidden data can be extracted completely as long as the stego-images are not distorted, but the other schemes cannot do this without small distortions. In terms of the complexity of the extraction, our proposed scheme only requires a simple modular operator, and this is very easy to provide.

Table 4. Comparisons of other relevant schemes and our proposed scheme.

	Xiao et al.'s Scheme [24]	**Liao et al.'s Scheme [25]**	**Our Proposed Scheme**
Compression ratio β	0.8	0.8	0.8
The embedding capacity	65,536 bits	131,072 bits	208,896 bits
Size of stego-image	512 × 408	512 × 512	512 × 408
Quality of the recovered image before extracting the hidden data	26.9 dB	37.98 dB	36.04 dB
Quality of the recovered image after extracting the hidden data	40.8 dB	45.75 dB	-
If the compression causes distortion of the hidden data	Yes	Yes	No
The complexity of extraction process	Hard	Hard	Easy

4.2. Performance Analysis against Data-Loss

In the experiment in Section 4.1, we assumed that there was no distortion of the stego-image during transmission and storage, but that does not represent reality. Let us assume that the simulation experiments were performed in an ordinary communication environment with binary pulse amplitude modulation (BPAM) baseband signals and controllable additive white Gaussian noise (AWGN) [31]. The transmission characteristic of this system [31] with the bit-error rate (BER) versus the signal-to-noise ratio (SNR) is shown in Figure 9, where E_b is energy per bit and N_0 is the spectral density of the noise.

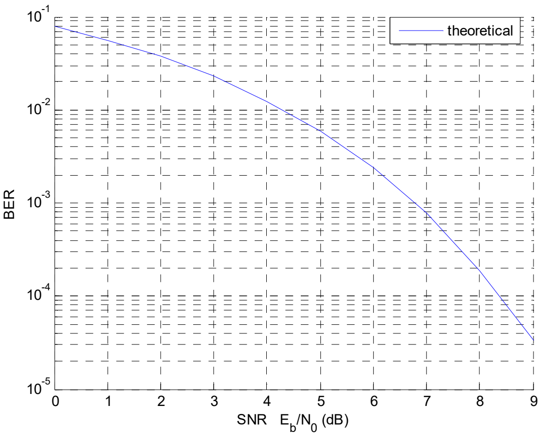


Figure 9. Graph of the bit-error rate (BER) versus the signal-to-noise ratio (SNR) in a selected communication system.

The stego-images in our experiment were transmitted in an ordinary communication environment, and controlled AWGN was added in the channel. We used the stego-images that were received to recover the protected images and extract the hidden images that corresponded to each BER, and the results are shown in Figure 10. Here, the compression ratio was 0.25 and the size of the block was 8.

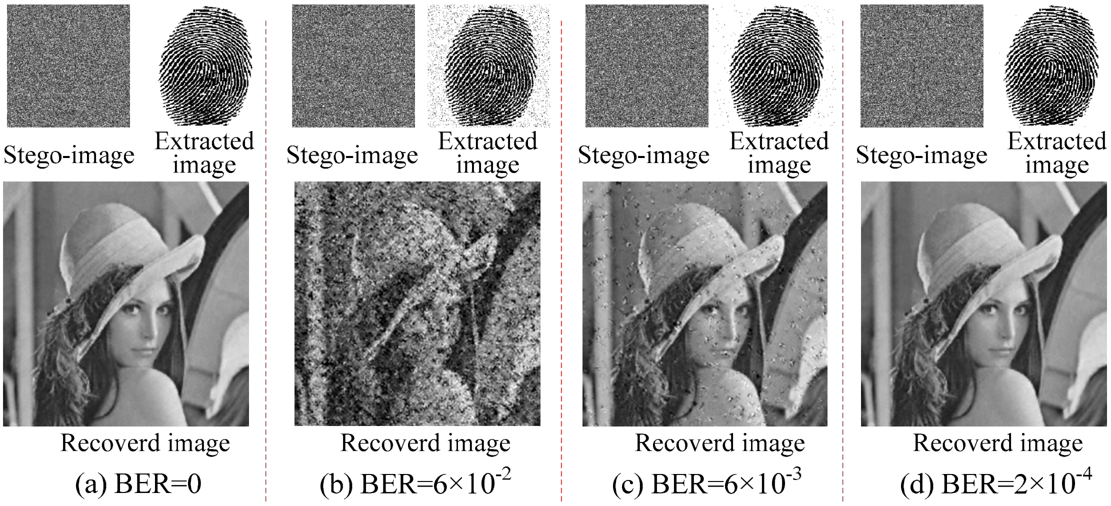


Figure 10. Extracted and recovered images of our proposed scheme with the different BER: (**a**) BER = 0, PSNR = 31.28 dB; (**b**) BER = 6 × 10^{-2}, PSNR = 12.26 dB; (**c**) BER = 6 × 10^{-3}, PSNR=22.41 dB; (**d**) BER = 2 × 10^{-4}, PSNR = 31.11 dB.

Figure 10 clearly shows that a poor transmission environment has more influence on the quality of the recovered images than it has on the quality of the extracted images. In Figure 10b, when BER was 6 × 10^{-2}, the fingerprint pattern of the extracted image clearly was discernible despite the presence of noise around them. But the PSNR of the recovered image was only 12.26 dB, which is hardly acceptable to the human eye. However, when the BER was lower, both the extracted image and the recovered image had good visual qualities.

We also conducted a group of cropping tests, and the results are shown in Figure 11. To illustrate the cropping ratio more visually, cropped parts of the images are shown with black pixels. Obviously, when the stego-image was cut in half, i.e., the cropping ratio was 50% in Figure 11a, the visual effect of the extracted image was diminished by noise, but the fingerprint pattern was still discernible, and the quality of the recovered image was satisfactory. As the cropping ratio decreased, the quality of the extracted and recovered images increased.

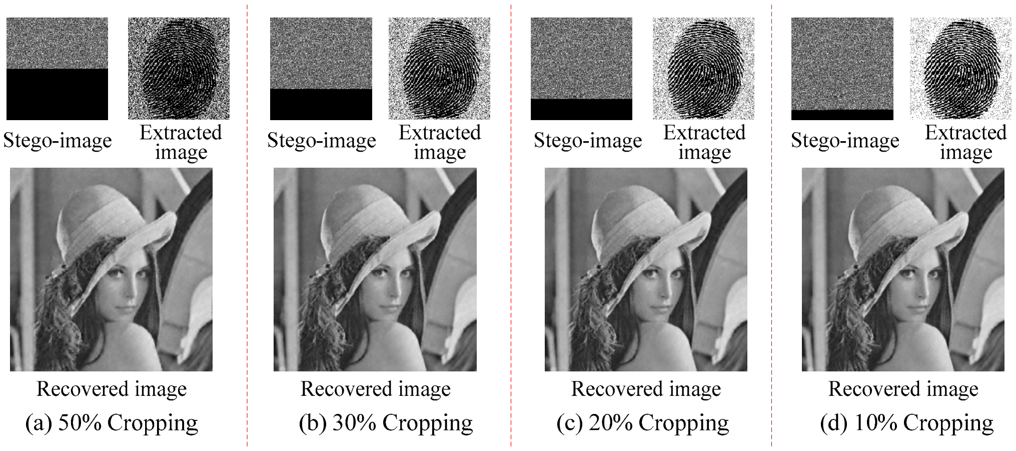


Figure 11. Cropping test in our proposed scheme with different cropping ratios: (**a**) 50% cropping, PSNR = 28.65 dB; (**b**) 30% cropping, PSNR = 30.04 dB; (**c**) 20% cropping, PSNR = 30.59 dB; (**d**) 10% cropping, PSNR = 31.10 dB.

Obviously, our proposed scheme is more effective in anti-cropping characteristics than in anti-error characteristics. This is because the bit error of the most significant bit (MSB) in the quantization process can lead to large distortion of the measurement value, which seriously affects the reconstruction of the image. Absolutely, the larger the compression ratio, the better the quality of image, and the larger the size of the stego-image.

4.3. Complexity and Security Performance Analysis

In the processes of encryption, we analyze the complexity of encryption part including compressive sensing and scrambling. Assuming that the image size is $M \times N$ and the block size is $B \times B$, and the size of the two projection multiplication matrices for each block are $\beta B \times B$ and $B \times B$, respectively, where $\beta \leq 1$ is the compression rate, the projection complexity is $O(\beta B \times B \times B \times l)$, where l is the number of blocks in the image. The complexities of scrambling and data hiding are related to the size of the stego-image. So their complexity are both $O(\beta M \times N)$. Therefore, the complexity of the processed of encryption and data embedding is $T(n) = O(\beta M \times N) + O(\beta M \times N) + O(\beta B \times B \times B \times l)$. In the processes of data extraction and image recovery, the main time consumption the CS recovery. It is a convex optimization problem that can be solved with different kinds of methods by the receiver with abundant resource.

The security of our scheme mainly is apparent in compressive sensing and scrambling. In BCS reconstruction, matrix $\boldsymbol{\Phi}_B$, which is the same as the encryption, was used in the decryption. Moreover, we also used the random number to arrange the measurement values. If an unauthorized user cannot accurately obtain this matrix and the random number, then the original protected image cannot be reconstructed. However, the data volume is huge. If a user does not know in advance, it is impossible to guess the right matrix and measurement values.

5. Conclusions

In this paper, we proposed a new scheme of separable data-hiding schemes for encrypted images based on block compressive sensing (SDH-EI-BCS). First, the protected private image is compressed and encrypted using a measurement matrix, which is known as an encryption key. Then, some additional data can be hidden into the four least significant bits of measurement using the data-hiding key during the process of encoding. With an encrypted image that contains hidden data, the receiver can extract the hidden data or decrypt/reconstruct the protected private image based on the key that he/she possesses. This scheme has features of flexible compression and anti-data-loss. Moreover, the reconstruction of the image and the extraction of data are separate. Also, compared with the previous work of SDH-EI-CS, our proposed scheme reduces the complexity of the scheme and achieves better performance in compression, anti-data-loss, and hiding capacity. So, the proposed scheme provides convenience and security for the processing, storage, and the transmission of users' data in cloud computing. However, due to the distortion of the images, our proposed scheme cannot be used in medical, military, or other areas that require the reconstruction of images without distortion. In future work, we will focus on the research of separable reversible data-hiding scheme for encrypted image in order to provide a security guarantee for user data in every fields.

Author Contributions: Conceptualization, L.L. and Y.-Q.S.; Methodology, C.-C.C.; Software, L.L.; Supervision, L.W.; Validation, L.W., Y.-Q.S. and C.-C.C.; Writing—original draft, L.L.; Writing—review and editing, C.-C.C.

Funding: This research received no external funding.

Acknowledgments: This work was supported in part by The National Natural Science Foundation of China (No. 61540009), Shanxi Province Natural Science Foundation (No. 201801D121129). The first author was supported by a scholarship from China Scholarship Council (CSC).

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

References

1. Shen, J.; Zhou, T.; Chen, X.; Li, J.; Susilo, W. Anonymous and Traceable Group Data Sharing in Cloud Computing. *IEEE Trans. Inf. Foren. Sec.* **2018**, *13*, 912–925. [CrossRef]
2. Armbrust, M.; Fox, A.; Griffith, R.; Joseph, A.D.; Katz, R.; Konwinski, A. A view of cloud computing. *Int. J. Comput. Technol.* **2013**, *4*, 50–58. [CrossRef]
3. Shiau, W.L.; Chau, P.Y.K. Understanding behavioral intention to use a cloud computing classroom. *Inf. Manag.* **2016**, *53*, 355–365. [CrossRef]
4. Sony, S. The interconnected multi-cloud: The future of digital security. *Comput. Fraud Secur.* **2017**, *5*, 19–20. [CrossRef]
5. Available online: https://www.gartner.com/smarterwithgartner/7-hidden-cloud-growth-opportunities-for-technology-service-providers/ (accessed on 20 June 2018).
6. Ali, M.; Khan, S.U.; Vasilakos, A.V. Security in cloud computing: Opportunities and challenges. *Inf. Sci.* **2015**, *305*, 357–383. [CrossRef]
7. Xia, Z.; Zhu, Y.; Sun, X.; Qin, Z.; Ren, K. Towards privacy-preserving content-based image retrieval in cloud computing. *IEEE Trans. Cloud Comput.* **2018**, *6*, 276–286. [CrossRef]
8. Wu, X.; Tang, S.; Yang, P.; Xiang, C.; Zheng, X. Cloud is safe when compressive: Efficient image privacy protection via shuffling enabled compressive sensing. *Comput. Commun.* **2018**, *117*, 36–45. [CrossRef]
9. Liu, L.; Chang, C.C.; Wang, A. Data hiding based on extended turtle shell matrix construction method. *Multimed. Tools Appl.* **2016**, *76*, 1–18. [CrossRef]
10. Lin, Y.K. High capacity reversible data hiding scheme based upon discrete cosine transformation. *J. Syst. Softw.* **2012**, *85*, 2395–2404. [CrossRef]
11. Shen, S.Y.; Huang, L.H. A data hiding scheme using pixel value differencing and improving exploiting modification directions. *Comput. Secur.* **2015**, *48*, 131–141. [CrossRef]
12. Liu, L.; Chang, C.C.; Wang, A. Reversible data hiding scheme based on histogram shifting of n-bit planes. *Multimed. Tools Appl.* **2016**, *75*, 11311–11326. [CrossRef]
13. Wu, Y.; Zhou, Y.; Agaian, S. Image encryption using the Sudoku matrix. *Proc. SPIE Int. Soc. Opt. Eng.* **2010**, *7708*, 247.
14. Assad, S.E.; Farajallah, M. A new chaos-based image encryption system. *Signal Process. Image Commun.* **2016**, *41*, 144–157. [CrossRef]
15. Narayan, B.; Bhaskar, S.; Kailash, M.; Mahyavanshi, N. A secure method for data hiding in encrypted image using progressive recovery. *Int. J. Comput. Appl.* **2017**, *161*, 1–5. [CrossRef]
16. Qian, Z.; Zhang, X. Reversible data hiding in encrypted images with distributed source encoding. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *26*, 636–646. [CrossRef]
17. Puech, W.; Chaumont, M. A reversible data hiding method for encrypted images. *J. Electron. Imaging* **2008**, *6819*, 68191E.
18. Zhang, X. Reversible data hiding in encrypted image. *IEEE Signal Proc. Lett.* **2011**, *18*, 255–258. [CrossRef]

19. Li, M.; Xiao, D.; Peng, Z.; Nan, H. A modified reversible data hiding in encrypted images using random diffusion and accurate prediction. *Etri J.* **2014**, *36*, 325–328. [CrossRef]
20. Zhang, X. Separable reversible data hiding in encrypted image. *IEEE Trans. Inf. Forensics Secur.* **2012**, *7*, 826–832. [CrossRef]
21. Kim, C.; Shin, D.; Leng, L.; Yang, C.N. Separable reversible data hiding in encrypted halftone image. *Displays* **2018**, *55*, 71–79. [CrossRef]
22. Xu, D.; Chen, K.; Wang, R.; Su, S. Separable reversible data hiding in encrypted images based on two-dimensional histogram modification. *Sec. Commun. Netw.* **2018**, *2018*, 1734961. [CrossRef]
23. Xiao, D.; Chen, S. Separable data hiding in encrypted image based on compressive sensing. *Electron. Lett.* **2014**, *50*, 598–600. [CrossRef]
24. Xiao, D.; Cai, H.; Wang, Y.; Bai, S. High-capacity separable data hiding in encrypted image based on compressive sensing. *Multimed. Tools Appl.* **2016**, *75*, 13779–13789. [CrossRef]
25. Liao, X.; Li, K.; Yin, J. Separable data hiding in encrypted image based on compressive sensing and discrete Fourier transform. *Multimed. Tools Appl.* **2017**, *76*, 20739–20753. [CrossRef]
26. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [CrossRef]
27. Liu, L.; Wang, A.; Chang, C.C.; Li, Z. A novel real-time and progressive secret image sharing with flexible shadows based on compressive sensing. *Signal Process. Image Commun.* **2014**, *29*, 128–134. [CrossRef]
28. Rachlin, Y.; Baron, D. The secrecy of compressed sensing measurements. In Proceedings of the Conference on Communication, Control, & Computing, Urbana-Champaign, IL, USA, 23–26 September 2008; pp. 813–817.
29. Hossein, S.A.; Tabatabaei, A.E.; Zivic, N. Security analysis of the joint encryption and compressed sensing. In Proceedings of the 20th Telecommunications Forum, Belgrade, Serbia, 20–22 November 2012; pp. 799–802.
30. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–614. [CrossRef] [PubMed]
31. Proakis, J.G.; Salehi, M. *Communication Systems Engineering*, 2nd ed.; Prentice Hall: Englewood Cliffs, NJ, USA, 2002.