Privacy-preserved data hiding using compressive sensing and fuzzy C-means clustering

Ming Li¹,², Lanlan Wang¹,² and Haiju Fan¹,²

Abstract
Nowadays, digital images are confronted with notable privacy and security issues, and many research works have been accomplished to countermeasure these risks. In this article, a novel scheme for data hiding in encrypted domain is proposed using fuzzy C-means clustering and compressive sensing technologies to protect privacy of the host image. The original image is preprocessed first to generate multiple highly correlated classes with fuzzy C-means clustering algorithm. Then, all classes are further divided into two parts according to proper threshold. One is encrypted by stream cipher, and the other is encrypted and compressed simultaneously with compressive sensing technology for easy data embedding by information hider. The receiver can extract additional data and recover the original image with data-hiding key and encryption key. Experiments and analysis demonstrate that the proposed scheme can achieve higher embedding rate about additional data and better visual quality of recovered image than other state-of-the-art schemes.

Keywords
Encryption, data hiding, fuzzy C-means, compressive sensing, privacy

Date received: 23 September 2019; accepted: 4 February 2020
Handling Editor: Hongxiang Li

Introduction
With the continuous development of computer network technology, more and more data, especially the multimedia data (images, audios, and videos) captured by the sensors, are uploaded to cloud platforms by users to alleviate storing and computing burdens.¹ However, some unexpected results may happen under this circumstance, such as some illegal individuals or groups infringe copyright, disclose privacy, steal commercial confidential information, and so on. Therefore, how to manage users’ data in the complex network environment with a privacy-preserved way has become an urgent problem for the enterprises.²

Digital image is one of the most typical multimedia. An important feature of digital images is data redundancy, which means that strong correlations always exist in the spatial domain of natural images. This is also the reason why images can be compressed effectively. In addition, because the information entropy of the image is generally far from the maximum value, data hiding in the host image is possible. Data hiding has been considered as an effective way to manage the host media in terms of data authentication, protection, classification, retrieval, and so on.³⁻⁵

¹ College of Computer and Information Engineering, Henan Normal University, Xinxiang, China
² Big Data Engineering Laboratory for Teaching Resources & Assessment of Education Quality, Xinxiang, China

Corresponding authors:
Ming Li, College of Computer and Information Engineering, Henan Normal University, Xinxiang 453007, China.
Email: liming@htu.edu.cn
Haiju Fan, College of Computer and Information Engineering, Henan Normal University, Xinxiang 453007, China.
Email: 121064@htu.edu.cn
Generally, there are two criteria to evaluate the performance of data hiding. One is data embedding capacity, which can be measured by bit per pixel (bpp). The other is the quality of stego-image, which can be measured by peak signal-to-noise ratio (PSNR). With the aim of preserving privacy, a growing number of original data is encrypted before uploading it onto the cloud. Therefore, data hider needs to embed additional information into encrypted images directly without knowing any information about the original data, which is called data hiding in encrypted image.

Compared with traditional data hiding in plaintext image, few redundancy spaces can be used to embed additional data, since the information entropy of the host media is close to the maximum value after encryption. In other words, encryption brings new challenge to data hiding. To perform data hiding in encrypted images effectively, additional space should be vacated in the host image. According to the sequence of creating additional space and encryption, existing embedding mechanisms in encrypted image can be classified into two categories: (1) The result of classification is determined by class number and threshold, and the size of vacated capacity is controlled by CS sampling rate; (2) The other category where the absolute differences of each class are greater than the threshold is encrypted via exclusive-or operation. The proposed scheme aims to encrypt image and reserve room simultaneously for effective data hiding.

Image encryption techniques have been greatly advanced recently. One of the most attractive signal-processing techniques today is compressive sensing (CS), which is used for efficient signal acquiring and reconstructing. The principle of CS is that a sparse signal can be recovered by optimization from far fewer samples than the required according to the Nyquist–Shannon sampling theorem. CS can also be designed with cryptosystem property to achieve signal compression and encryption simultaneously, and it is especially suitable for the resource restricted wireless sensors network (WSN). However, data hiding in CS domain is difficult because the information entropy of the signal obtained by CS has increased greatly. Many solutions combining CS with data hiding have been proposed. Some of them are based on the segmentation or pre-treatment of the host signal, while others processed the measurements directly. As stated before, reversible data hiding can only be achieved under the condition that the information entropy of host media has not reached the maximum value. In other words, additional space must be vacated in the encrypted domain before data hiding, and it is better to achieve CS-based encryption and vacating room simultaneously.

Motivated by the requirements of the privacy and fidelity of the host image and high embedding capacity of the additional data, this work proposes a novel privacy-preserved data-hiding scheme. The original image is pre-processed and divided into multiple classes through fuzzy C-means (FCM) clustering algorithm by image owner, and then all classes are classified into two major categories according to the given threshold. The category where the absolute differences of each class are greater than the threshold is encrypted via exclusive-or operation. The other category is encrypted and compressed synchronously by CS technology, where the measurement matrix is served as encryption key. The whole encrypted and compressed image generated will be transmitted to data hider to embed additional information with data-hiding key. After obtaining the marked, encrypted image, the receiver can extract additional information exactly and recover original image with satisfactory visual quality. The proposed scheme can achieve secure transmission of secret data, protection and management of host images by applying data hiding in the CS domain.

The contributions of our scheme are summarized as follows: (1) The original image is divided into multiple classes, but only partial classes whose the absolute difference less than the threshold are compressed to vacate room for additional data. Thus, the visual quality of recovered image can be well guaranteed. (2) The combination of FCM clustering algorithm and CS technology makes full use of the sparsity of original image to achieve higher embedding capacity. (3) By combining CS with cryptosystem, data privacy can be protected effectively while making room for embedding additional information. (4) The result of classification is determined by class number and threshold, and the size of vacated capacity is controlled by CS sampling rate; therefore, the visual quality of the recovered image and the embedding rate of the host image can be adjusted according to specific requirements.

The remaining article is arranged as following. Relevant knowledge is shown in section “Relevant knowledge.” The details of our scheme are introduced elaborately in section “Proposed scheme.” Experimental results and performance comparisons with other related works are given in section “Experimental results and performance analysis.” Finally, section “Conclusion” concludes our work.

**Relevant knowledge**

**Tent-logistic chaotic system**

Chaos system is extremely sensitive to initial state and regulatory parameters, so any slight change of initial state can be exponentially amplified. Tent-logistic chaotic system (TLCS) is a combination of two
The central problem after reconstructing the signal vector in space \( f \) through the nonlinear reconstruction algorithm accurately or inversely, the original signal can be recovered through the sampling rate is much lower than Nyquist.

In certain transform domain under the condition that the randomness by applying the sparse character of nature signal, the chaos range of TLCS is much larger than single logistic or tent system, and it can realize more unpredictability chaotic performance.

**CS technology**

CS is a novel signal sampling technology described in detail by Donoho. The signal can be sampled randomly by applying the sparse character of nature signal in certain transform domain under the condition that the sampling rate is much lower than Nyquist. Inversely, the original signal can be recovered through the nonlinear reconstruction algorithm accurately or with high probability.

Assume that the original signal \( x \) is the \( N \)-dimension vector, \( \{\psi_i\}_{i=1}^N \) represents \( N \)-dimension orthogonal vector in space \( \mathbb{R}^N \), and \( \Psi = [\psi_1, \psi_2, \ldots, \psi_N] \) signifies an orthogonal basis. Thus, the signal \( x \) can be represented as

\[
x = \Psi \theta \quad \text{or} \quad x = \sum_{i=1}^N \theta_i \psi_i
\]

where \( \theta \) is the projection coefficient vector of signal \( x \) on orthogonal basis \( \Psi \). The central problem after sparse representation is to find a proper random measurement matrix \( \Phi \) to meet the restricted isometry property (RIP), and the non-adaptive linear sampling process can be expressed as

\[
y = \Phi x = \Phi \Psi \theta
\]

The process can also be shown in Figure 1.

Due to the number of measurements \( M \ll N \), the random measurement process of a signal is regarded as compression process. We can get the approximate value \( \hat{x} \) of the original signal by solving the convex optimization problem by equations (6) and (7)

\[
\min \|\theta\|_1 \text{ s.t. } y = \Phi \Psi \theta
\]

\[
\hat{x} = \Psi \hat{\theta}
\]

where \( \hat{\theta} \) is the solution of the convex optimization problem and \( \hat{x} \) means the recovered signal.

**FCM clustering algorithm**

FCM is a fuzzy clustering algorithm based on objective function for data clustering analysis, the core idea of which is the objects divided into the same cluster having the largest similarity but different clusters possessing the smallest similarity. In addition, FCM as an improvement of the ordinary C-means algorithm is a flexible fuzzy division. The explicit process about implementation of FCM is described as follows.

Suppose the sample set \( X = \{x_1, x_2, \ldots, x_n\} \), where \( c \) and \( n \) represent the number of clusters and the number of samples in set, respectively. Here, \( n \) samples would be divided into \( c \) classes, and the membership matrix of \( n \) samples corresponding to \( c \) classes is expressed as

\[
U_{c, n} = \begin{bmatrix}
u_{11} & u_{12} & \cdots & u_{1n} \\
u_{21} & u_{22} & \cdots & u_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
u_{c1} & u_{c2} & \cdots & u_{cn}
\end{bmatrix}
\]
which also can be written as $U_{c, n} = \{u_{i, j} | i \in [1, c], j \in [1, n]\}$, where each item $u_{i, j}$ of $U_{c, n}$ is the probability of $i$th sample $x_j$ belonging to the $i$th cluster. Meanwhile, each element $u_{i, j}$ should conform to two restricted conditions represented by equations (9) and (10)

$$u_{i, j} \in [0, 1], \text{ where } i = 1, 2, \ldots, c; \quad j = 1, 2, \ldots, n \quad (9)$$

$$\sum_{i=1}^{c} u_{i, j} = 1, \text{ where } i = 1, 2, \ldots, c; \quad j = 1, 2, \ldots, n \quad (10)$$

Equation (9) means the value of $u_{i, j}$ changing between interval $[0, 1]$, and equation (10) denotes that each $x_j$ of $X$ should belong to a definite class. That is, $i$th cluster $(i \in [1, c])$ is with certain probability $u_{i, j}$ and the sum is equal to 1. The generalized form of objective function about FCM clustering algorithm is shown in equation (11)

$$J_{FCM}(U, V, X) = \sum_{i=1}^{c} \sum_{j=1}^{n} (u_{i, j})^m (d_{i, j})^2 \quad (11)$$

where parameter $V = \{V_1, V_2, \ldots, V_c\}$ denotes the cluster centers of each cluster, and the exponent $m$ refers to predefined weighting divisor and impacts the clustering effect indirectly. In reference Bezdek et al., Pal and Bezdek consider that the effect is better when the value of $m$ is in interval $[1.5, 2.5]$. Thus, the complex problem of fuzzy clustering is transformed to the single issue of minimization of equation (12). $d_{i, j}$ signifies the Euclidean distance between $i$th cluster center $V_i$ and $j$th sample $x_j$, which can be obtained by equation (12)

$$d_{i, j} = |x_j - V_i| = \sqrt{(x_j - V_i)^T (x_j - V_i)} \quad (12)$$

The implementation of the whole FCM clustering process is illustrated in Figure 2, which is also described step by step as follows.

**Step 1.** Set cluster number $c$ ($2 \leq c \leq n$), weighting exponent $m$, threshold $\varepsilon$ used for terminating iteration, random initialization $U^{(0)}$, and iteration counter $b = 0$.

**Step 2.** Update the cluster center matrix $V^{(b)}$ by equation (13)

$$V_i^{(b)} = \frac{\sum_{j=1}^{n} (u_{i, j}^{(b)})^m \cdot x_j}{\sum_{j=1}^{n} (u_{i, j}^{(b)})^m}, \quad i = 1, 2, \ldots, c \quad (13)$$

**Step 3.** Update the fuzzy membership matrix $U^{(b + 1)}$ according to equation (14)

$$U_i^{(b + 1)} = \sum_{k=1}^{c} \left( \frac{d_{i, k}^2}{d_{i, j}^2} \right)^{\frac{1}{m-1}} \quad (14)$$

**Step 4.** If $\sum_{k=1}^{c} \left( \frac{d_{i, k}^2}{d_{i, j}^2} \right)^{\frac{1}{m-1}} < \varepsilon$, output the clustering information including clustering center matrix $V$ and membership matrix $U$, and stop the process. Otherwise, set $b = b + 1$ and go to Step 2 to continue next iteration. Note that $\| \cdot \|$ means appropriate distance norm.

**Proposed scheme**

The proposed scheme mainly consists of image encryption, data hiding, data extraction, and image recovery. The overall framework of our scheme is illustrated in Figure 3. In the phase of image encryption, the original image is encrypted and compressed synchronously via traditional stream cipher and CS technology. As for data hiding, after receiving the whole processed image, data hider can embed additional information directly with data-hiding key to facilitate management and operation of the host data. In the end, the receiver can extract additional information exactly and recovery.
original image with satisfactory visual quality from the marked, encrypted image.

**Image encryption**

During this stage, the original image $I$ with size of $M \times N$ is split into $c$ classes expressed as $C = \{C_1, C_2, \ldots, C_i, \ldots, C_c\}$ by FCM clustering algorithm described in section “Relevant knowledge.” Then, according to the given threshold $T$, all classes are divided into two categories $C_1$ and $C_2$ by equations (15) and (16).

$$D_i = \text{max}(C_i) - \text{min}(C_i), 1 \leq i \leq c$$  
(15)

$$\begin{cases} 
    C_i \in C_1, & \text{if } D_i > T \\
    C_i \in C_2, & \text{if } D_i \leq T 
\end{cases}$$  
(16)

The classification results of Lena with $c = 13$, $T = 10$ and $c = 13$, $T = 15$ are shown in Figure 4, in which the white and black regions represent $C_1$ and $C_2$, respectively. Each class in $C_1$ is encrypted by traditional stream encryption, and the gray value of each pixel ranging from 0 to 255 in $C_1$ can be expressed as eight binary bits using equation (17).

$$c_1(x, y, k) = \text{mod}\left(\left\lfloor \frac{c_1(x, y)}{2^k} \right\rfloor, 2\right),$$  
(17)

$$0 \leq x < M, \ 0 \leq y < N, \ k = 0, 1, \ldots, 7, \ i = 0, 1, \ldots, c$$

In the process of image encryption, all pixels in the same class belonging to categories $C_1$ are encrypted with the same way through equation (18) by encryption key $k_i$ generated via TLCS

$$c_1'(x, y, k) = c_1(x, y, k) \oplus k_i(k)$$  
(18)

where $k_i(k)$ is the random bit corresponding to encryption key $k_i$ to encrypt the $k$th bit plane of the class $C_1$, and $c_1'(x, y, k)$ represents the encrypted result by key $k_i(k)$ and $c_1(x, y, k)$ with exclusive-or operation. Then, collect all the encrypted binary bits $c_1'(x, y, k)$ to generate the cipher class $C_1'$ as shown in equation (19), and the final encrypted category $C_1'$ is produced by making up all cipher classes

$$C_1'(x, y) = \sum_{k=0}^{7} c_1'(x, y, k) \cdot 2^k$$  
(19)

Meanwhile, each class $C_2_i$ in categories $C_2$ is encrypted and compressed simultaneously through CS technology. The measurement matrix is created by TLCS, and initial parameter serving as encryption key. The implementation of encryption and compression is represented by following equations. Each class $C_2_i$, $(i = 0, 1, \ldots, c)$is transformed into a vector $\nu_2 = \{\nu(i, 1), \nu(i, 2), \ldots, \nu(i, \tau)\}$, then, CS technology is utilized to encrypt each vector $\nu_2$.
with data-hiding key $k_w$, because the size of the original image is decreased through CS technology. In Figure 6, $D$ represents the vacated spaces used for embedding additional data. To realize data extraction and image recovery, some auxiliary information as a part of payload should be embedded into the cipher image, which includes: (1) classes $c$ in interval $1,32$ with six bits by letter $p$ for representation, (2) cluster tag of each pixel in original image is marked $L$, (3) 6-bit delimiter “0 0 1 1 1” used to differentiate two cipher categories $C_1$, $C_2$ and each encrypted class in $C_1$, (4) $q$ bits of classes belonging to categories $C_1$. Thus, the embedding rate $r$ could be calculated by equation (24)

$$r = \frac{(1 - \rho) \times n \times 8 - L - p - q - 6 \times \text{count}}{m \times n}$$

where $n$ represents the number of pixels compressed by CS technology in categories $C_2$, and $\text{count}$ is the number of classes in $C_2$.

**Data extraction and image recovery**

In this section, there are three cases to handle data extraction and image recovery by receiver. (1) The decrypted operation can be accomplished only when encryption key is available. (2) Additional data can be extracted precisely only when data-hiding key is available. (3) Not only additional data can be obtained accurately, but also the decryption can be implemented when both encryption key and data-hiding key are available. The flowchart of this section is illustrated in Figure 7.

---

**Figure 4.** Classification results through FCM clustering algorithm with Lena image. (a) Experimental result with parameters $c = 13$ and $T = 10$ and (b) experimental result with parameters $c = 13$ and $T = 15$
Only encryption key is available. In this stage, five steps should be executed to accomplish image decryption.

**Step 1.** Final cipher image $C_e$ is decrypted with encryption key $k_e$ through exclusive-or operation by equation (25) to obtain cipher $C'$

$$C' = C_e \oplus k_e$$  \hspace{1cm} (25)

**Step 2.** According to the auxiliary information, $C'$ is classified into two encrypted parts $C1'$ and $C2'$.

**Step 3.** As for $C1'$, the identical operation with encryption process is executed to acquire decrypted category $C1$ accurately by equation (26)

$$C1 = C1' \oplus k_i$$  \hspace{1cm} (26)

**Step 4.** And for $C2'$, we reconstruct each class signal with overwhelming probability through solving the following convex optimization problem by equations (27) and (28)
\[
\min \| \tilde{\theta} \|, \text{ s.t. } y = \Phi \Psi \theta
\]  \hspace{1cm} (27)

\[
\tilde{v} = \Psi \tilde{\theta}
\]  \hspace{1cm} (28)

where \( \tilde{\theta} \) means the solution of optimization problem and \( \tilde{v} \) denotes reconstructed signal. We obtain the recovered category \( C_2 \) until all class is reconstructed.

**Step 5.** Combine the recovered category \( C_1 \) and \( C_2 \) to generate the final decrypted image \( I_o \). Note that, we cannot extract any additional information without data-hiding key.

**Only data-hiding key is available.** In this situation, we can just extract all additional data perfectly with data-hiding key without knowing any information about original content. For example, in Figure 6, we can directly extract all information in domain \( D \) with data-hiding key but we need not know anything about \( C_1 \) and \( C_2 \).

**Both encryption key and data-hiding key are available.** The receiver can not only decrypt cipher image and but also extract additional data with both encryption key and data-hiding key. The order of two operations is exchangeable and independent. That is, the receiver can decrypt cipher image with encryption key first by applying the procedure which is the same as section “Only encryption key is available,” and then extract additional data with data-hiding key as stated in section “Only data-hiding key is available,” and vice versa.

#### Experimental results and performance analysis

In this section, to show the effectiveness and superiority of our scheme, sufficient experimental results and comparisons are presented with a mass of standard images. Our experiment is conducted in a personal computer with a 3.20 GHz Inter i5 processor, 4.00 GB memory and MATLAB R2014a in windows 10 operating system. There are mainly two aspects of the experiment, that is, security analysis of cipher image, and performance analysis in terms of recovered image quality and embedding capacity.

**Security of cipher image**

**Histogram analysis.** Image histogram is an intuitive means to reflect the distribution status of pixels value in a gray image. In general, an effective encryption scheme should mask the pixel value distribution of original image and generate an absolutely uniform distribution.
Histograms of multiple plain images and their corresponding cipher images are given in Figure 8 to show the security performance of our encryption scheme. Figure 8(a1)–(h1) represents the original images and the corresponding encrypted images with crowd, woman, peppers, and plane. Figure 8(a2)–(h2) are the histograms corresponding to Figure 8(a1)–(h1) with cluster number $c = 16$, threshold $T = 13$, sampling rate $\rho = 0.4$. We can observe that the encrypted images become disordered and unsystematic, and the histogram of each encrypted image has the same uniform distribution, meaning that any information about original image wouldn’t be acquired without encryption key.

Information entropy analysis. In 1949, Shannon proposed the concept of information entropy to solve information quantification and measurement problem. In general, information entropy is used for measuring the uncertainty by calculating the probability of a
randomness variable appearing. The information entropy \( H(x) \) of source \( x \) can be calculated by equation (29)

\[
H(x) = -\sum_{i=1}^{K} p(x_i) \log_2(p(x_i))
\]  

(29)

where \( p(x_i) \) represents the probability of pixel \( x_i \), and \( K \) means the number of pixels in an image. Ideally, the value of information entropy closer to eight indicates more disordered state and greater uncertainty for a gray-scale image. Table 1 gives the entropy values of several test images and corresponding encrypted images, and comparisons with other literatures. It can be seen that all entropy values are close to eight after test images are encrypted, and the average value of information entropy reaches 7.9987, which is higher than in previous studies. Hence, our encryption scheme is capable of resisting entropy attacks.

**Correlation analysis.** An effective encryption scheme should obviously decrease the correlation between adjacent pixels of original image. 4000 pairs of adjacent pixels are selected randomly from original image and encrypted image to analyze the spatial correlations in three directions: horizontal, vertical and diagonal. They can be calculated according to equations (30)–(32)

\[
r_{xy} = \frac{E(x - E(x))(y - E(y))}{\sqrt{D(x)D(y)}}
\]  

(30)

\[
E(x) = \frac{1}{N} \sum_{i=1}^{N} x_i
\]  

(31)

\[
D(x) = \frac{1}{N} \sum_{i=1}^{N} (x_i - E(x))^2
\]  

(32)

where \( N \) is the amount of samples, \( x_i \) and \( y_i \) represent gray values of selected adjacent pixels in \( i \)th pair. In Table 2, it can be found that the pixel correlations in three directions are decreased significantly after encryption, and the cipher images obtained by the proposed encryption scheme has relatively lower correlation coefficients than in previous studies. An example with Lena image is given in Figure 9 to show distribution of pixel correlation. We can observe that the distribution of adjacent pixel pairs is fairly centralized in plain image but scattered in encrypted image.
Recovered image and embedding rate

In this section, two typical criterions are applied to evaluate quality of recovered images, which are PSNR and structural similarity index measurement (SSIM). We can calculate PSNR and SSIM by equations (33)-(38)

\[
\text{PSNR}(I, I') = 10 \times \log_{10} \frac{255^2}{\text{MSE}} \quad (33)
\]

\[
\text{MSE} = \frac{1}{M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} [I(x, y) - I'(x, y)]^2 \quad (34)
\]

where \(I(x, y)\) and \(I'(x, y)\) represent the pixels used for evaluation of original image \(I\) and recovered image \(I'\) at the coordinate \((x, y)\), respectively, and \(M \times N\) denotes the size of images \(I\) and \(I'\). In general, the value of PSNR closer to positive infinity means that the recovered image \(I'\) is more similar to original image \(I\).

\[
\text{SSIM}(I, I') = [l(I, I')^{\alpha} c(I, I')^{\beta} s(I, I')^{\gamma}] \quad (35)
\]

\[
l(I, I') = \frac{2\mu_I\mu_{I'}}{\mu_I^2 + \mu_{I'}^2 + c_1} \quad (36)
\]

\[
c(I, I') = \frac{\sigma_I + \sigma_{I'}}{\mu_I^2 + \mu_{I'}^2 + c_2} \quad (37)
\]

\[
s(I, I') = \frac{\sigma_I + \sigma_{I'}}{\sigma_I\sigma_{I'} + c_3} \quad (38)
\]

where \(l(I, I'), c(I, I')\) and \(s(I, I')\) are three contrast modules constituting the SSIM measurement system, that is, brightness, contrast, and structure. \(\mu_I, \mu_{I'},\) and \(\sigma_I, \sigma_{I'}\) denote the mean values and the standard deviations of images \(I\) and \(I'\). \(\sigma_{II'}\) represents the covariance of images \(I\) and \(I'\). \(c_1, c_2,\) and \(c_3\) are the constants. And \(\alpha > 0, \beta > 0, \gamma > 0\), in actual engineering calculation, we set \(\alpha = \beta = \gamma = 1\) and \(c_3 = c_2/2\). SSIM can also be calculated by equation (39)

\[
\text{SSIM}(I, I') = \frac{(2\mu_I\mu_{I'} + c_1)(\sigma_{II'} + c_2)}{(\mu_I^2 + \mu_{I'}^2 + c_1)(\mu_I^2 + \mu_{I'}^2 + c_2)} \quad (39)
\]

The SSIM value of two images \(I\) and \(I'\) falls in the interval ranging from 0 to 1. The larger the SSIM value, the smaller difference between original image \(I\) and recovered image \(I'\).

Quality of recovered image. Figure 10 shows the experimental results of original images and recovered images with some standard test images sized 512 × 512 under parameters \(c = 10, T = 5, \rho = 0.7\). PSNR and SSIM of the recovered image are provided to show the performance of the proposed scheme. Figure 10(a1)–(f1) represent original test images with Baboon, bridge, couple, plane, man, and Lena. Figure 10(a2)–(f2) are the recovered images corresponding to Figure 10(a1)–(f1). For example, Figure 10(a1) is the original Baboon image and Figure 10(a2) displays the recovered Baboon image with PSNR = 46.4522, SSIM = 0.9950. Figures 11–13 give the quality changes of recovered images with Barbara, peppers, and lake under the condition that two parameters keep invariable, but the other is changing. It can be
seen from Figures 11 and 12 that larger threshold and lower sampling rate could cause relatively poor image quality under the uniform parameters \((c, \rho)\) and \((c, T)\). However, in Figure 13, we found that the qualities of recovered images appear inconsistent change, going down first and then up with the changing of cluster number \(c\), since the number of clusters compressed by CS is more when the clusters \(c\) is large under parameter \(T = 12, \rho = 0.7\). But each cluster would have higher sparsity to achieve more excellent quality about recovered image with CS reconstructed algorithm when the number of clusters settled is plentiful enough. Thus, based on the experimental results, it can be concluded that the visual quality of recovered image is quite satisfactory since all PSNR values are larger than 30 dB and SSIM are close to 1.
Tables 3–6 list the embedding rates and PSNR values of recovered images with test images Lena, baboon, lake, and milkdrop under different parameter values. We set the values of parameter \( r \) as 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, and the number of clusters are 15 and 20, respectively. In addition, the threshold \( T \) is chosen to realize high embedding rate and good visual quality according to textural features of each image. It can be observed that if the embedding rate is increasing, corresponding visual quality of recovered image would be worse when the clusters number keep unchanged. Meanwhile, the embedding rate of each test image is different even under same condition because each image has unique textural feature and pixel correlation. In our scheme, the distortion of the host image is only caused by CS, and the embedding and extraction of additional data is completely reversible, which is the same as other reversible data-hiding schemes. Thus, we give the comparison results about maximum embedding rate in the proposed scheme and some reversible data hiding literatures with different images in Table 7. And it can be found that our scheme has higher embedding rate than the literatures 36–40 and in previous studies.12,15,44

**Figure 12.** Recovered images of peppers under parameters \( c = 20, T = 12 \) and different sampling rate \( r \). (a) \( r = 0.8 \) with PSNR = 41.2419, SSIM = 0.9670. (b) \( r = 0.7 \) with PSNR = 40.1987, SSIM = 0.9582. (c) \( r = 0.6 \) with PSNR = 39.2111, SSIM = 0.9638. (d) \( r = 0.5 \) with PSNR = 37.7777, SSIM = 0.9370. (e) \( r = 0.4 \) with PSNR = 37.4497, SSIM = 0.9670.

**Figure 13.** Recovered images of lake under parameters \( T = 12, r = 0.7 \) and different number of clusters \( c \). (a) \( c = 12 \) with PSNR = 45.9588, SSIM = 0.9921. (b) \( c = 16 \) with PSNR = 40.6543, SSIM = 0.9708. (c) \( c = 20 \) with PSNR = 39.7365, SSIM = 0.9701. (d) \( c = 24 \) with PSNR = 40.9996, SSIM = 0.9761. (e) \( c = 28 \) with PSNR = 42.1196, SSIM = 0.9810.

**Table 3.** Embedding rate of Lena image with parameter \( c, T, r \).

| \( r \) (bpp) | PSNR | \( T (c = 15) \) | \( T (c = 20) \) |
|--------------|------|----------------|----------------|
| \( c \) | 11   | 12             | 13             | 9     | 10             | 11             |
| 0.3          | 1.123, 28.137 | 1.427, 27.625 | 1.815, 27.396 | 1.037, 28.062 | 1.556, 26.643 | 1.470, 27.280 |
| 0.35         | 0.834, 34.120 | 1.199, 33.114 | 1.404, 33.899 | 0.669, 34.616 | 1.112, 33.199 | 1.043, 32.413 |
| 0.4          | 0.488, 39.348 | 0.859, 37.067 | 1.169, 37.743 | 0.495, 40.686 | 0.817, 40.024 | 0.832, 39.962 |
| 0.45         | 0.298, 39.221 | 0.400, 38.836 | 0.742, 38.203 | 0.231, 40.969 | 0.449, 40.346 | 0.585, 40.250 |
| 0.5          | −0.178, 39.803 | 0.331, 38.701 | 0.409, 38.592 | −0.281, 41.343 | −0.082, 40.526 | 0.160, 40.346 |
| 0.55         | −0.551, 40.311 | 0.064, 39.065 | 0.031, 39.030 | −0.574, 41.641 | −0.415, 40.786 | −0.238, 40.824 |
| 0.6          | −0.851, 40.710 | −0.471, 39.720 | −0.295, 39.354 | −0.828, 41.539 | −0.712, 41.150 | −0.553, 41.054 |

**Embedding capacity.** Tables 3–6 list the embedding rates and PSNR values of recovered images with test images Lena, baboon, lake, and milkdrop under different parameter values. We set the values of parameter \( r \) as 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, and the number of clusters are 15 and 20, respectively. In addition, the threshold \( T \) is chosen to realize high embedding rate and good visual quality according to textural features of each image. It can be observed that if the embedding rate is increasing, corresponding visual quality of recovered image would be worse when the clusters number keep unchanged. Meanwhile, the embedding rate of each test image is different even under same condition because each image has unique textural feature and pixel correlation. In our scheme, the distortion of the host image is only caused by CS, and the embedding and extraction of additional data is completely reversible, which is the same as other reversible data-hiding schemes. Thus, we give the comparison results about maximum embedding rate in the proposed scheme and some reversible data hiding literatures with different images in Table 7. And it can be found that our scheme has higher embedding rate than the literatures 36–40 and in previous studies.12,15,44

**Conclusion**

In this article, we proposed a privacy-preserved data-hiding scheme in encrypted domain by FCM clustering and CS technologies. Low-correlated classes of the...
original image are encrypted by traditional stream cipher, and high-correlated classes are encrypted and compressed by CS. Additional data can be embedded into the room vacated by CS in the high-correlated classes. The receiver can extract the embedded data and recover the encrypted image in a flexible way according to different keys. Vast experimental results and comparisons show that the proposed scheme has higher security performance, larger embedding rate, and better visual quality of recovered image than other related

Table 4. Embedding rate of baboon image with parameter c, T, ρ.

| r (bpp), PSNR | T (c = 15)  | T (c = 20)  |
|---------------|-------------|-------------|
|               | 11          | 12          | 13          | 9   | 10   | 11   |
| 0.3           | 1.457, 27.680 | 1.874, 27.084 | 2.029, 27.154 | 1.328, 26.502 | 1.436, 27.259 | 1.535, 26.871 |
| 0.35          | 1.241, 31.857 | 1.589, 31.327 | 1.700, 32.824 | 0.895, 34.897 | 1.144, 34.023 | 1.142, 28.741 |
| 0.4           | 1.166, 37.718 | 1.088, 37.756 | 1.357, 35.400 | 0.561, 37.827 | 0.664, 39.819 | 0.734, 35.983 |
| 0.45          | 0.541, 38.390 | 0.400, 38.836 | 0.949, 37.486 | 0.252, 40.715 | 0.422, 39.458 | 0.538, 40.152 |
| 0.5           | 0.109, 38.660 | 0.299, 38.225 | 0.460, 37.696 | -0.305, 40.900 | 0.120, 40.348 | -0.015, 40.533 |
| 0.55          | -0.020, 38.348 | -0.134, 38.293 | 0.087, 37.991 | -0.422, 41.061 | -0.273, 40.736 | -0.400, 40.850 |
| 0.6           | -0.558, 39.144 | -0.326, 38.748 | -0.311, 38.230 | -0.826, 41.521 | -0.923, 41.612 | -0.728, 41.133 |

Table 5. Embedding rate of lake image with parameter c, T, ρ.

| r (bpp), PSNR | T (c = 15)  | T (c = 20)  |
|---------------|-------------|-------------|
|               | 16          | 17          | 18          | 10  | 11   | 12   |
| 0.3           | 1.341, 27.966 | 1.636, 27.277 | 1.857, 28.045 | 0.617, 27.196 | 1.186, 29.040 | 1.328, 28.073 |
| 0.35          | 0.836, 31.753 | 1.260, 30.647 | 1.268, 32.839 | 0.225, 35.447 | 0.757, 34.229 | 1.041, 31.642 |
| 0.4           | 0.463, 34.765 | 0.379, 35.382 | 1.092, 32.827 | 0.124, 39.053 | 0.664, 39.819 | 0.954, 37.642 |
| 0.45          | 0.388, 35.996 | 0.707, 35.693 | 0.539, 34.190 | -0.261, 39.245 | 0.416, 37.667 | 0.432, 37.905 |
| 0.5           | -0.079, 37.125 | 0.070, 36.263 | 0.377, 35.795 | -0.498, 39.098 | -0.386, 38.976 | -0.006, 38.181 |
| 0.55          | -0.278, 36.992 | -0.121, 36.088 | 0.050, 36.043 | -0.530, 39.145 | -0.630, 38.949 | -0.560, 38.455 |
| 0.6           | -0.602, 37.184 | -0.360, 36.807 | -0.336, 36.373 | -1.424, 40.629 | -0.972, 39.250 | -0.822, 38.876 |

Table 6. Embedding rate of Milkdrop image with parameter c, T, ρ.

| r (bpp), PSNR | T (c = 15)  | T (c = 20)  |
|---------------|-------------|-------------|
|               | 17          | 19          | 20          | 15  | 16   | 17   |
| 0.3           | 1.530, 30.619 | 1.685, 31.151 | 2.367, 30.224 | 1.370, 29.034 | 1.359, 30.333 | 2.118, 28.570 |
| 0.35          | 1.215, 36.703 | 1.519, 31.820 | 1.825, 32.966 | 1.210, 34.705 | 1.192, 34.194 | 1.512, 33.510 |
| 0.4           | 0.954, 38.913 | 1.011, 38.224 | 1.612, 35.999 | 0.530, 40.190 | 0.785, 38.725 | 1.293, 38.892 |
| 0.45          | 0.552, 38.737 | 0.877, 38.006 | 1.206, 37.417 | 0.260, 40.344 | 0.525, 39.558 | 0.705, 39.119 |
| 0.5           | 0.080, 39.257 | 0.498, 38.877 | 0.625, 38.056 | -0.040, 40.727 | -0.167, 40.159 | 0.356, 39.464 |
| 0.55          | -0.114, 39.864 | -0.156, 39.180 | 0.146, 38.473 | -0.587, 41.007 | -0.331, 40.509 | -0.020, 39.735 |
| 0.6           | -0.721, 40.301 | -0.564, 39.560 | -0.373, 39.103 | -0.693, 41.211 | -0.689, 40.832 | -0.460, 40.251 |

Table 7. Maximum embedding rate in proposed scheme and some related literatures with different images.

| r_{max} (bpp) | 36 | 15 | 38 | 44 | 39 | 12 | 37 | 40 | proposed |
|---------------|----|----|----|----|----|----|----|----|---------|
|               | DHS_2 | DHS_3 | PHS_2 | PHS_3 | 2 × 2 | 3 × 3 |
| Lena          | 1.889 | 0.35 | 0.12 | 0.13 | 0.15 | 0.105 | 0.122 | 0.097 | 0.015 | 1.605 | 1.722 | 2.018 |
| Plane         | 2.280 | 0.35 | 0.19 | 0.21 | 0.19 | 0.163 | 0.186 | 0.148 | 0.023 | 1.710 | 1.903 | 2.188 |
| Man           | 1.797 | 0.01 | 0.05 | 0.04 | 0.12 | 0.080 | 0.095 | 0.071 | 0.014 | 1.293 | 1.327 | 1.528 |
| Crowd         | 1.456 | 0.35 | 0.21 | 0.16 | 0.11 | 0.175 | 0.201 | 0.175 | 0.023 | 1.484 | 1.511 | 1.728 |
|               |     |     |     |     |     |     |     |     | 1.814 |

International Journal of Distributed Sensor Networks
works. However, an obvious deficiency is the indistinctive improvement of recovered image quality caused by CS reconstruction even when the embedding capacity is lower. Thus, we expect especially that a more precise and appropriate CS reconstruction algorithm can be explored to optimize the visual quality of reconstructed image.

Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Natural Science Foundation of China (grant no. 61602158), the Key Scientific Research Plan of Henan Higher Education Institutions (grant no. 20A413007), and the PhD Scientific Research Foundation of Henan Normal University (grant no. 510119170143).

ORCID iD
Ming Li https://orcid.org/0000-0003-3385-8364

References
1. Qin C, Ji P, Zhang XP, et al. Fragile image watermarking with pixel-wise recovery based on overlapping embedding strategy. *Sig Process* 2017; 138: 280–293.
2. Li M, Wang LL, Fan JJ, et al. Fidelity preserved data hiding in encrypted highly autocorrelated data based on homomorphism and compressive sensing. *IEEE Access* 2019; 7: 69808–69825.
3. Petricolas FAP, Anderson RJ and Kuhn MG. Information hiding—a survey. *Proc IEEE* 1999; 87: 1062–1078.
4. Vleeschouwer CD, Delaigle JF and Macq B. Invisibility and application functionalities in perceptual watermarking: an overview. *Proc IEEE* 2002; 90: 64–77.
5. Zhang XP. Reversible data hiding with optimal value transfer. *IEEE Trans Multimedia* 2013; 15: 316–325.
6. Qin C, He ZH, Luo XY, et al. Reversible data hiding in encrypted image with separable capability and high embedding capacity. *Inform Sci* 2018; 465: 285–304.
7. Wang MX, Wang XY, Zhang YQ, et al. A novel chaotic encryption scheme based on image segmentation and multiple diffusion models. *Opt Laser Technol* 2018; 108: 558–573.
8. Fu C, Huang JB, Wang NN, et al. A symmetric chaos-based image cipher with an improved bit-level permutation strategy. *Entropy* 2014; 16: 770–788.
9. Liu HJ and Wang XY. Color image encryption using spatial bit-level permutation and high-dimension chaotic system. *Optics Commun* 2011; 284: 3895–3903.
10. Li M, Lu DD, Wen WY, et al. Cryptanalyzing a color image encryption scheme based on hybrid hyper-chaotic system and cellular automata. *IEEE Access* 2018; 6: 47102–47111.
11. Cao XC, Du L, Wei XX, et al. High capacity reversible data hiding in encrypted images by patch-level sparse representation. *IEEE Trans Cybern* 2016; 46: 1132–1143.
12. Huang FJ, Huang JW and Shi YQ. New framework for reversible data hiding in encrypted domain. *IEEE Trans Inform Forensic Secur* 2016; 11: 2777–2789.
13. Qian ZX, Zhang XP and Feng GR. Reversible data hiding in encrypted images based on progressive recovery. *IEEE Signal Process Lett* 2016; 23: 1672–1676.
14. Qian ZX, Zhang XP, Ren YL, et al. Block cipher based on separable reversible data hiding in encrypted images. *Multimedia Tool Appl* 2016; 75: 13749–13763.
15. Wu XT and Sun W. High-capacity reversible data hiding in encrypted images by prediction error. *Sig Process* 2014; 104: 387–400.
16. Yin ZX, Abel A, Tang J, et al. Reversible data hiding in encrypted images based on multi-level encryption and block histogram modification. *Multimedia Tool Appl* 2017; 76: 3899–3920.
17. Li M, Xiao D and Zhang YS. Reversible data hiding in block compressed sensing images. *Etri J* 2016; 38: 159–163.
18. Zhang W, Ma KD and Yu NH. Reversibility improved data hiding in encrypted images. *Sig Process* 2014; 94: 118–127.
19. Hong WE, Chen TS and Wu HY. An improved reversible data hiding in encrypted images using side match. *IEEE Signal Process Lett* 2012; 19: 199–202.
20. Xu DW and Wang RD. Separable and error-free reversible data hiding in encrypted images. *Sig Process* 2016; 123: 9–21.
21. Chai XL, Fu XL, Gan ZH, et al. A color image cryptosystem based on dynamic DNA encryption and chaos. *Sig Process* 2019; 155: 44–62.
22. Middya R, Chakravarty N and Naskar M. Compressive sensing in wireless sensor networks—a survey. *IETE Tech Rev* 2016; 34: 642–654.
23. Zhang YS, Zhang LY, Zhou JT, et al. A review of compressive sensing in information security field. *IEEE Access* 2016; 4: 2507–2519.
24. Gong LH, Qiu KD, Deng C, et al. An image compression and encryption algorithm based on chaotic system and compressive sensing. *Opt Laser Technol* 2019; 115: 257–267.
25. Wang Q, Lin D, Yang P, et al. An energy-efficient compressive sensing-based clustering routing protocol for WSNs. *IEEE Sens J* 2019; 19: 3950–3960.
26. Hua G, Xiang Y, Bi G, et al. When compressive sensing meets data hiding. *IEEE Signal Process Lett* 2016; 23: 473–477.
27. Xiao D and Chen S. Separable data hiding in encrypted image based on compressive sensing. *Electron Lett* 2014; 50: 598–600.
28. Xiao D, Cai HK, Wang Y, et al. High-capacity separable data hiding in encrypted image based on compressive sensing. *Multimedia Tool Appl* 2016; 75: 13779–13789.
29. Xiao D, Zhao J, Wang M, et al. Controllable high-capacity separable data hiding in encrypted images by
compressive sensing and data pretreatment. Multimedia Tool Appl 2018; 77: 23949–23968.

30. Liao X, Li KD and Yin JJ. Separable data hiding in encrypted image based on compressive sensing and discrete Fourier transform. Multimedia Tool Appl 2017; 76: 20739–20753.

31. Li M, Fan H, Ren H, et al. Meaningful image encryption based on reversible data hiding in compressive sensing domain. Secur Commun Netw 2018; 2018: 1–12.

32. Song YJ, Zhu ZL and Zhang W. Joint image compression–encryption scheme using entropy coding and compressive sensing. Nonlinear Dyn 2019; 95: 2235–2261.

33. Donoho DL. Compressed sensing. IEEE Trans Inf Theory 2006; 52: 1289–1306.

34. Wang JX, Mao NX and Chen X. Multiple histograms based reversible data hiding by using FCM clustering. Sig Process 2019; 159: 191–203.

35. Bezdek JC, Ehrlich R and William F. FCM: the fuzzy c-means clustering algorithm. Comput Geosci 1984; 10: 191–203.

36. Tang ZJ, Xu SJ, Yao H, et al. Reversible data hiding with differential compression in encrypted image. Multimedia Tool Appl 2018; 78: 9691–9715.

37. Liu ZL and Pun CM. Reversible data-hiding in encrypted images by redundant space transfer. Inform Sci 2018; 433–434: 188–203.

38. Yin Z, Luo B and Hong W. Separable and error-free reversible data hiding in encrypted image with high payload. Sci World J 2014; 2014: 604876.

39. Zhou J. Secure reversible image data hiding over encrypted domain via key modulation. IEEE Trans Circuits Syst Video Technol 2016; 26: 441–452.

40. Yi S and Zhou YC. Separable and reversible data hiding in encrypted images using parametric binary tree labeling. IEEE Trans Multimedia 2019; 21: 51–64.

41. Chen JX, Zhu ZL, Fu C, et al. An efficient image encryption scheme using lookup table-based confusion and diffusion. Nonlinear Dyn 2015; 81: 1151–1166.

42. Hu GQ, Xiao D, Wang Y, et al. Cryptanalysis of a chaotic image cipher using Latin square-based confusion and diffusion. Nonlinear Dyn 2017; 88: 1305–1316.

43. Li M, Lu DD and Xiang Y. Cryptanalysis and improvement in a chaotic image cipher using two-round permutation and diffusion. Nonlinear Dyn 2019; 96: 31–47.

44. Yin Z, Wang H, Zhao H, et al. Complete separable reversible data hiding in encrypted image. In: Cloud computing and security, Nanjing, China, 13–15 August 2015, pp.101–110. Cham: Springer.