computers

MDPI

Article

High-Capacity Reversible Data Hiding Based on Two-Layer Embedding Scheme for Encrypted Image Using Blockchain

Arun Kumar Rai [1], Hari Om [1], Satish Chand [2] and Chia-Chen Lin [3,*]

[1] Indian Institute of Technology, Indian School of Mines, Dhanbad 826004, India; arunrai.dei@gmail.com (A.K.R.); hariom4india@gmail.com (H.O.)
[2] School of Computer & Systems Science, Jawaharlal Nehru University, New Delhi 110067, India; schand20@gmail.com
[3] Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taichung 411030, Taiwan
* Correspondence: ally.cclin@ncut.edu.tw

Abstract: In today's digital age, ensuring the secure transmission of confidential data through various means of communication is crucial. Protecting the data from malicious attacks during transmission poses a significant challenge. To achieve this, reversible data hiding (RDH) and encryption methods are often used in combination to safeguard confidential data from intruders. However, existing secure reversible hybrid hiding techniques are facing challenges related to low data embedding capacity. To address these challenges, the proposed research presents a solution that utilizes block-wise encryption and a two-layer embedding scheme to enhance the embedding capacity of the cover image. Additionally, this technique incorporates a blockchain-enabled RDH method to ensure traceability and integrity by storing confidential data alongside the hash value of the stego image. The proposed work is divided into three phases. First, the cover image is encrypted. Second, the data are embedded in the encrypted cover image using a two-layer embedding scheme. Finally, the stego image along with the hash value are deployed through blockchain technology. The proposed method reduces challenges associated with traceability and integrity while increasing the embedding capacity of images compared to traditional methods.

Keywords: steganography; reversible data hiding; two-layer embedding scheme (2LES); blockchain

Citation: Rai, A.K.; Om, H.; Chand, S.; Lin, C.-C. High-Capacity Reversible Data Hiding Based on Two-Layer Embedding Scheme for Encrypted Image Using Blockchain. *Computers* **2023**, *12*, 120. https://doi.org/10.3390/computers12060120

Academic Editors: Aditya Kumar Sahu, Amine Khaldi and Jatindra Kumar Dash

Received: 14 April 2023
Revised: 17 May 2023
Accepted: 24 May 2023
Published: 12 June 2023

1. Introduction

During times of war, the effective exchange of information between different departments of the armed forces is critical in managing the overall situation. While the digitization of defense-related data has shown rapid growth in recent times, ensuring its security remains a pressing concern. As part of the digitization process, data are stored in a digitized format before being transmitted between departments. However, the transformation of this data during transmission raises important security concerns. To address these concerns, the concept of data hiding offers a promising solution for securing defense-related data. By implementing data hiding techniques, the security and efficiency of data transmission can be significantly improved between different ends [1,2].

To ensure secure data transmission, two commonly used techniques are cryptography and steganography. Cryptography involves the use of algorithms such as advanced encryption standard (AES), digital encryption system (DES), and elliptic curve cryptography (ECC) to encrypt data, ensuring its security during transmission. On the other hand, steganography involves embedding data into different media such as images, audio, and video. This technique can be reversible or irreversible, depending on whether the cover image needs to be recovered or not. Reversible data hiding (RDH) is gaining popularity for its ability to securely send and receive data [3–5]. In RDH, confidential data are embedded into the cover image before transmission, which is different from traditional cryptographic

methods. The use of RDH is preferred over traditional cryptographic methods due to its effectiveness in maintaining data confidentiality. Thus, the use of RDH is a promising solution to address security concerns associated with the transmission of sensitive data [6–18]. In the case of the RDH technique, recovery of embedded data along with its original cover image is ensured at the destination end. The defense sector generates a wide range of data, such as information about the deployment of armed forces, geographical locations of armed bases, and locations used for storing weapons, among others. It is essential to handle this data consistently, reliably, and with the utmost security.

Currently, the field of reversible data hiding in encrypted images (RDHEI) presents itself as a promising area for research. This methodology combines cryptography and steganography to ensure secure data transmission through images [19–29]. There are two common techniques for implementing RDHEI: vacating room after encryption (VRAE) and vacating room before encryption (VRBE) [6]. In VRAE, data embedding occurs after the cover image is encrypted, while in VRBE, data embedding is performed after encryption of the cover image. However, using the RDHEI technique for secure data transmission through a cover image raises a major concern of maintaining the integrity of the cover image. In some cases, intruders may modify the pixel values of the cover image, which can lead to potential security breaches. Therefore, ensuring the integrity of the cover image is crucial for the success of this technique. This concern has opened up a new area of research for various researchers who are exploring ways to overcome this challenge and ensure that the integrity of the cover image is not compromised during the data transmission process.

While using steganography for secure data transmission using cover images, one major concern is the potential for an intruder to modify the pixel values of the cover image, compromising its integrity. This issue has opened up avenues of research to develop more secure and reliable techniques for steganography and to improve the integrity of cover images used in the transmission of sensitive data. When using the aforementioned techniques for secure transmission of defense data through military cover images, ensuring integrity and traceability of the cover image poses a significant challenge. It is crucial to address these concerns to ensure that the cover image is not compromised during transmission and that it can be traced back to its original source. This area of research offers opportunities for various researchers to explore potential solutions for enhancing the integrity and traceability of cover images used in secure data transmission.

Hence, to address the challenges mentioned earlier, researchers have proposed the use of blockchain technology, which is increasingly being adopted as a solution to overcome data integrity and traceability issues. Blockchain is a distributed ledger technology that has gained popularity in recent years [30–39]. It is particularly useful in scenarios where trust is a concern among various parties. In a blockchain, data arestored in the form of a chain where each block is concatenated with a hash value. Because blockchain works in a decentralized manner, the ledger containing verified transactional data is stored locally. This makes the system more secure and reliable, as it is extremely difficult for an intruder to modify the data without being detected.

The proposed work focuses on the implementation of the reversible data hiding encrypted image (RDHEI) technique for secure data transmission. To further enhance the security and reliability of data transmission with military cover images in various defense operations, we propose the use of blockchain technology. Our approach involves concatenating the output generated through RDHEI with its hash value and deploying it through blockchain technology, thus ensuring integrity and traceability. The research paper presents several innovative contributions in the field of secure data transmission.

1. Firstly, a two-layer embedding-scheme-based high-capacity reversible data hiding in encrypted image (RDHEI) methodology is proposed which enhances the embedding capacity as compared to Zhang et al.'s [20] embedding process.
2. Secondly, the work highlights the use of blockchain technology to infuse the RDH method with an enhanced protection level, making data transmission more secure.

3. Furthermore, the work emphasizes the importance of ensuring the integrity of the cover image while using RDH with blockchain technology. To address this issue, the proposed solution uses the hash value within the blockchain technology to accomplish the integrity of the cover image.

The rest of the paper is organized as follows. In Section 2, the background and the literature review of the existing work are discussed. The proposed scheme is discussed in Section 3 and results are discussed in Section 4. Finally, the paper is concluded in Section 5.

2. Background and Literature

The background and the literature section is divided into two sections. In the first section, we will discuss the literature related to the data hiding for the encrypted images. Later on, we discuss the background and the literature related to the blockchain technology for providing more security during communication.

Methods for data hiding in the encrypted images: In the past, steganography has been used to hide secret data by utilizing cover images [1–4,14]. It is essential that the quality of the cover image is not distorted (i.e., remains approximately the same) while embedding the secret data. Therefore, only a few or some bits can be allowed to embed to ensure that the quality of the cover image is not significantly affected. However, the limited embedding capacity of the cover image has prompted researchers to seek ways to enhance its capacity. As time passes, it has become apparent that the recovery of the cover image is also an important aspect during the extraction of secret data. This concept plays a vital role in the deployment of various applications in the field of defense services, healthcare services, and more. To achieve this task, researchers have proposed various algorithms, known as reversible data hiding algorithms. Commonly used reversible data hiding techniques include lossless [2,5–7,17,18], histogram shifting [20], pixel value difference [8,10,11], prediction error expansion [3,10–12,15,16], and pixel value ordering [4,8,14], among others. The categorization of the techniques is shown in Figure 1.

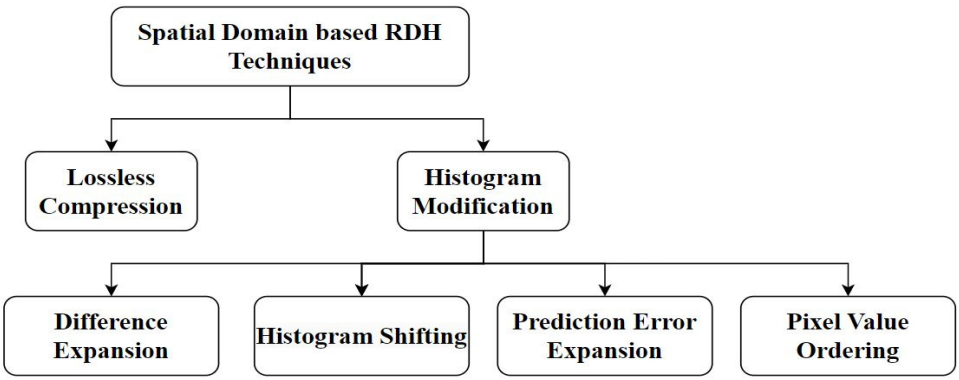


Figure 1. Different types of RDH techniques.

The shifting of existing technologies to cloud computing has also opened up new areas of research in the field of data hiding. In cloud computing, data hiding is conducted by a third party known as a data hider. Due to the introduction of the third party, the security of the cover image becomes a challenge in cloud computing. To resolve this challenge, the RDHEI technique is used, where the sender sends the cover image in an encrypted form to the data hider (i.e., third party) for embedding of secret data. The encrypted form of the cover image provides an extra layer of security, ensuring that the cover image remains secure even when it is in the possession of the third party. This technique ensures that the integrity and confidentiality of the cover image are maintained throughout the data hiding process, making it more suitable for use in cloud computing environments.

In the field of reversible data hiding, researchers have proposed various techniques to embed secret data into cover images while minimizing the distortion of the cover image. For instance, Ou et al. [15] have proposed the pairwise prediction expansion technique,

while researchers [4,13] have proposed adaptive pairing schemes. Wang et al. [19] have introduced a novel data hiding scheme based on the significant bit difference expansion (SBDE) technique. Additionally, efficient schemes based on image data hiding have been proposed by authors [11–14]. These techniques have contributed to the development of high-capacity and low-distortion reversible data hiding algorithms.

In papers [8,10,11], various techniques for data embedding based on pixel value differencing have been introduced. Puech et al. in their research [17] discussed a bit substitution scheme that enables the embedding of confidential data in an encrypted image. Zhang [20] proposed a new RDH scheme where data and cover image recovery are performed using separate keys. Mallik [24] et.al. proposed a new model based on the RDHEI scheme which makes use of the prediction error technique. Research [4] demonstrated the implementation of RDH techniques using medical images. Zhang et al. [20] proposed a method based on encryption and permutation on image blocks, which is considered the most efficient among existing strategies as shown in Figure 2. In this method, all the pixels are classified into three categories based on their locations and each category of pixels uses different predictors to compute prediction errors. Data embedding is performed using the histogram expansion technique and histogram shifting process. However, this research [20] has some drawbacks, such as shifting of pixel gray level and modification of spatial location while conducting image encryption, overflow problems, inability of the public key encryption system to share the data, low embedding capacity, tradeoff image fidelity, etc.

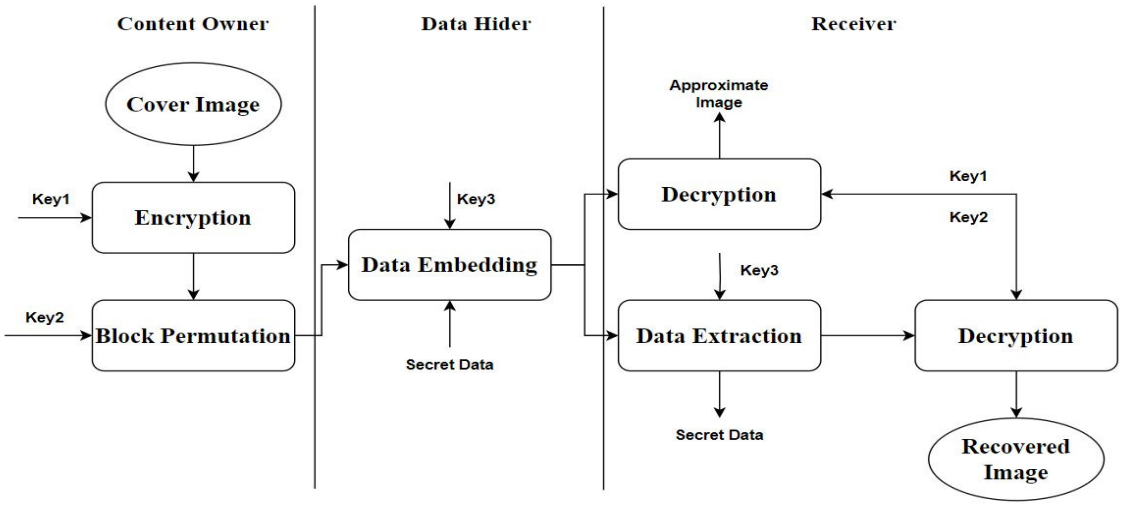


Figure 2. RDHEI Method proposed by Zhang et al. [20].

Kumar et al. propose a multimedia information hiding method for AMBTC compressed images using LSB substitution [40]. They employ quantization levels of color images and an adaptive RDH (ARDH) scheme that selects an embedding strategy based on the image subblock category [41]. Zhao et al. present a privacy-preserving federated learning method with fault detection [42,43]. Brahma et al. discuss reversible data hiding using lower magnitude error channel pair selection [44]. They improve pixel selection by considering the local variance of color channels and the grayscale image. Another paper [45] conducts a comprehensive study on reversible data hiding schemes based on pixel value ordering (PVO). Kaur et al. propose a two-pass technique for data embedding [46]. In pass 1, prediction errors are calculated for the first and last pixels, modifying pairwise mapping for embedding. In pass 2, the middle pixel predicts first and last pixels, adjusting their values. Kumar et al. discuss low-bandwidth data hiding for multimedia systems based on bit redundancy [46]. Their scheme utilizes complex image blocks to embed secret data without degrading image quality.

Methods for data hiding with blockchain technology: Blockchain is a concept which can be implemented through a distributed platform where a shared ledger can be used to

store various transactions and secure the information [32–35]. It provides multiple benefits when different parties are communicating with each other (without any trust) through a distributed network. In this technology, confidential data are stored in the block where secret data are concatenated with a related hash value, i.e., signature [35], as shown in Figure 3. Here, each block is of a unique hash value, and for generating the hash value of the current block, we use the below-mentioned formula where the hash value of the previous block is used as an input for generating the hash value of the current block. Thus, this process keeps the data safe using the blockchain's mechanism as shown in Figure 4.

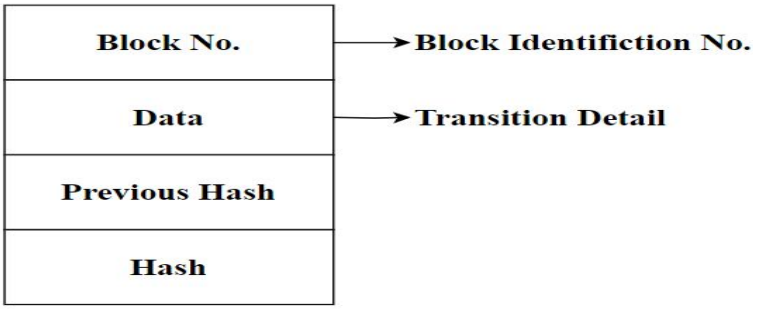


Figure 3. Individual Block.

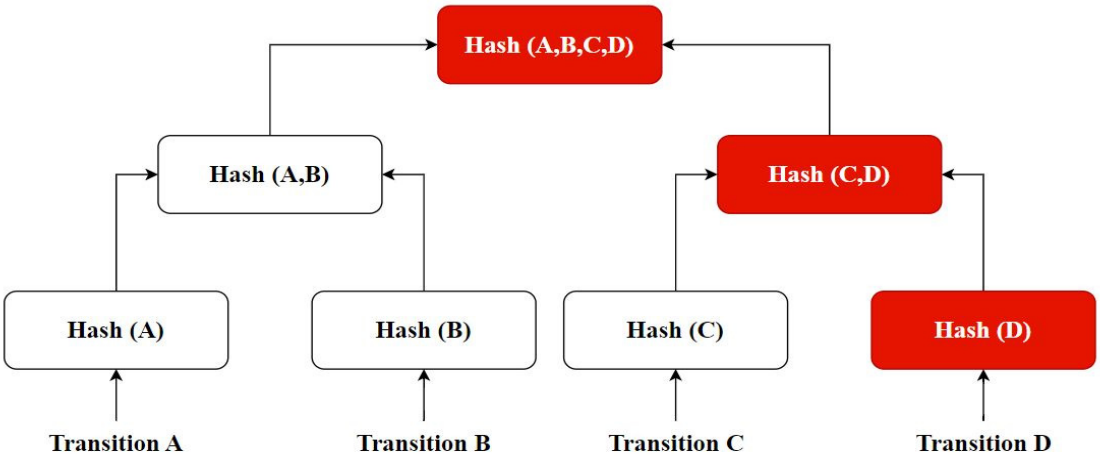


Figure 4. Keeping data safe using the blockchain's mechanism.

The hash function is written as:

$$hash_{new} = F(hash_{old}, transition\_detail)$$

where F represents the hash function, $hash_{new}$ represents the hash value of the current block, $hash_{old}$ is the hash value of the previous block, and $transition\_detail$ is the data we are going to use.

The hash function shows the avalanche effect, i.e., a small change in the input may show a higher change in the output in Figure 5. Additionally, this is aone-way process, i.e., vice versa is not possible. Blockchain also makes use of consensus protocols, for example, PoW (proof of work), PoS (proof of stake), and the Byzantine generals problem [31].

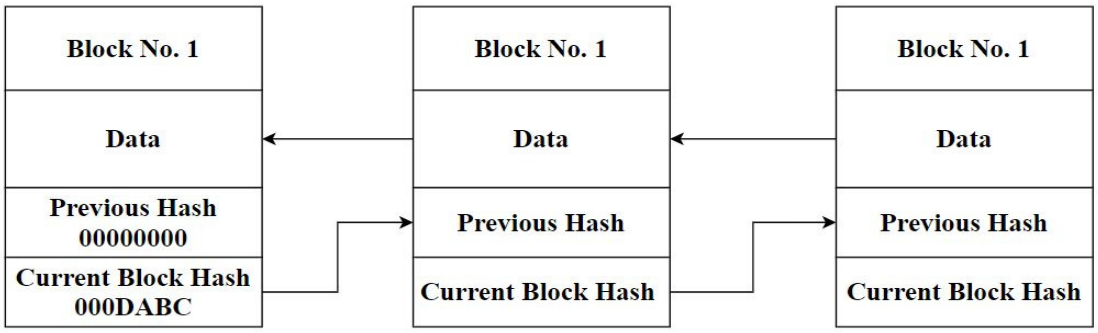


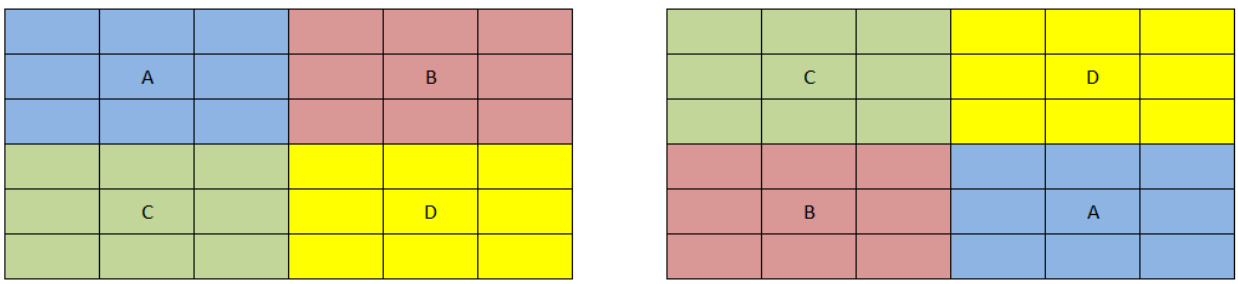
Figure 5. Function of blockchain.

To address the aforementioned issues, we propose a novel data hiding technique. Our method is based on the two-layer embedding scheme (2LES) which allows for high-capacity reversible data hiding in encrypted images (RDHEI) while preserving the quality of the cover image. We also incorporate blockchain technology to enhance the protection and traceability of the transmitted data, especially in military operations. Our approach includes the concatenation of the output generated through RDHEI with its hash value, which is then deployed through blockchain technology. Thus, our proposed technique provides a more secure and efficient way of transmitting data with improved embedding capacity and cover-image fidelity.

3. Proposed Scheme

The complete working of the proposed process is divided into three phases which are given as follows:

First Phase: In this phase, the cover image is first divided into non-overlapping blocks, each containing 9 pixels arranged in a 3×3 grid. These blocks are then reordered using a key called "k1", which has a constant size for all the blocks. The process of re-ordering the block sequence is shown in Figure 6 as before Figure 6a and after Figure 6b. The key is derived from the cover image itself to ensure that the resulting encrypted image is more resistant to attacks. This reordering step helps to increase the security of the encrypted image by introducing additional complexity and randomness. Once the blocks have been reordered, the image is encrypted using a key called "k2", which has a variable size depending on the desired level of security. The size of "k2" can range from 64 to 256 bits, with larger key sizes providing greater security against brute-force attacks. After encryption, the image undergoes a preprocessing step, where the pixel values of each block are converted to their equivalent gray code. This conversion helps to further increase the security of the encrypted image by reducing the impact of noise and other distortions that may occur during transmission or storage.



(**a**) (**b**)

Figure 6. Block sequence before (**a**) and after (**b**) re-ordering.

Second Phase: In the second phase, the secret data are embedded into the encrypted cover image using a two-layer embedding scheme (2LES). This method helps to ensure that

the secret data are securely embedded into the image, while also minimizing the risk of data loss or corruption.

Third Phase: The final phase involves retrieving the original data and cover image from the encrypted stego image. At the receiver end, the secret data are accessed from the stego image using a reverse process of the 2LES method. This process involves using the same keys "k1" and "k2" that were used during the encryption and embedding phases. The block diagram of the proposed RDHEI technique is given in Figure 7 and the complete flow diagram of the proposed work is given in Figure 8.

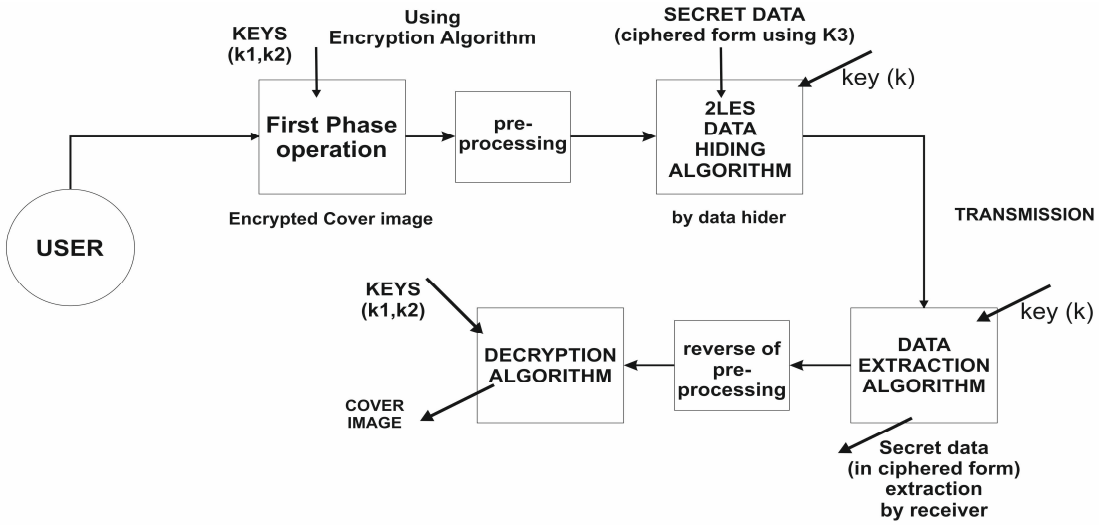


Figure 7. Block Diagram of Proposed RDHEI Technique.

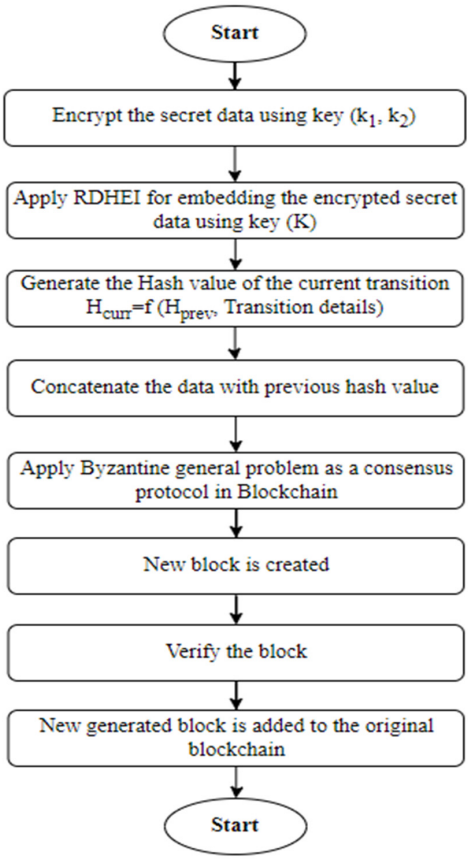


Figure 8. Flowchart of proposed work.

3.1. Embedding Algorithm

The proposed embedding algorithm is explained in the *Algorithm 1* using the following steps:

Algorithm 1: Proposed Embedding algorithm

Input for Algorithm: Military cover image, secret data (encrypted form)
Keys used: k1, k2, k3, and k.
Output: Stego image

(**Step 1**) The first step involves splitting the military cover image into non-overlapping blocks with a size of 9 pixels each, in the form of 3 × 3 blocks. This process is performed to facilitate the embedding of secret data into the image.
(**Step 2**) After the blocks have been split, the next step involves re-ordering the blocks according to the order specified by key "k1". This key is a constant value that is the same for all blocks.
(**Step 3**) The reordered image blocks are then encrypted using a symmetric key encryption algorithm, with the key "k2". The size of the key "k2" depends on the desired security level and can range from 64 to 256 bits.
(**Step 4**) The next step is preprocessing the blocks by converting the pixel values from binary to gray scale. This is conducted to ensure that the image blocks are ready for data embedding.
(**Step 5**) Before embedding the secret message, it is first encrypted using a symmetric key encryption algorithm, with the key "k3". This adds an additional layer of security to the secret message.
(**Step 6**) In this step, a two-layer embedding scheme is applied to embed the encrypted secret message into the encrypted cover image. The two-layer embedding scheme involves embedding the data in the least significant bit (LSB) and second LSB planes of the image blocks. The embedding process uses the key "k" to ensure that the secret message is embedded in the correct blocks.
(**Step 7**) The final step in the process is the generation of the stego image, which is an encrypted version of the military cover image with the secret message embedded within it. The output of the algorithm is the stego image, which is sent to the intended receiver.

3.2. Illustration of the Embedding Process Example

The embedding algorithm considers a cover image with a resolution of 512 × 512 pixels. A block of size 3 × 6 is selected from the cover image for processing as shown in Figure 9. This block is split into two non-overlapping blocks of a size of 9 pixels each, using a 3 × 3 block formation. This split is performed without overlapping between the blocks. The first block selected from the cover image consists of 9 pixels with values of 98, 100, 101, 101, 99, 102, 100, 102, and 101. Similarly, the second block selected from the cover image consists of 9 pixels with values of 100, 100, 99, 101, 100, 102, 100, 102, and 101.

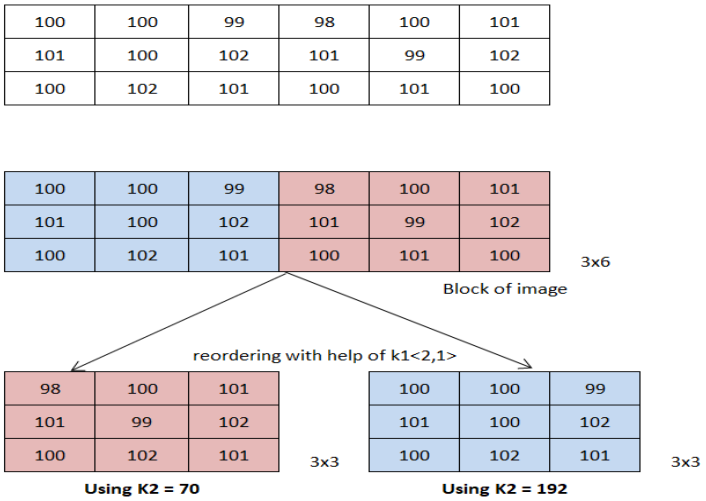


Figure 9. Block re-ordering.

The security key k2 is assigned values of 70 and 192 for both blocks. To embed secret data, each pixel value is subjected to an XOR operation with the security key k2 of the respective block. The updated values of the pixels in the first block are 36, 34, 35, 35, 37, 32, 34, 32, and 35, while those of the second block are 164, 164, 163, 165, 164, 166, 164, 166, and 165 as shown in Figure 10.

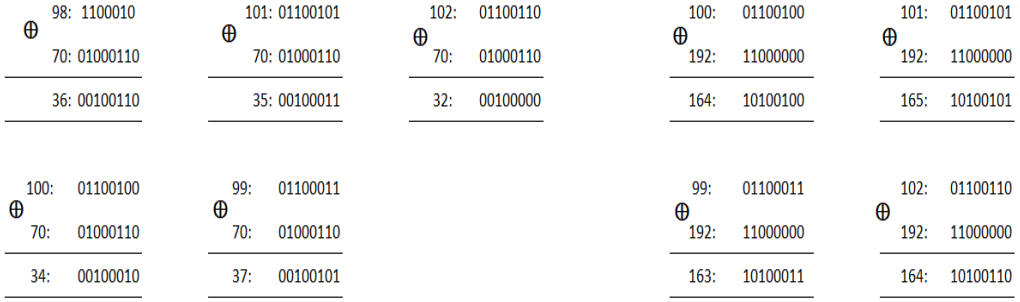


Figure 10. Image Encryption with K2.

The next step is to apply the processing on the encrypted image. The binary code values of the first block (36, 34, 35, 35, 37, 32, 34, 32, 35) and second block (164, 164, 163, 165, 164, 166, 164, 166, 165) are then converted into gray code values. The updated gray code values of the first block are 54, 51, 50, 50, 55, 48, 51, 48, and 50, while those of the second block are 246, 246, 242, 247, 246, 245, 245, 245, and 247. In the data embedding phase, the gray code block is divided into two parts based on the least significant bit (LSB) and most significant bit (MSB) values. The MSB block uses 6 bits, while the LSB block uses 2 bits. As a result, the new values of the first block are 13, 12, 12, 12, 13, 12, 12, 12, and 12, while those of the second block are 2, 3, 2, 2, 3, 0, 3, 0, and 2. These new values are generated from the gray code block.

In the first layer embedding of the 2LES scheme, if the secret data bits are "10" and we want to embed data in the pixel value $x_{ij} = 13$, we first pick all the adjacent pixels of $x_{ij} = 13$ and arrange them in ascending order by considering the inequality $v1 \leq v2 \leq v3 \leq v4$ ($12 \leq 12 \leq 12 \leq 12$). Then, we calculate the average pixel values $\bar{P}_1$ and $\bar{P}_2$ as $\bar{P}_1 = \left\lfloor \frac{v1+v2+v3}{3} \right\rfloor$ and $\bar{P}_2 = \left\lceil \frac{v2+v3+v4}{3} \right\rceil$, respectively. In this case, $\bar{P}_1 = \bar{P}_2$, then $\bar{P}_2 = \bar{P}_1 + 1$, which means $\bar{P}_1 = \frac{12+12+12}{3} = 12$ and $\bar{P}_2 = \frac{12+12+12}{2} = 12$. Therefore, $\bar{P}_1 = 12, \bar{P}_2 = 13$. Here, $e_1 = \bar{P}_2 - \bar{P}_1 = 13 - 12 = 1$. Because the e_1 value is 1, we embed the data in the current pixel $x_{(i,j)} = 13$ as $x_{(i,j)} + b = 13 + 1 = 14$. Thus, the updated pixel value will be $\bar{x}_{(i,j)} = 14$. Therefore, the updated block after the first layer embedding will be as follows:

In the second layer embedding of the scheme, the first step is to calculate the error $\bar{e}_{(i,j)} = \bar{x}_{(i,j)} - \bar{P}_2$, where $P_2 = 13$ and $\bar{x}_{(i,j)} = 14$, resulting in $\bar{e}_{(i,j)} = 14 - 13 = 1$. If $\bar{e}_{(i,j)} = -1$, then the new pixel value will be $\bar{\bar{x}}_{(i,j)} = \bar{x}_{(i,j)} - b_2$, where b_2 is the secret data bit to be embedded in the second layer. If $\bar{e}_{(i,j)} < -1$, then the new pixel value will be $\bar{\bar{x}}_{(i,j)} = \bar{x}_{(i,j)} - 1$. Otherwise, if $e_{(i,j)} > -1$, then there is no change in the pixel value during embedding, and the updated pixel value is $\bar{\bar{x}}_{(i,j)} = \bar{x}_{(i,j)}$.

Thus, the updated block after the second layer embedding will be (13,12,12,12,4,12,12,12,12) and (2,3,2,2,3,0,3,0,2).

Now the stego image is generated by concatenating the bits of I_{msb} and I_{lsb} as 54, 51, 50, 50, 63, 48, 51, 48, and 50.

3.3. Extraction Algorithm

To extract and recover the cover image, we reverse the process used for embedding the secret data. This entire process is depicted in the Algorithm 2.

Algorithm 2: Proposed Extraction algorithm

Input for Algorithm: Stego image (encrypted coverimage with secret data)
Keys used: k1, k2, and k3
Output: Military cover image, secret data

(**Step1**) Firstly, we extract the data from the second layer in parts, and then the remaining data areextracted from the first layer. This process involves decoding the hidden message and extracting it from the image using a decryption algorithm.
(**Step2**) Once the secret data are extracted, we use the key "k3" to decrypt the data and convert the mback to their original form. This step is crucial to recover the actual secret message from the steganographic image. Without proper decryption, the message will remain in a scrambled or encrypted form.
(**Step3**) In this step, we convert the block pixels from gray scale to binary scale. This conversion is necessary to perform further operations on the image data. The conversion process involves assigning a binary code to each pixel value, which allows us to store the data more efficiently.
(**Step4**) In this step, the blocks of pixel data are reordered using the key "k2". This step is essential to restore the original order of the blocks, which may have been altered during the embedding process. The use of the key "k2" ensures that the blocks are reordered correctly, and the original image can be reconstructed.
(**Step5**) The final step in the process involves decrypting the block pixels using the key "k1" to obtain the cover image in its original form. This step is crucial to reconstruct the original image, which may have been modified during the embedding process to hide the secret message. The use of the key "k1" ensures that the original image is restored with minimal distortion.

3.4. Illustration of the Extraction Process Example

In this section, an illustration of the extraction process example is discussed. Here, we perform data extraction, which is the reverse process of the embedding process used in the previous section. Now, as per the second layer, I_{msb} and I_{lsb} are generated from the stego image 54, 51, 50, 50, 63, 48, 51, 48, and 50 as (13,12,12,12,4,12,12,12,12) and (2,3,2,2,3,0,3,0,2), respectively.

The two-layer embedding scheme (2LES) involves a two-stage data extraction process. First, we extract data from the second layer, followed by extracting the remaining data from the first layer of 2LES. To extract data from the second layer, we calculate the value of E by subtracting $\bar{P}_2$ from $\bar{\bar{x}}_{(i,j)}$, where $\bar{\bar{x}}_{(i,j)} = 14$, $\bar{P}_1 = 12$, and $\bar{P}_2 = 13$. In this case, $E = \bar{\bar{x}}_{(i,j)} - P_2 = 14 - 13 = 1$, which is greater than -1. Therefore, no bits are embedded, meaning that b_{ij} remains unchanged after the second phase of decoding.

Now we extract the data bits from the first phase of the embedding process. The value of the extracted bit, $\bar{b}_{(i,j)}$, depends on the calculated value of E', which is determined by subtracting the value of the pixel $\bar{P}_1$ from the value of $\bar{x}_{(i,j)}$. If $E' = 1$, the extracted bit will be 0, and if $E' = 2$, the extracted bit will be 1. If E' is greater than or equal to 2, then the value of the pixel $x_{(i,j)}$ is calculated as $\bar{x}_{(i,j)} - 1$. Otherwise, the value of the pixel $x_{(i,j)}$ is equal to $\bar{x}_{(i,j)}$.

In this specific case, E' is equal to 2, so the value of the extracted bit, $\bar{b}_{(i,j)}$, is 1. The value of the pixel $x_{(i,j)}$ is calculated as $\bar{x}_{(i,j)} - 1 = 14 - 1 = 13$, so the pixel value of Imsb is changed from 14 to 13. Finally, by using the concatenation parameter $K = 2$, the values of Imsb and Ilsb are combined to obtain the resulting pixel values, which are (54, 51, 50, 50, 55, 48, 51, 48, and 50).

Now, we convert the grey to binary using reverse preprocessing, and the updated value will be (36,34,35,35,37,32,34,32, and 35).

After decryption of the image block with the help of key k2,the outcome of the converted block (36,34,35,35,37,32,34,32, and 35) undergoes an XOR operation with the k2, where k2 is 70 for Block (a). Similarly, another k2 key will be used for another Block (b) k2, i.e., 192.

Lastly, we apply reordering, which is the reverse of the reordering of blocks used in the embedding with the help of the key k1<1,2>.

3.5. Blockchain-Based Reversible Data Hiding Scheme

To ensure the integrity and traceability of data in the proposed RDHEI scheme, we employ the latest emerging blockchain technology. The proposed work discusses the security challenges associated with sharing confidential data within the defense sector, particularly among departments such as the army, navy, air force, DRDO, and HAL. Given the high level of trust between these entities, we believe that a consortium blockchain is the optimal technology for providing a secure environment for data sharing. This approach enables multiple departments to share data in a highly secure and transparent manner, while also maintaining data integrity and accountability.

In this paper, we have discussed various consensus algorithms, including proof of work (PoW), proof of stake (PoS), and the Byzantine algorithm. While PoS and PoW are heavyweight algorithms and consume significant time, we propose the use of the Byzantine algorithm for our RDHEI scheme. This is a lightweight algorithm that offers low time consumption compared to the other algorithms. As illustrated in Figure 11, DRDO shares secret data with other parties such as HAL, army, navy, etc., using our proposed RDHEI algorithm. DRDO generates a stego image and verifies the transition of the image. After successful verification, the data are reflected to the other party's local system with the help of a blockchain. If a party wants to access the secret data, it generates a hash value from the received data and matches it with the hash value concatenated with the received stego image. A successful match indicates that the integrity of the secret data has been attained.

New block generation and verification

To create the stego image, we employ the methodologies mentioned earlier, which include encrypting the image, reordering its blocks, and encrypting the secret data, using the 2LES algorithm. Once the stego image is generated, we calculate its hash value using the following equation.

$$H_{current} = Hash(\text{stego\_image}, H_{previous}) \quad (1)$$

Here, *Hash*() is a function based upon SHA-512. After a block is generated, it undergoes a consensus process where the PBZT lightweight code is utilized to verify the block in small time intervals. Once the verification process is completed, the newly generated block is added to the blockchain system.

Test the Integrity

To verify the integrity of a received stego image, we consider the following equation:

$$H'_{current} = Hash\left(\text{stego\_image}, H'_{previous}\right) \quad (2)$$

Here, $H_{current}$ is obtained by concatenating the hash value of the previous stego image with the received stego image, i.e., $H_{current}$ = Hash (stego image, $H_{previous}$). Next, we compare $H_{current}$ with the received $H'_{previous}$. If the comparison results in a match, it indicates that the block is intact and the stego image has not been tampered with. However, if the computed results do not match, it signifies that the stego image has been altered.

Security Analysis

In this section, we will discuss the security analysis of the proposed algorithm in detail to ensure data security. Firstly, in the encryption phase, the cover image is encrypted using a key "k1", which provides security by preventing unauthorized access to the original

content. The second step involves the reordering of blocks using key "k2", which ensures that the original ordering of the blocks cannot be obtained without the key "k2". The third step utilizes key "k3" to encrypt the secret data, thus enhancing its security. Additionally, the fourth step involves preprocessing, where the binary code of the secret data is converted to gray code, thereby further increasing the security of the algorithm.

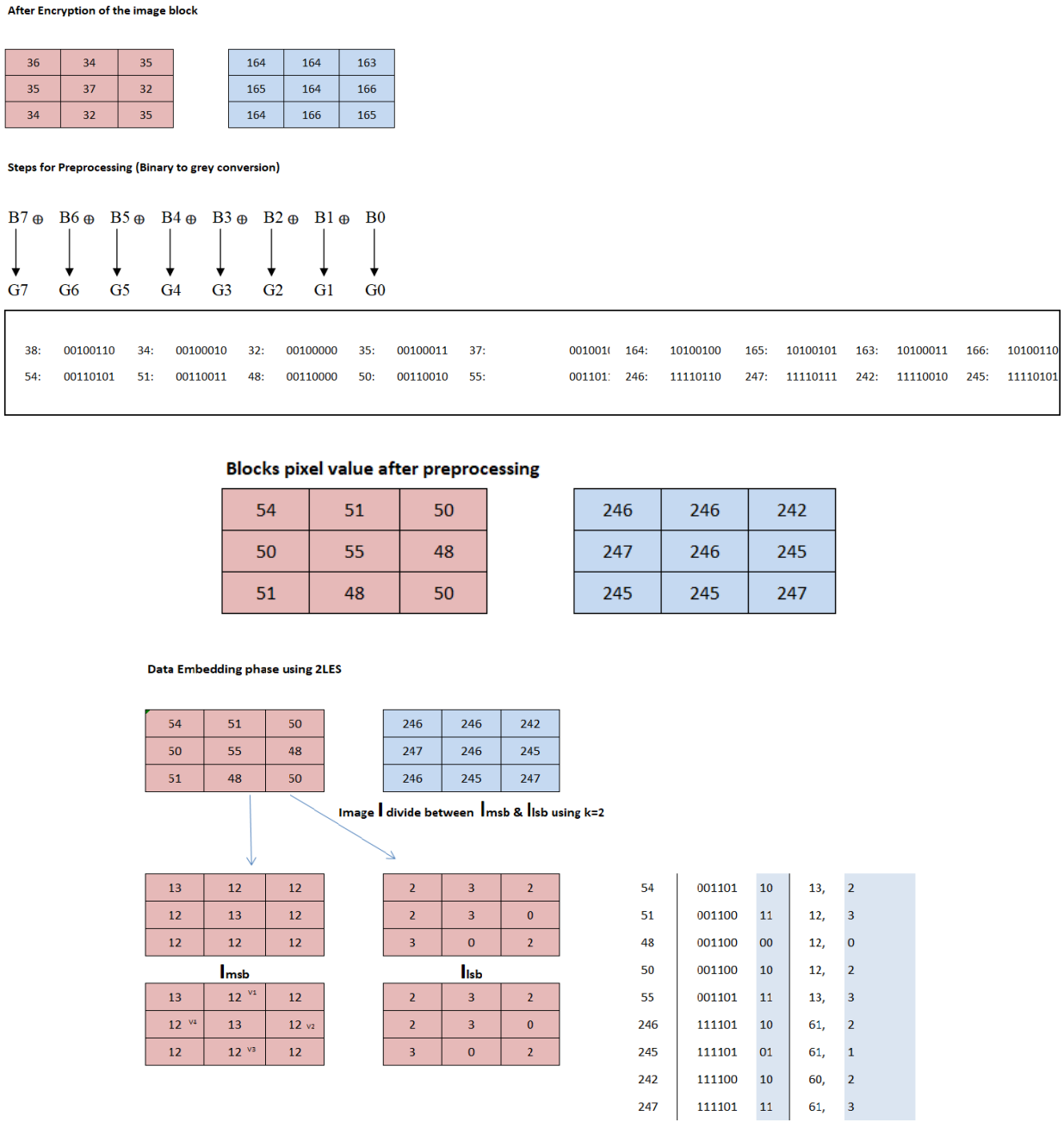


Figure 11. Illustration of the proposed work.

Furthermore, the proposed algorithm employs the 2LES approach to increase the embedding capacity of the cover image. During the embedding phase, the encrypted cover image (I) is split into two parts, Imsb and Ilsb, using key "k". Without key "k", the original image cannot be merged. The blockchain concept is also utilized to enhance the security of the proposed algorithm. The hash code is one of the strongest concepts for securing data, and due to the avalanche effect of the hash code, the proposed algorithm achieves enhanced security. The blockchain helps to secure the content from intruders by requiring

the hash value of the previous block to access the current data block. This process continues further, creating a chain, and the complex nature of the blockchain mechanism ensures a highly secure model.

However, when computing the hash value of the previous block while using the initial block, a new methodology is employed. As there is no previous block in this case, the blockchain mechanism uses a random value as the hash value of the previous block while accessing the initial block. This random hash value helps achieve high security. Moreover, two critical parameters, integrity and traceability, can be achieved through the use of the blockchain mechanism. By utilizing the hash value through the blockchain mechanism, both integrity and traceability can be achieved in the proposed model.

4. Results and Discussions

This section presents the experimental results of our proposed RHDEI and its application to medical images, utilizing a PC with Intel® Core™ i7-3770 CPU @ 3.40 GHz and 8 GB RAM running MATLAB R2017a with the Windows 10 Professional operating system. In this work, we have used two matrices, namely embedding capacity and PSNR (peak signal-to-noise ratio). Embedding capacity refers to the maximum amount of data or information that can be hidden or stored within a digital medium or carrier. PSNR (peak signal-to-noise ratio) is a metric used to measure the quality of a reconstructed or compressed signal by comparing it to the original, uncompressed signal. It quantifies the ratio between the maximum possible power of a signal (the peak signal) and the power of the noise or distortion present in the signal. We use six standard test images, including "Airplane", "Baboon", "Boat", "Lena", "Peppers", and "Sailboat", to demonstrate the effectiveness of our approach, with the encrypted images shown in Figure 12. Figures 13 and 14 show the preprocessed and postprocessed images with embedded data, respectively, while the histograms of the test images and their corresponding encrypted images are given in Figures 15 and 16. The histogram of the encrypted images is evenly distributed over the entire range of gray levels, indicating a high security level of our proposed image encryption scheme. Figure 17 shows the histogram after preprocessing the embedding phase, which is more concentrated for smooth images and more distributed for complex images such as "Baboon". Notably, the histogram band of (128, 192) gray levels is very "clear", with some pixels distributed in the band, as shown in Figure 18. To achieve complete reversibility, the metadata includes three data segments, with the first segment recording the threshold gth with six bits, the second segment recording the number of pixels in the vacating band with twelve bits, and the last segment containing all details of each pixel in the vacating band by one byte for its gray level and two bytes for its coordinates in the image. The total length of metadata is $6 + 12 + (3 \times 8 \times Np)$, where Np is the number of pixels in the vacating band. After histogram shifting and data embedding, the histogram in Figure 17 changes to the distribution shown in Figure 19, where the applied threshold of the vacating band is $gth = 64$. Figure 20 shows the resulting histogram of the marked encrypted image after postprocessing, which is still evenly distributed and preserves a high security level. As the total number of pixels in the vacating band is small, we can efficiently recover the histogram by retrieving the first two segments of metadata to obtaingth and determine the remaining length of metadata to be discarded. Thus, the recovery of the vacating band can be skipped, and the process can proceed directly to extract the secret data stream. To assess the visual quality of the approximated image, we use the PSNR index.

13	12	12
12	14	12
12	12	12

2	3	2
2	3	0
3	0	2

Figure 12. Blocks.

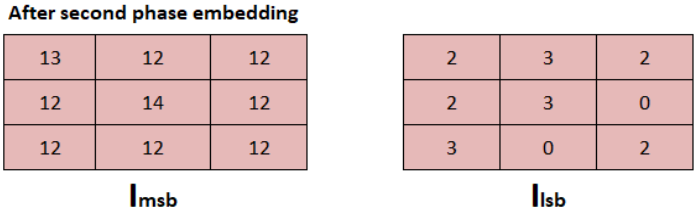


Figure 13. Updated Blocks.

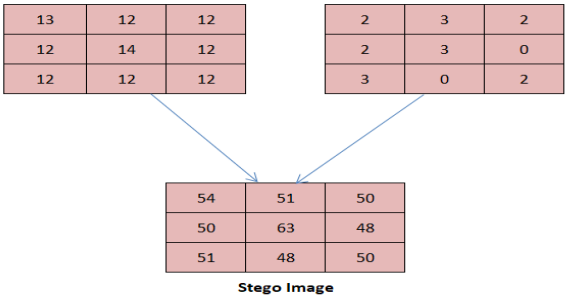


Figure 14. Final Blocks.

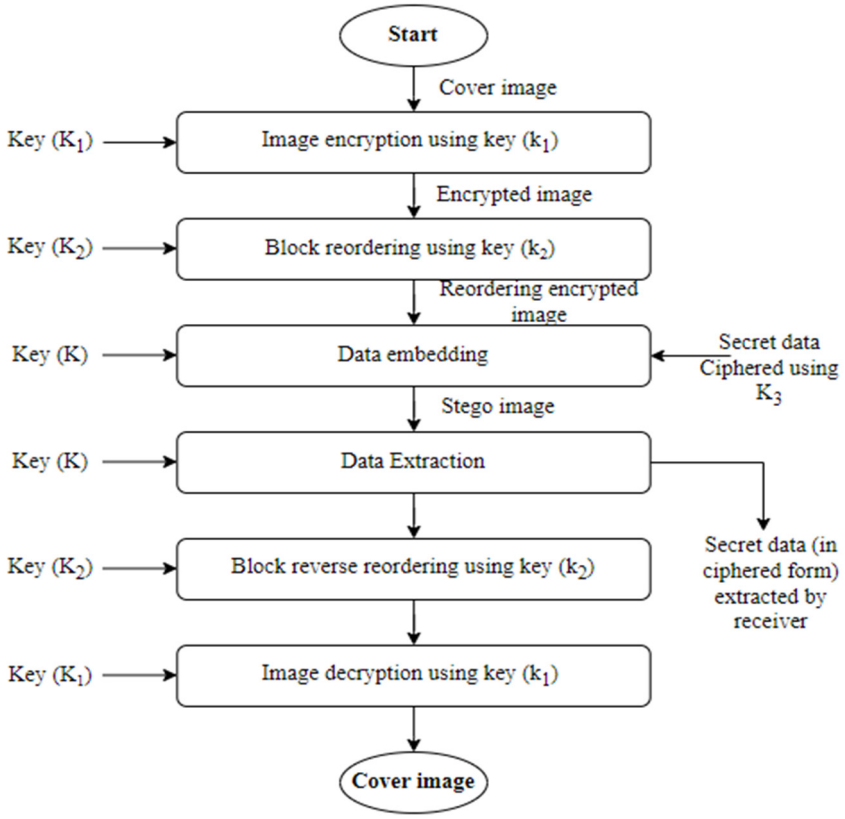


Figure 15. Flowchart of embedding and extraction process.

Furthermore, Figure 21 shows the process of data protection using blockchain efficiently [47]. After that, the Figure 22 showcases a comparative analysis of the proposed approach against other relevant methods, including VRAE-based schemes such as Zhang et al.'s schemes [20] and VRBE-based schemes such as Horng et al.'s scheme [48], Weng et al. [28], Wang et al. [19], and Zhang et al.'s [20] scheme using six commonly used test images, namely "Airplane", "Baboon", "Boat", "Lena", "Peppers", and "Sailboat". All these techniques are entirely reversible, and the PSNR values are compared when some information is discarded during image deciphering. The proposed scheme offers a

high embedding rate and high-quality approximation images, outperforming the other methods. In contrast, the scheme in [19] is VRBE-based and only provides a fixed embedding rate. Our proposed technique offers a flexible threshold (gth) that allows the user to adjust the embedding capacity or PSNR as desired, a feature that is not available in the recently proposed RDH technique by Zhang et al.'s schemes [20]. Furthermore, Zhang et al.'s schemes [20] proposed a homomorphic encryption technique for RDH, and our proposed approach demonstrates comparable performance in terms of ER, PSNR, and flexible threshold configuration. Overall, the experimental results demonstrate that the proposed approach offers superior performance compared to existing techniques while providing greater flexibility in terms of embedding capacity and PSNR.

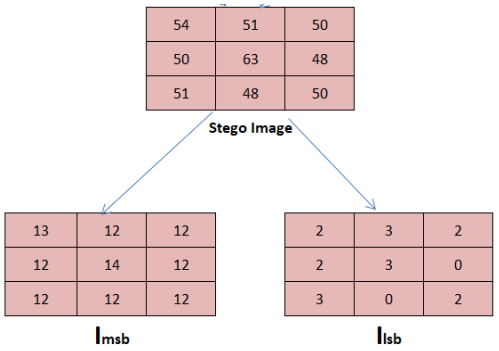


Figure 16. Block Partitioning.

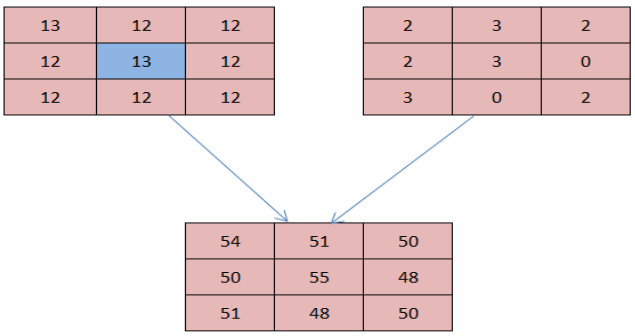


Figure 17. Image merging.

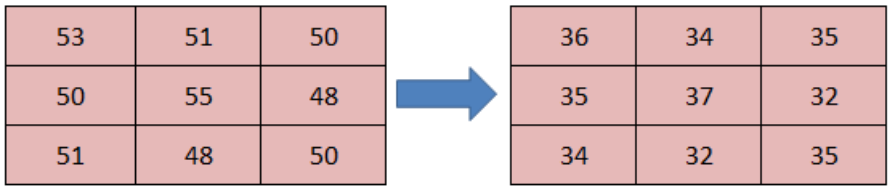


Figure 18. Transformation.

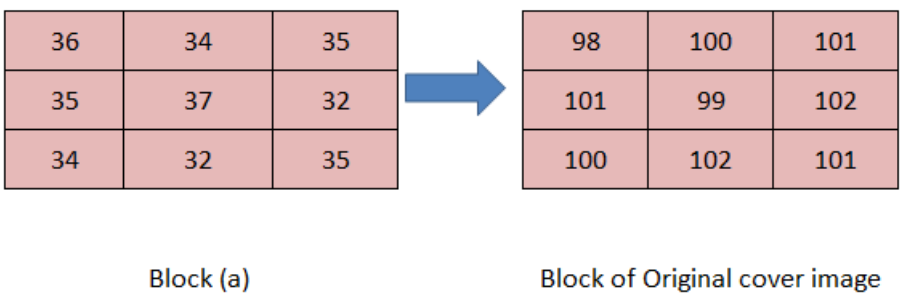


Figure 19. Reverse-reordering.

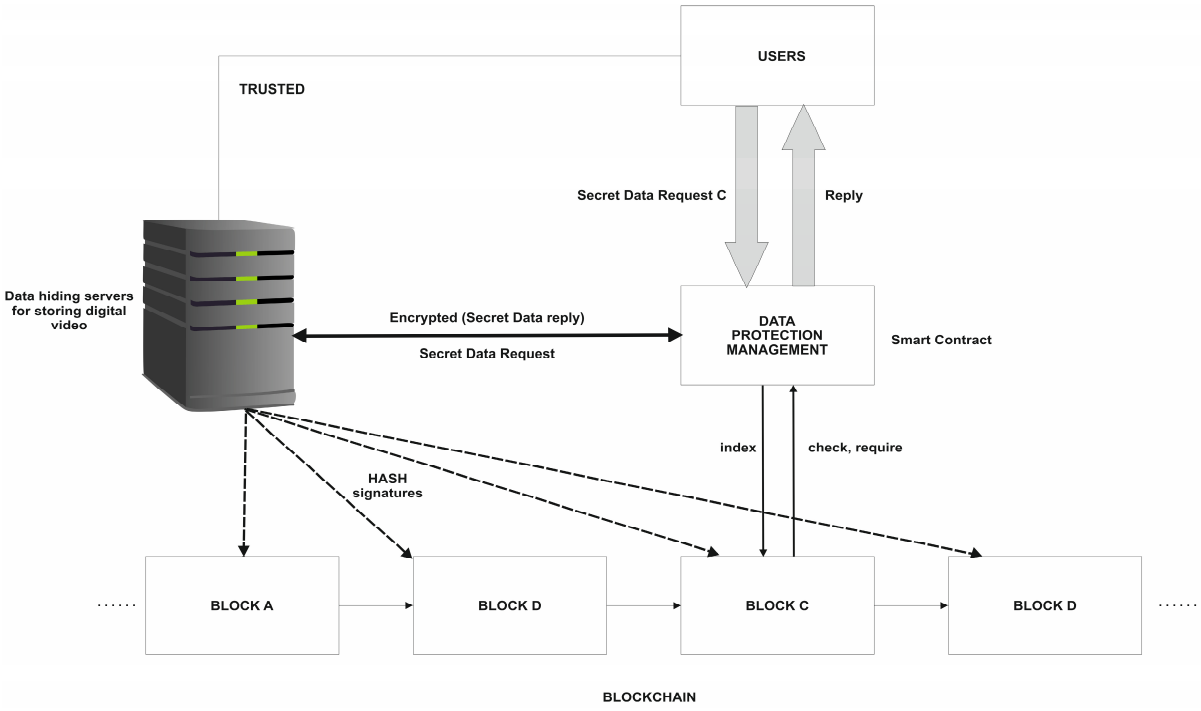


Figure 20. Data concealing in encrypted images using a blockchain-based architecture.

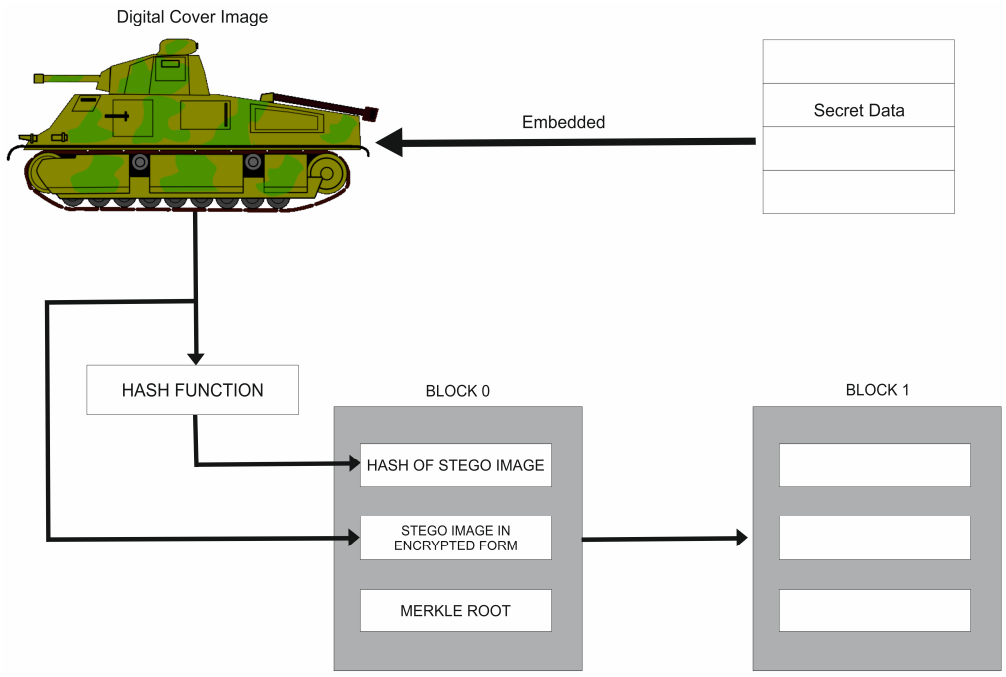
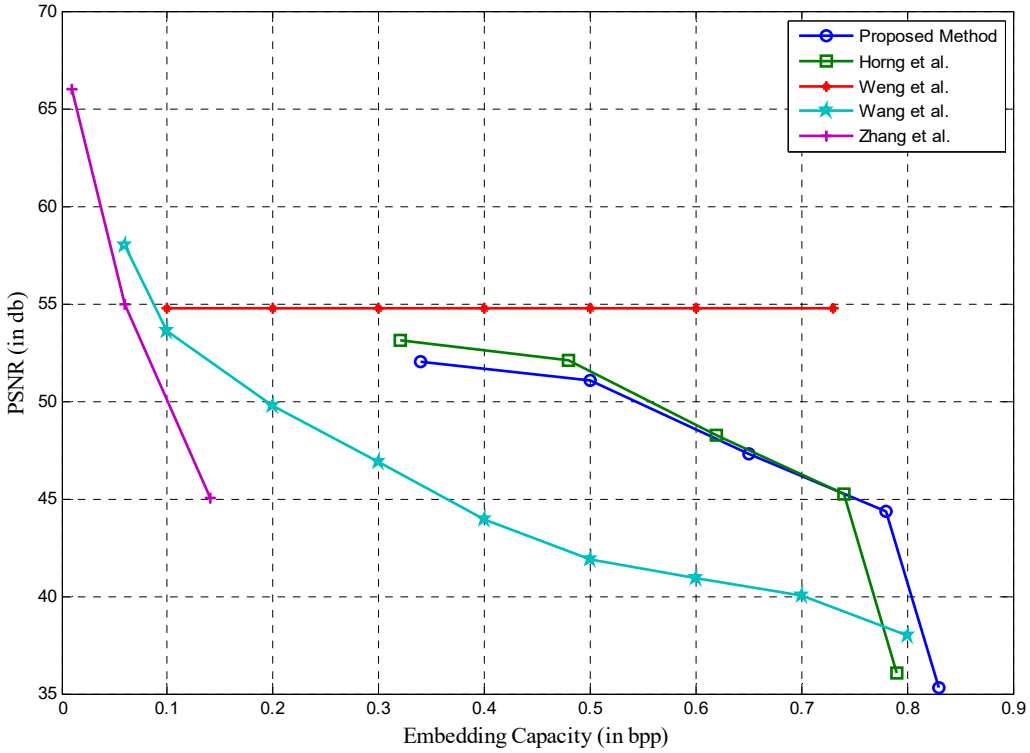
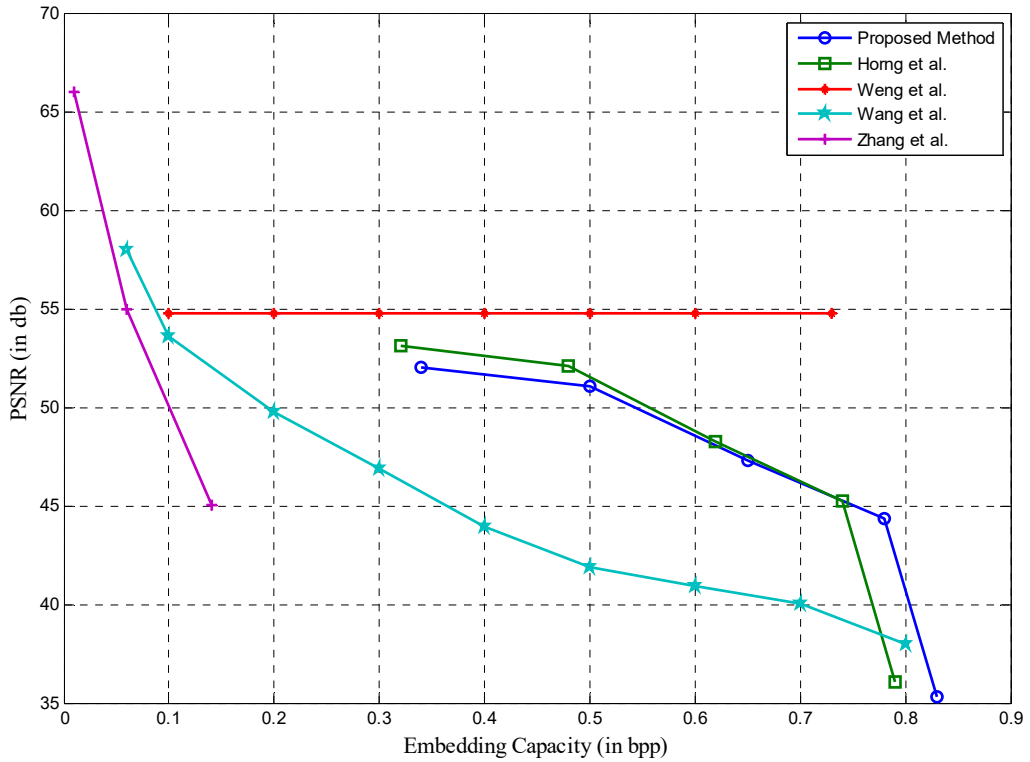


Figure 21. Data protection using blockchain.

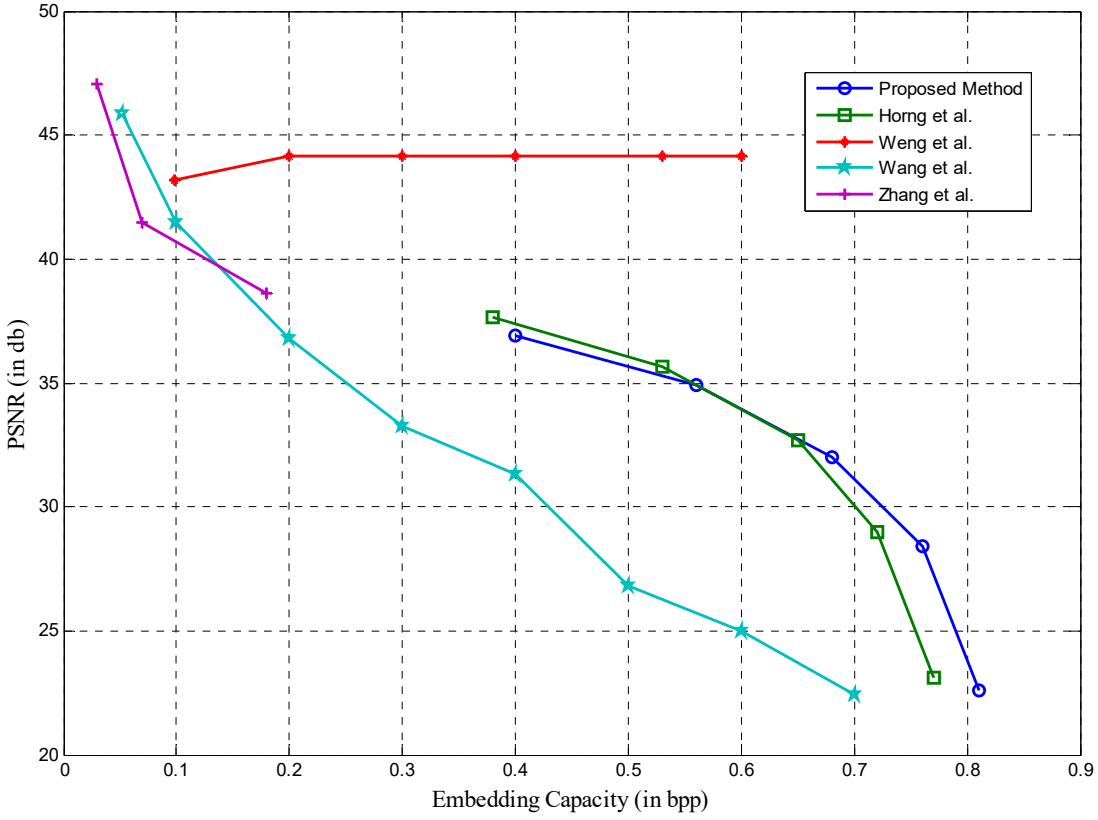


(**a**) **Lena**

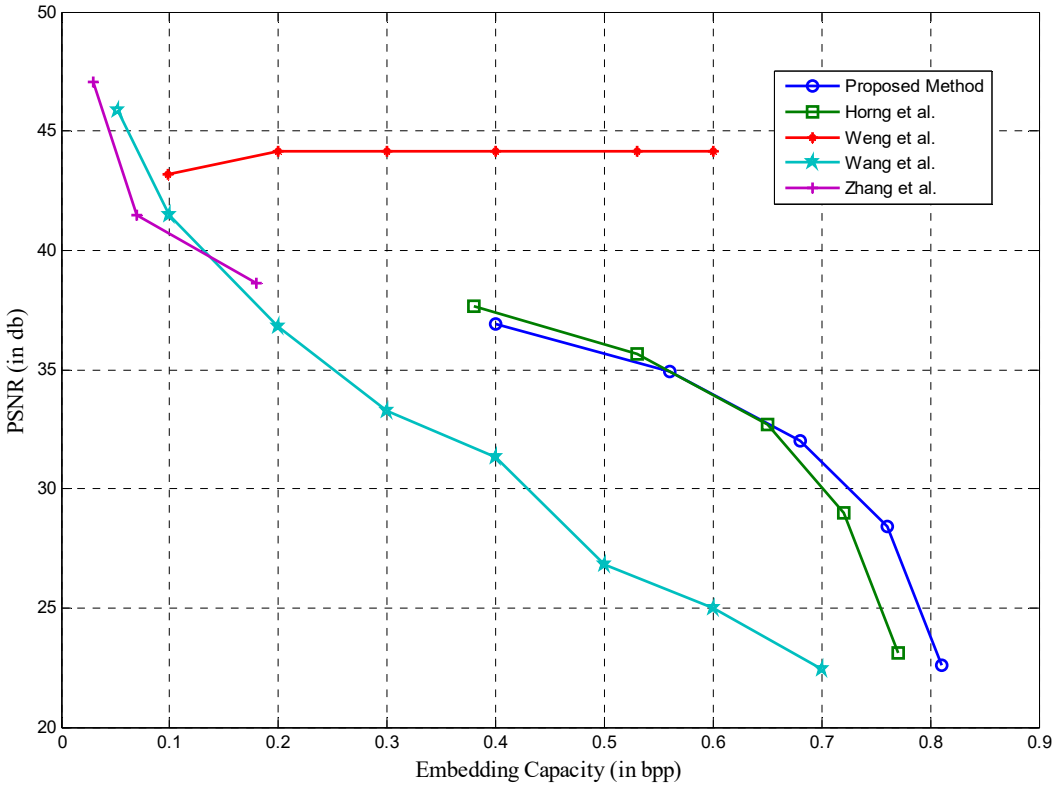


(**b**) **Baboon**

Figure 22. *Cont.*

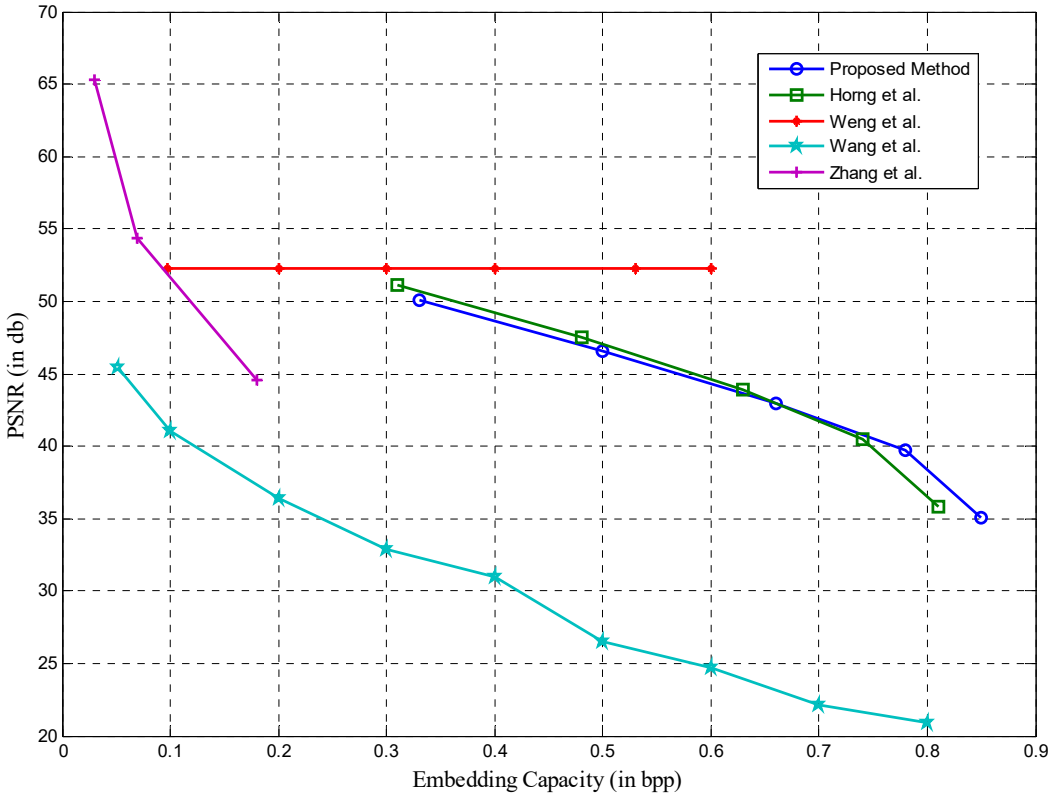


(**c**) **Airplane**

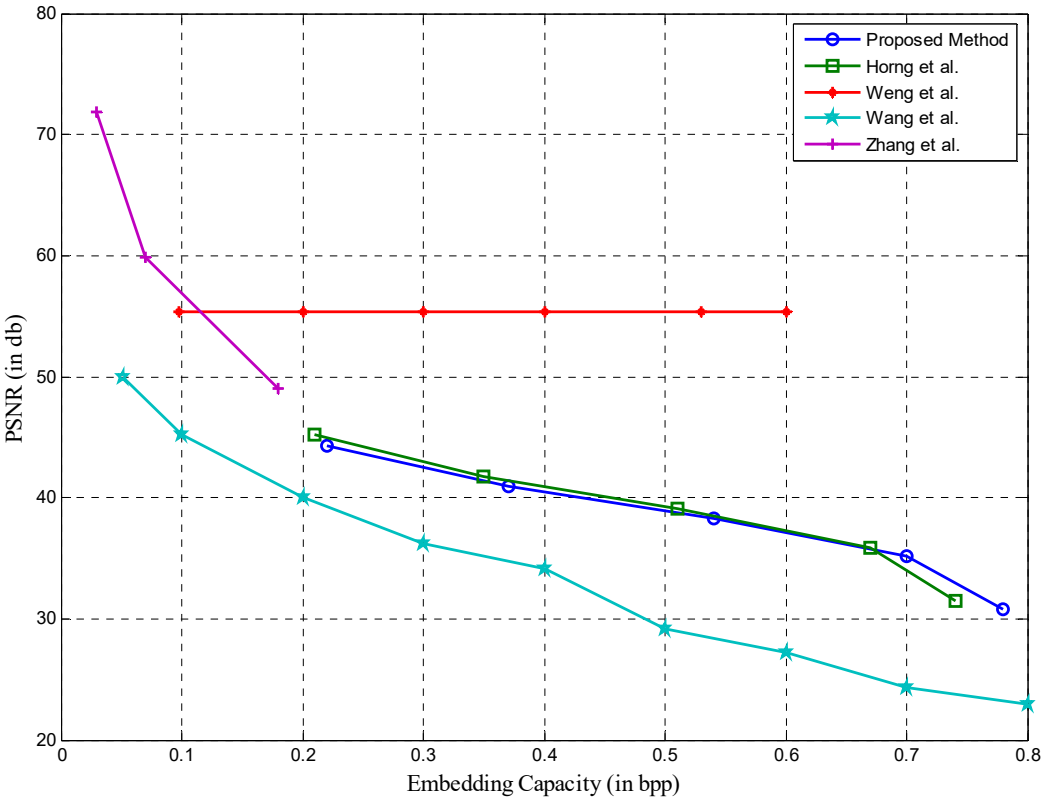


(**d**) **Boat**

Figure 22. *Cont*.



(e) Pepper



(f) Sailboat

Figure 22. (**a**–**f**) PSNR vs. embedding capacity comparison of the proposed method with the state-of the art methods [19,20,28,48].

5. Conclusions

This research paper proposes a novel approach for secure data transmission using a two-layer embedding scheme based on high-capacity reversible data hiding in encrypted image (RDHEI) methodology and blockchain technology. Our approach offers significant advantages over existing methods, including enhanced embedding capacity and improved security through the use of blockchain technology. By concatenating the output generated through RDHEI with its hash value and deploying it through blockchain technology, we ensure the integrity and traceability of data transmission with military cover images in various defense operations. Overall, this research paper presents several innovative contributions in the field of secure data transmission, demonstrating the potential of RDH and blockchain technology for enhancing the security and reliability of data transmission in various applications. The future work directions include optimizing performance, scalability, security analysis, integration with emerging technologies, application expansion, and improving user experience in the proposed approach for secure data transmission using RDHEI and blockchain technology.

Author Contributions: Conceptualization, H.O., S.C. and C.-C.L.; Methodology, A.K.R., H.O., S.C. and C.-C.L.; Software, A.K.R.; Validation, H.O., S.C. and C.-C.L.; Investigation, H.O.; Writing—original draft, A.K.R.; Writing—review & editing, H.O., S.C. and C.-C.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research issupported by funding from NSTC, Taiwan, grants no. 111-2410-H-167 -005 -MY2.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Ni, Z.; Shi, Y.Q.; Ansari, N.; Su, W. Reversible data hiding. *IEEE Trans. Circuits Syst. Video Technol.* **2006**, *16*, 354–362.
2. Du, Y.; Yin, Z.; Zhang, X. Improved lossless data hiding for jpeg images based on histogram modification. *Comput. Mater. Contin.* **2018**, *55*, 495–507.
3. Kumar, M.; Agrawal, S. Reversible data hiding based on prediction error expansion using adjacent pixels. *Secur. Commun. Netw.* **2016**, *9*, 3703–3712. [CrossRef]
4. Rad, R.M.; Wong, K.; Guo, J.-M. Reversible data hiding by adaptive group modification on histogram of prediction errors. *Signal Process.* **2016**, *125*, 315–328. [CrossRef]
5. Fridrich, J.; Goljan, M.; Du, R. Lossless data embedding—New paradigm in digital watermarking. *URASIP J. Appl. Signal Process.* **2002**, *2*, 185–196. [CrossRef]
6. Celik, M.; Sharma, G.; Tekalp, A.; Saber, E. Lossless generalized-LSB data embedding. *IEEE Trans. Image Process.* **2005**, *14*, 253–266. [CrossRef] [PubMed]
7. Honsinger, C.; Jones, P.; Rabbani, M. Lossless Recovery of an Original Image Containing Embedded Data. U.S. Patent #6278791, August 2001.
8. Malik, A.; Singh, S.; Kumar, R. Recovery based high capacity reversible data hiding scheme using even-odd embedding. *Multimed. Tools Appl.* **2018**, *77*, 15803–15827. [CrossRef]
9. Kumar, R.; Chand, S. A reversible high capacity data hiding scheme using pixel value adjusting feature. *Multimed. Tools Appl.* **2016**, *75*, 241–259. [CrossRef]
10. Tian, J. Reversible data embedding using a difference expansion. *IEEE Trans. Circuits Syst. Video Technol.* **2003**, *13*, 890–896. [CrossRef]
11. Alattar, A. Reversible Watermark Using the Difference Expansion of a Generalized Integer Transform. *IEEE Trans. Image Process.* **2004**, *13*, 1147–1156. [CrossRef]
12. Thodi, D.M.; Rodriguez, J.J. Expansion Embedding Techniques for Reversible Watermarking. *IEEE Trans. Image Process.* **2007**, *16*, 721–730. [CrossRef]
13. Wu, X.; Memon, N. Context-based, adaptive, lossless image coding. *IEEE Trans. Commun.* **1997**, *45*, 437–444. [CrossRef]
14. Kamstra, L.H.J.; Heijmans, A.M. Reversible data embedding into images using wavelet techniques and sorting. *IEEE Trans. Image Process.* **2005**, *14*, 2082–2090. [CrossRef]

15. Ou, B.; Li, X.; Zhao, Y.; Ni, R.; Shi, Y.-Q. Pairwise Prediction-Error Expansion for Efficient Reversible Data Hiding. *IEEE Trans. Image Process.* **2013**, *22*, 5010–5021. [CrossRef]
16. Ou, B.; Li, X.; Zhang, W.; Zhao, Y. Improving Pairwise PEE via Hybrid-Dimensional Histogram Generation and Adaptive Mapping Selection. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 2176–2190. [CrossRef]
17. Ni, Z.; Shi, Y.Q.; Ansari, N.; Su, W.; Sun, Q.; Lin, X. Robust lossless image data hiding. *IEEE Int. Conf. Multimed. Expo* **2005**, *3*, 2199–2202. [CrossRef]
18. Ni, Z.; Shi, Y.Q.; Ansari, N.; Su, W.; Sun, Q.; Lin, X. Robust Lossless Image Data Hiding Designed for Semi-Fragile Image Authentication. *IEEE Trans. Circuits Syst. Video Technol.* **2008**, *18*, 497–509. [CrossRef]
19. Wang, W.; Ye, J.; Wang, T.; Wang, W. Reversible data hiding scheme based on significant-bit-difference expansion. *IET Image Process.* **2017**, *11*, 1002–1014. [CrossRef]
20. Zhang, R.; Lu, C.; Liu, J. A high capacity reversible data hiding scheme for encrypted covers based on histogram shifting. *J. Inf. Secur. Appl.* **2019**, *47*, 199–207. [CrossRef]
21. Zhang, Y.X. Separable reversible data hiding in encrypted image. *IEEE Trans. Inf. Secur.* **2012**, *7*, 826–832. [CrossRef]
22. Ma, K.; Zhang, W.; Zhao, X.; Yu, N.; Li, F. Reversible data hiding in encrypted images by reserving room before encryption. *IEEE Trans. Inf. Secur.* **2013**, *8*, 553–562. [CrossRef]
23. Cao, X.; Du, L.; Wei, X.; Meng, D.; Guo, X. High Capacity Reversible Data Hiding in Encrypted Images by Patch-Level Sparse Representation. *IEEE Trans. Cybern.* **2015**, *46*, 1132–1143. [CrossRef] [PubMed]
24. Malik, X.A.; Wang, H.-X.; Chen, Y.; Khan, A.N. A reversible data hiding in encrypted image based on prediction- error estimation and location map. *Multimed. Tools Appl.* **2020**, *79*, 11591–11614. [CrossRef]
25. Wang, W.; Ye, J.; Wang, T.; Wang, W. A high capacity reversible data hiding scheme based on right-left shift. *Signal Process.* **2018**, *150*, 102–115. [CrossRef]
26. Li, J.-J.; Lee, C.-F.; Chang, C.-C.; Lin, J.-Y.; Wu, Y.-H. Reversible Data Hiding Scheme Based on Quad-Tree and Pixel Value Ordering. *IEEE Access* **2019**, *7*, 142947–142962. [CrossRef]
27. Aziz, F.; Ahmad, T.; Malik, A.H.; Uddin, M.I.; Ahmad, S.; Sharaf, M. Reversible data hiding techniques with high message embedding capacity in images. *PLoS ONE* **2020**, *15*, e0231602. [CrossRef]
28. Weng, S.; Pan, J.-S.; Li, L. Reversible data hiding based on an adaptive pixel-embedding strategy and two-layer embedding. *Inf. Sci.* **2016**, *369*, 144–159. [CrossRef]
29. Weng, S.; Liu, Y.; Pan, J.-S.; Cai, N. Reversible data hiding based on flexible block-partition and adaptive block-modification strategy. *J. Vis. Commun. Image Represent.* **2016**, *41*, 185–199. [CrossRef]
30. Puech, W.; Chaumont, M.; Strauss, O. A reversible data hiding method for encrypted images. In Proceedings of the Security, Forensics, Steganography, and Watermarking of Multimedia Contents X, San Jose, CA, USA, 28–30 January 2008; SPIE: Bellingham, WA, USA, 2008; Volume 6819, p. 68191E.
31. SGao; Yu, T.; Zhu, J.; Cai, W. T-PBFT: An EigenTrust-based practical Byzantine fault tolerance consensus algorithm. *China Commun.* **2019**, *16*, 111–123. [CrossRef]
32. Nakamoto, S. Bitcoin: A peer-to-peer electronic cash system. *Decentralized Bus. Rev.* **2008**, 21260.
33. Atzei, N.; Bartoletti, M.; Cimoli, T. A Survey of Attacks on Ethereum Smart Contracts (SoK). In *International Conference on Principles of Security and Trust*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 164–186.
34. Ha, P.H.; Tsigas, P.; Anshus, O.J.; Sname, F. SocioNet: A social-based multimedia access system for unstructured P2P networks. *IEEE Trans. Parallel Distrib. Syst.* **2010**, *21*, 1027–1041.
35. Liu, A.D.; Du, X.H.; Wang, N.; Li, S.Z. Research progress of blockchain technology and its application in information security. *J. Softw.* **2018**, *29*, 2092–2115.
36. Kosba, A.; Miller, A.; Shi, E.; Wen, Z.; Papamanthou, C. Hawk: The Blockchain Model of Cryptography and Privacy-Preserving Smart Contracts. *IEEE Symp. Secur. Priv.* **2016**, 839–858. [CrossRef]
37. Monero: A Note on Chain Reactions in Traceability in Cryptonote 2.0 (2017). Available online: https://lab.getmonero.org/pubs/MRL-0001.pdf (accessed on 12 February 2023).
38. Counsell, C.; Dennis, M.; McDowall, M. Predicting functional outcome in acute stroke: Comparison of a simple six variable model with other predictive systems and informal clinical prediction. *J. Neurol. Neurosurg. Psychiatry* **2004**, *75*, 401–405. [CrossRef]
39. Neisse, R.; Steri, G.; Nai-Fovino, I. A Blockchain-based Approach for Data Accountability and Provenance Tracking. In Proceedings of the 12th International Conference on Availability, Reliability and Security, Benevento, Italy, 29 August–1 September 2017; Volume 14, pp. 1–10. [CrossRef]
40. Kumar, R.; Malik, A. Multimedia information hiding method for AMBTC compressed images using LSB substitution technique. *Multimed. Tools Appl.* **2022**, *82*, 8623–8642. [CrossRef]
41. Kumar, N.; Kumar, R.; Malik, A.; Singh, S.; Jung, K.-H. Reversible data hiding with high visual quality using pairwise PVO and PEE. *Multimed. Tools Appl.* **2023**, 1–26. [CrossRef]
42. Zhao, K.; Hu, J.; Shao, H.; Hu, J. Federated multi-source domain adversarial adaptation framework for machinery fault diagnosis with data privacy. *Reliab. Eng. Syst. Saf.* **2023**, *236*, 109246. [CrossRef]
43. Zhao, K.; Jia, F.; Shao, H. A novel conditional weighting transfer Wasserstein auto-encoder for rolling bearing fault diagnosis with multi-source domains. *Knowl.-Based Syst.* **2023**, *262*, 110203. [CrossRef]

44. Brahma, S.R.; Singh, S.; Gupta, D.K.; Malik, A. A reversible data hiding technique using lower magnitude error channel pair selection. *Multimed. Tools Appl.* **2022**, 1–22. [CrossRef]
45. Kaur, G.; Singh, S.; Rani, R.; Kumar, R. A Comprehensive Study of Reversible Data Hiding (RDH) Schemes Based on Pixel Value Ordering (PVO). *Arch. Comput. Methods Eng.* **2020**, *28*, 3517–3568. [CrossRef]
46. Kaur, G.; Singh, S.; Rani, R.; Kumar, R.; Malik, A. High-quality reversible data hiding scheme using sorting and enhanced pairwise PEE. *IET Image Process.* **2021**, *16*, 1096–1110. [CrossRef]
47. Kumar, N.; Kumar, R.; Malik, A.; Singh, S. Low bandwidth data hiding for multimedia systems based on bit redundancy. *Multimed. Tools Appl.* **2021**, *81*, 35027–35045. [CrossRef]
48. Horng, J.-H.; Chang, C.-C.; Li, G.-L.; Lee, W.-K.; Hwang, S.O. Blockchain-Based Reversible Data Hiding for Securing Medical Images. *J. Health Eng.* **2021**, *2021*, 9943402. [CrossRef] [PubMed]