Reversible Data Hiding: Principles, Techniques, and Recent Studies

Masoud Nosrati * Ronak Karimi Mehdi Hariri
Eslamabad-E-Gharb branch, Eslamabad-E-Gharb branch, Eslamabad-E-Gharb branch,
Islamic Azad University, Islamic Azad University, Islamic Azad University,
Eslamabad-E-Gharb, Iran Eslamabad-E-Gharb, Iran Eslamabad-E-Gharb, Iran
minibigs_m@yahoo.co.uk rk_respina_67@yahoo.com mehdi.hariri@yahoo.com

Abstract: In this paper, we are going to have a survey on one of data hiding techniques called "Reversible Data Hiding (RDH)". Due to this, basic principles and notions of RHD are introduced. Also, primary techniques as the principles of RHD are talked. They are: Pair-Wise Logical Computation (PWLC) data hiding technique and Data Hiding by Template ranking with symmetrical Central pixels (DHTC) technique. Afterwards, some of proposed algorithms in the field of RHD were investigated. They were: Lossless Compression and Encryption of Bit-Planes by Fridrich et al., Reversible Data Hiding at Low Pixel-Levels by Mehmet et al., Based on Integer Wavelet Transform by Xuan et al., High Capacity Watermarking Based on Difference Expansion by Tian, and Reversible Data Hiding by Histogram Shifting by Ni et al. All of nominated algorithms are investigated in details to present a good comparative approach to existing methods.

Key word: Reversible data hiding, data hiding, data security, RHD, PWLC, DHTC

I. INTRODUCTION

Data hiding are a group of techniques used to put a secure data in a host media (like images) with small deterioration in host and the means to extract the secure data afterwards. For example, steganography can be named [1]. Steganography is one such pro-security innovation in which secret data is embedded in a cover [2]. But, this paper will get into reversible data hiding. Reversible data-hiddings insert information bits by modifying the host signal, but enable the exact (lossless) restoration of the original host signal after extracting the embedded information. Sometimes, expressions like distortion-free, invertible, lossless or erasable watermarking are used as synonyms for reversible watermarking [3].

In most applications, the small distortion due to the data embedding is usually tolerable. However, the possibility of recovering the exact original image is a desirable property in many fields, like legal, medical and military imaging. Let us consider that sensitive documents (like bank checks) are scanned, protected with an authentication scheme based on a reversible data hiding, and sent through the Internet. In most cases, the watermarked documents will be sufficient to distinguish unambiguously the contents of the documents. However, if any uncertainty arises, the possibility of recovering the original unmarked document is very interesting [3].

Lossless data embedding techniques may be classified into one of the following two categories: Type-I algorithms [4] employ additive spread spectrum techniques, where a spread spectrum signal corresponding to the information payload is superimposed on the host in the embedding phase. At the decoder, detection of the embedded information is followed by a restoration step where watermark signal is removed, i.e. subtracted, to restore the original host signal. Potential problems associated with the limited range of values in the digital representation of the host signal, e.g. overflows and underflows during addition and subtraction, are prevented by adopting modulo arithmetic. Payload extraction in Type-I algorithms is robust. On the other hand, modulo arithmetic may cause disturbing salt-and-pepper artifacts. In Type II algorithms [5][6], information bits are embedded by modifying, e.g. overwriting, selected features (portions) of the host signal -for instance least significant bits or high frequency wavelet coefficients-. Since the embedding function is inherently irreversible, recovery of the original host is achieved by compressing the original features and transmitting the compressed bit-stream as a part of the embedded payload. At the decoder, the embedded payload- including the compressed bit-stream- is extracted, and original host signal is restored by replacing the modified features with the decompressed original features. In general, Type II algorithms do not cause salt-and-pepper artifacts and can facilitate higher embedding capacities, albeit at the loss of the robustness of the first group [7].
II. BASIC RDH TECHNIQUES

II.1. PWLC data hiding technique [3]

To the best of our knowledge, the only proposed reversible data hiding technique for binary images is PWLC (Pair-Wise Logical Computation) [8][9]. However, it seems that sometimes PWLC does not correctly extract the hidden data, and fails to recover perfectly the original cover image.

PWLC uses neither the spread spectrum nor any compression technique. It uses XOR binary operations to store the payload in the host image. It scans the host image in some order (for example, in raster scanning order). Only sequences “000000” or “111111” that are located near to the image boundaries are chosen to hide data. The sequence “000000” becomes “001000” if bit 0 is inserted, and becomes “001100” if bit 1 is inserted. Similarly, the sequence “111111” becomes “110111” if bit 0 is inserted, and becomes “110011” if bit 1 is inserted. However, the papers [8][9] do not describe clearly how to identify the modified pixels in the extraction process. The image boundaries may change with the watermark insertion. Moreover, let us suppose that a sequence “001000” (located near to an image boundary) was found in the stego image. The papers do not describe how to discriminate between an unmarked “001000” sequence and an originally “000000” sequence that became “001000” with the insertion of the hidden bit 0 [3].

II.2. DHTC data hiding technique [3]

The technique proposed in this paper (RDTC) is based on the non-reversible data hiding named DHTC (Data Hiding by Template ranking with symmetrical Central pixels) [10]. DHTC flips only low-visibility pixels to insert the hidden data and consequently images marked by DHTC have excellent visual quality and do not present salt-and-pepper noise.

```
```

Figure 1: A 3×3 template ranking with symmetrical central pixels in increasing visual impact order.

Hatched pixels match either black or white pixels (note that all central pixels are hatched). The score of a given pattern is that of the matching template with the lowest impact. Mirrors, rotations and reverses of each pattern have the same score [3].

III. INVESTIGATION OF SOME RDH ALGORITHMS

In this section reversible data hiding algorithms proposed so far in the lossless watermarking literature are presented. Abstraction and investigation of features was done by Mohammad Awrangjeb [11] in a separate paper. We use the abstracts in this section.

III.1. Lossless Compression and Encryption of Bit-Planes

Fridrich et al. propose this algorithm in [12]. Space to hide data is found by compressing proper bit-plane that offers minimum redundancy to hold the hash (authentication information). Lowest bit-plane offering lossless compression can be used unless the image is not noisy. In completely noisy image some bit-planes exhibit strong correlation. These bit-planes can be used to find enough room to store the hash. Hash length is generally 128 bit using MD5 algorithm [13]. The algorithm starts lossless compression from 5th bit-plane and calculates redundancy by subtracting
compressed data size from number of pixels. The authors use the JBIG lossless compression method [14] to compress the bit-planes. During embedding the algorithm first calculates the hash of the original image, finds the proper bit-plane, and adds the hash with the compressed bit-plane data. Then it replaces selected bit-plane by concatenated data. For more security the concatenated hash with compressed data is encrypted using symmetric key encryption based on 2-dimensional chaotic maps [15]. This algorithm takes variable sized blocks and gives the encrypted message as long as the original message, so no padding is needed. Other public or symmetric key algorithms can be used, but they require padding to embed the encrypted message and hence increase distortion. During decoding after key bit-plane selection the data is decrypted and hash is separated from the compressed original bit-plane data. The bit-plane is replaced by the decompressed data; hence the exact copy of the original image is found. The hash of the reconstructed image is calculated and compared with the extracted hash; if both are same the image in question is authentic [11].

The advantages of this algorithm are – (i) high capacity, (ii) security is equivalent to the security provided by cryptographic authentication, and (iii) can be applied for the authentication purposes of JPEG files, complex multimedia objects, audio files, digitized hologram, etc. The disadvantages are – (i) noisy image forces the algorithm to embed information in higher bit-plane when the distortions are higher and easily visible, (ii) single bit-plane in a small image does not offer enough space to hide hash after compression, so two or more bit-planes are required and the artifacts must be visible, and (iii) capacity is not high enough to embed large payload [11].

III.2. Reversible Data Hiding at Low Pixel-Levels

Mehmet et al. [16] propose a reversible data hiding technique that uses prediction based conditional entropy coder utilizing static portions of the input signal as side-information to improve the compression efficiency. Hence the lossless data embedding capacity is increased. This spatial domain method is the modification of generalized LSB embedding technique and uses very simple signal features: lowest levels of raw pixels. It follows the general principal [17] of lossless embedding. The algorithm searches the whole image to have the first L (say, L = 4) lowest levels of pixel values. It compresses these pixel values using CALIC lossless image compression algorithms [18] and check whether it gives enough space (128-bit for hash). If the given capacity is lower than expectation the algorithm increases L and continues searching. Once it finds enough capacity, it concatenates the hash with compressed pixel values. The concatenated bit-string is converted into L-ary symbols to replace the lowest L-levels of pixel values. The decoding process is just the reverse of the embedding phase. The advantages are – (i) simple algorithm, and (ii) higher capacity can be found with the increase of embedding level L. The disadvantages are – (i) capacity depends on image structure, smooth images give higher capacity than irregular textured images, and (ii) artifacts are visible with the increase of embedding level L. Though the algorithm gives a very high capacity, it gives incredible distortions to the original image [11].

III.3. Based on Integer Wavelet Transform

In Xuan et al. [19] propose a lossless data hiding having large capacity based on integer wavelet transform. It hides authentication information and bookkeeping data into a middle bit-plane of integer wavelet coefficients in high frequency sub-bands. The histogram modification or integer modulo addition is used to prevent gray scale overflowing during data embedding. The method uses second-generation wavelet transform IWT [20].

The authors find more bias between 1s and 0s starting from 2nd bit-plane to higher bit-planes of IWT coefficients. To make the watermarked image perceptually as same as the original image and to have high PSNR they tell to embed information into middle bit-plane and in the high frequency sub-bands respectively. To compress it they use arithmetic coding from [21]. The watermark payload concatenated with compressed data is embedded with a secret key. In extraction phase, the watermark (say, hash of original image) is extracted and the original image is reconstructed in the opposite manner. To prevent the gray-scale overflow either histogram modification or gray scale modification is used as pre-processing and post-processing during embedding and extraction phases respectively.

The advantages are – (i) high capacity, and (ii) use of secret key during embedding increases security. The disadvantages are – (i) often multiple bit-planes are required to have enough space when the artifacts become visible, and (ii) gray scale mapping [11].
III.4. High Capacity Watermarking Based on Difference Expansion

In [22][23] Tian propose a high quality reversible watermarking method with high capacity based on difference expansion. Pixel differences are used to embed data; this is because of high redundancies among the neighboring pixel values in natural images.

During embedding – (i) differences of neighboring pixel values are calculated, (ii) changeable bits in that differences are determined, (iii) some differences are chosen to be expandable by 1-bit, so changeable bits increases, (iii) concatenated bit-stream of compressed original changeable bits, the location of expanded difference numbers (location map), and the hash of original image (payload) is embedded into the changeable bits of difference numbers in a pseudo random order, (iv) use the inverse transform to have the watermarked pixels from resultant differences. During watermark extraction – (i) differences of neighboring pixel values are calculated, (ii) changeable bits in that differences are determined, (iii) extract the changeable bit-stream ordered by the same pseudo random order as embedding, (iv) separate the compressed original changeable bit-stream, the compressed bit-stream of locations of expanded difference numbers (location map), and the hash of original image (payload) from extracted bit-stream, (v) decompress the compressed separated bit-streams and reconstruct the original image replacing the changeable bits, (vi) calculate the hash of reconstructed image and compare with extracted hash. The advantages are – (i) no loss of data due to compression-decompression, (ii) also applicable to audio and video data, and (iii) encryption of compressed location map and changeable bit-stream of different numbers increases the security. The disadvantages include – (i) there may be some round off errors (division by 2), though very little, (ii) largely depends on the smoothness of natural image; so cannot be applied to textured image where the capacity will be zero or very low, and (iii) there is significant degradation of visual quality due to bit-replacements of gray scale pixels [11].

III.5. Reversible Data Hiding by Histogram Shifting

Ni et al. [24] utilizes zero or minimum point of histogram. If the peak is lower than the zero or minimum point in the histogram, it increases pixel values by one from higher than the peak to lower than the zero or minimum point in the histogram. While embedding, the whole image is searched. Once a peak-pixel value is encountered, if the bit to be embedded is ‘1’ the pixel is added by 1, else it is kept intact. Alternatively, if the peak is higher than the zero or minimum point in the histogram, the algorithm decreases pixel values by one from lower than the peak to higher than the zero or minimum point in the histogram, and to embed bit ‘1’ the encountered peak-pixel value is subtracted by 1. The decoding process is quiet simple and opposite of the embedding process. The algorithm essentially does not follow the general principle of lossless watermarking in [17].

The advantages of this method are – (i) it is simple, (ii) it always offers a constant PSNR 48.0dB, (iii) distortions are quite invisible, and (iv) capacity is high. The disadvantages are – (i) capacity is limited by the frequency of peak-pixel value in the histogram, and (ii) it searches the image several times, so the algorithm is time consuming.

There are some more efficient algorithms have also been proposed. We refer our survey report of lossless watermarking in [25] to have complete research knowledge of reversible data hiding [11].

IV. Conclusion

In this paper, we got into reversible data hiding principles and techniques. Basic notions of RHD and primary techniques including PWLC data hiding technique and DHTC data hiding technique were talked. Afterwards, some of proposed algorithms in the field of RHD were investigated. They were: Lossless Compression and Encryption of Bit-Planes by Fridrich et al., Reversible Data Hiding at Low Pixel-Levels by Mehmet et al., Based on Integer Wavelet Transform by Xuan et al., High Capacity Watermarking Based on Difference Expansion by Tian, and Reversible Data Hiding by Histogram Shifting by Ni et al. Features of each algorithm was talked separately in details.
REFERENCES


