Multimedia Tools and Applications (2019) 78:32239–32259
https://doi.org/10.1007/s11042-019-07997-0

A systematic survey on block truncation coding based data hiding techniques

Rajeev Kumar[1,2] · Ki-Hyun Jung[2]

Received: 17 October 2018 / Revised: 7 June 2019 / Accepted: 10 July 2019
Published online: 31 July 2019

Abstract
Block truncation coding is one of the simplest encoding methods which require insignificant computing cost to compress images. Due to the vast demand on embedding data into compressed images with low computing cost, a number of data hiding methods to improve block truncation coding have been proposed to be suitable for the low power devices such as IoT devices, field-programmable gate array, and portable image signal processor. In this paper, block truncation coding based data hiding methods will be discussed and analyzed on two key metrics - data hiding capacity and image quality - as many researchers are focusing to increase the image quality along with data hiding capacity. Here, our aim is to provide guidance to interested researchers for their future works in the field of block truncation coding based data hiding techniques. Finally, future directions of research with some suggestions will be discussed.

Keywords Data hiding · Block truncation coding · Information hiding · BTC · AMBTC · Steganography

1 Introduction

Digital communication over the Internet has become a very popular communication tool. However, the communication message/contents in the form of digital media like as text, image, video, audio confront the security problems. To address the security problem, many cryptography techniques such as DES and RSA have been discussed, where the secret message can be encoded in such a way

✉ Ki-Hyun Jung
khanny.jung@gmail.com

Rajeev Kumar
rajivgarg@outlook.com

[1] IT Research & Development Center, Wookyung Information Technology, Buk-gu, Daegu, 41519 Republic of Korea

[2] Department of Cyber Security, Kyungil University, 50 Gamasil-gil, Hayang-eup, Gyeongsan-si, Gyeongbuk 38428, Republic of Korea

so that its contents are not readable to any unauthorized person. Although cryptography techniques can provide higher security, the visibility of the ciphertext/encoded message may pull attention to attackers. To overcome the visible limitation, steganography also known as data hiding was introduced [33]. The data hiding techniques embed the secret message in a cover media which can be text, image, audio, and video, in such a way that its presence is not detectable to anyone [16, 25, 33, 44]. The data hiding techniques mainly work into three domains: spatial-domain [31, 34, 42, 49], frequency-domain [5, 32, 53], and compression domain [4, 6, 45]. The spatial domain-based data hiding techniques embed the secret data by modifying the pixel values directly. These techniques are the simplest and have the least computational complexity. The most prominent spatial data hiding techniques are least significant bits (LSB) [3] and pixel-value difference (PVD) [52]. The frequency domain-based data hiding techniques first transforms the original image into frequency coefficients, and then the secret data is embedded into the transformed frequency coefficients. The compression domain-based data hiding techniques first compress the original media into compressed codes using any available compression method like image compression such as joint photographic experts group (JPEG) [35], vector quantization (VQ) [4], graphics interchange format (GIF) [1], and block truncation coding (BTC) [15] and then the secret data is embedded into the compressed codes. The secret data concealed in the compressed image is not easily detectable and unauthorized people or attackers will not usually be suspicious on stego-images. The block diagram as given in Fig. 1 shows the current compression domain-based data hiding schemes.

In general, there are two types of compression methods namely lossless and lossy. The lossless compression method like LZW [15, 26], compresses the media in such a way that can be completely restored to its original form whereas the contents of the media are lost when it is compressed by lossy compression methods like block truncation coding (BTC). The lossy compression techniques are quite popular when the media to be dealt with in the form of image or video where the user needs to save bandwidth by reducing the media size significantly as the loss of some of the contents does not have a significant impact on the meaning of the message. The lossy compression methods are massively being used for the sharing of video/images on the internet where the efficient use of bandwidth has higher importance. Therefore, data hiding schemes in the lossy compression domain have seen a significant rise in recent

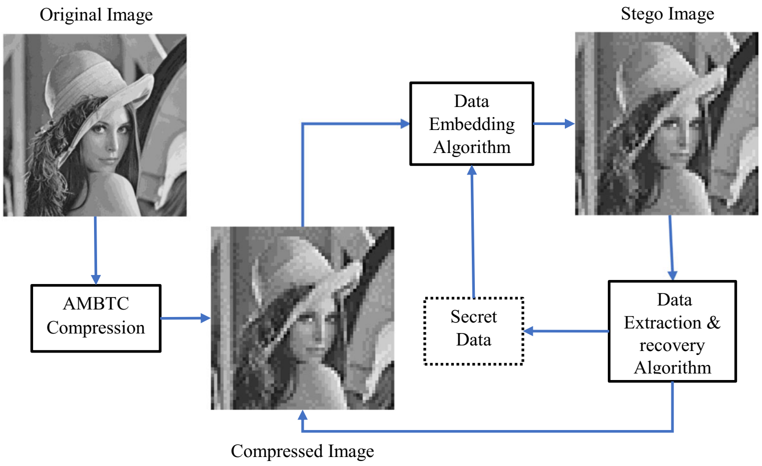


Fig. 1 Block diagram of BTC/AMBTC based data hiding schemes

years. In this paper, the existing BTC compression-based data hiding schemes will be discussed. In the following section, BTC [15] and AMBTC [28] methods are briefly reviewed.

2 Block truncation coding (BTC) encoding method

Block truncation coding (BTC) was proposed by Delp and Mitchell [15]. It is one of the simple image compression algorithms for grayscale images. BTC divides the image into $M \times M$ size blocks to reduce its gray levels. The main idea of the BTC method is to maintain certain statistical moments of each block by having two quantization levels and one-bit-plane per block. To improve the performance further, Lema and Mitchell introduced a new lossy compression method popularly known as AMBTC which stands for Absolute Moment Block Truncation Coding [28]. AMBTC is computationally simpler than the BTC method. The details of the image compression and decompression procedure of the AMBTC method are described as follows.

During the compression procedure, the original image is partitioned into non-overlapping blocks of size $M \times M$ pixels in first. For each block of the image, the mean pixel value AVG is calculated by using Eq. (1)

$$AVG = \frac{\sum_{i=1}^{M\times M} x_i}{M \times M} \tag{1}$$

Where x_i represents the i^{th} pixel value of a block of the image and $M \times M$ represents the size of the image block. Then, a bit-plane (B) is constructed for each block of the image in which every pixel of the block is represented by one bit only, using Eq. (2) as follows:

$$B_i = \begin{cases} 0 & if\ x_i < AVG \\ 1 & if\ x_i \geq AVG \end{cases} \tag{2}$$

The two quantization levels namely low mean values L_j and the high mean value H_j for each block can be derived from Eqs. (3) and (4), respectively.

$$L_j = \sum_{x_i < \mathrm{AVG}_j}^{M\times M} x_i/\beta \tag{3}$$

$$\mathrm{H_j} = \sum_{x_\mathrm{i} \geq AVG_\mathrm{j}}^{M\times M} \mathrm{x_i}/(M \times M - \beta) \tag{4}$$

Where β is the number of pixels that are greater than or equal to AVG. The mean values L_j and H_j are used to reconstruct the compressed image using the bit-plane by replacing '0's by low mean value L_j and '1's by high mean value H_j. To illustrate the working of the AMBTC method, an example is provided in the following subsection.

2.1 AMBTC example

Suppose we have an image block of size 4 × 4 pixels as shown in Fig. 2a. The mean AVG of the block can be calculated using Eq. (1) as $(130 + 92 + 161 + 165 + 133 + 97 + 90 + 170 + 147 + 175 + 140 + 183 + 124 + 194 + 106 + 116)/4*4 = 139$. To construct the bit-plane for the image block, the mean value AVG =139 is then taken as a threshold. The pixels having values greater than or equal to AVG are represented by 1 in the bit-plane; otherwise, '0' is used to represent the pixel value as shown in Fig. 2b. Two quantization levels L_j=111 and H_j =167 are computed by Eqs. (2) and (3), respectively. By using these two quantization levels L_j, H_j and a bit-plane, the compressed image block can be constructed as shown in Fig. 2c where '0' of the bit-plane is replaced by L_j and '1' is replaced by H_j.

3 Literature survey

In this section, we will discuss some BTC/AMBTC based data hiding schemes. In recent years, there has been quite a lot of work in the domain of BTC/AMBTC based data hiding schemes. BTC/AMBTC based data hiding schemes are majorly categorized into five categories, namely histogram shifting, prediction error expansion, type-1, block classification and miscellaneous based methods on their style of working as shown in Fig. 3. Most of the histogram shifting, prediction error expansion, type-1 and miscellaneous data hiding schemes are lossless/reversible data hiding schemes, whereas block classification-based schemes belong to lossy category. The reversible data hiding scheme enables the receiver to restore the marked image to its original compressed form after extraction of the secret data/message, whereas the original compressed image cannot be restored after extraction of the secret data in the case of lossy schemes.

3.1 Histogram shifting based data hiding schemes

In this sub-section, some of the popular AMBTC based data hiding schemes using histogram shifting are discussed. One of the most popular histogram shifting based reversible data hiding scheme for AMBTC encoded images is given by Lo et al. [38]. First, the scheme constructs low-mean and high-mean tables in addition to bit-planes stream. And then constructs a difference table that contains the difference of low mean value from the corresponding high-mean value additionally. The secret data is embedded in any of the two tables (*Low* mean table, *High* mean table, or difference value table) according to the choice of a user by using the histogram shifting strategy of Ni et al. [42]. The embedding process has been shown through an illustrative example for *Low* mean table (as the embedding process is the same for both the tables) in Fig. 4. The scheme preserves the ethos of BTC compression after hiding the secret data. Thus, good stego-image quality and high data hiding capacity are provided. However, the scheme needs to separately send the information of the peak and zero point pairs to the receiver to extract the secret data and recover the compressed image. An illustrative example of a reversible data hiding scheme given by Lo et al. [38] is shown in Fig. 4.

130	92	161	165
133	97	90	170
147	175	140	183
124	194	106	116

(a) Original image block

0	0	1	1
0	0	0	1
1	1	1	1
0	1	0	0

(b) Bit-plane of the image block

111	111	167	167
111	111	111	167
167	167	167	167
111	167	111	111

(c) Reconstructed image block

Fig. 2 An example of AMBTC method

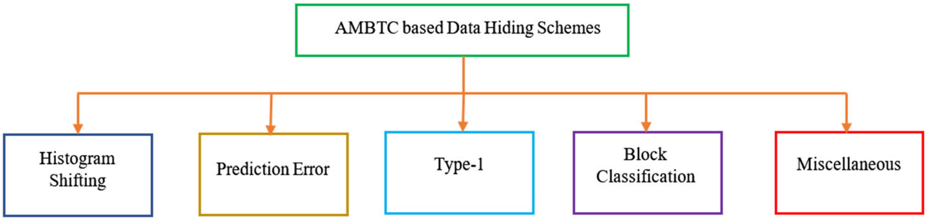
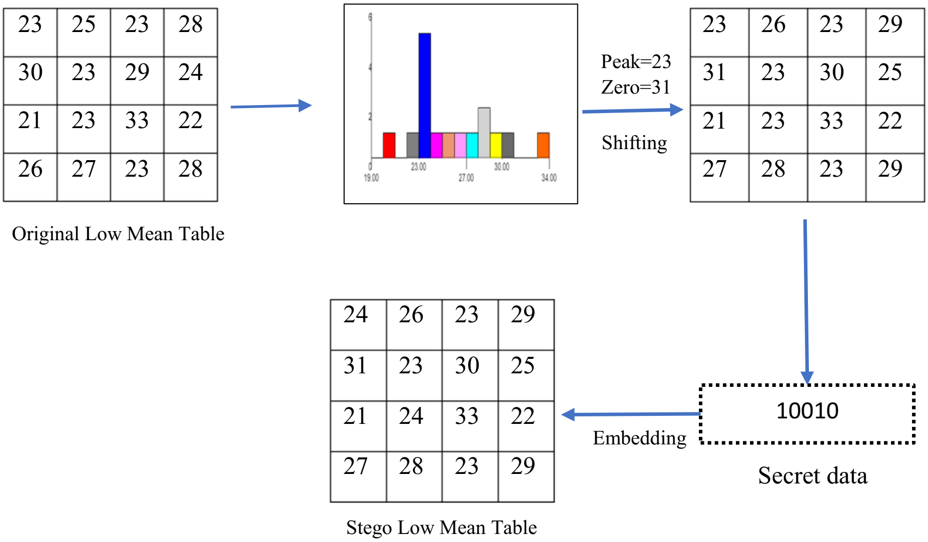


Fig. 3 Taxonomy of the AMBTC based data hiding schemes

Example: Let the low mean values in the low mean table be 23, 25, 23, 28, 30, 23, 29, 24, 21, 23, 33, 22, 26, 27, 23, 28 and secret data is "10010". The scheme constructs a histogram of the low mean table to identify the peak and zero points in first.

In this example, the peak point (most frequent number) is 23 and zero point (least frequent number) is 31 considering the pixel intensity is in the range of 21–33. It has been identified only one peak point and one zero point to make the example simple. Now, the pixel values starting from 24 (peak point + 1) till 30 (zero point-1) are shifted by one to create room for embedding the secret data in the peak point. Thus, low mean values after performing shifting operation are 23, 26, 23, 29, 31, 23, 30, 25, 21, 23, 33, 22, 27, 28, 23, and 29. The scheme [46] embeds the secret data into the peak points either by increasing its value by one or leaving it unchanged. The value is increased when the bit to be embedded is '1'. The first peak point is at first position in the matrix and bit to be embedded is '1', so the value of the peak point is increased by one. Thus, it becomes 24. The next peak point is at third position and bit to be embedded is '0', which means its value will remain unchanged. Following in the same manner, the complete secret data can be embedded into the mean table. Thus, the final values after embedding the secret data are 24, 26, 23, 29, 31, 23, 30, 25, 21, 24, 33, 22, 27, 28, 23, and 29. The extraction process is just reversed for the embedding process.

To overcome the limitation of Lo et al. [38], Chang et al. [7] discussed a new data hiding method that embeds the secret message in the bit-plane data along with quantization levels



23	25	23	28
30	23	29	24
21	23	33	22
26	27	23	28

Original Low Mean Table

23	26	23	29
31	23	30	25
21	23	33	22
27	28	23	29

24	26	23	29
31	23	30	25
21	24	33	22
27	28	23	29

Stego Low Mean Table

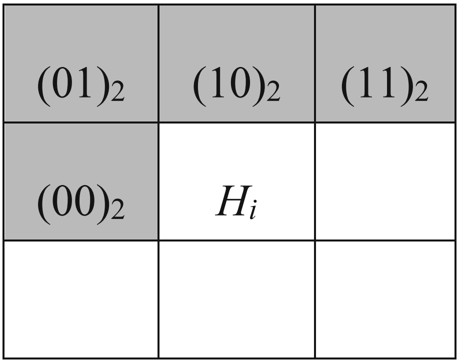
Fig. 4 An example of histogram shifting based method [38]

using slightly modified Ni et al. [42]. The modification of the histogram shifting strategy removes the limitation of sharing peak and zero point's information with the sender. The additional bit-plane data for embedding the secret data also increases the data hiding capacity. Thus, Chang et al.'s scheme is able to provide at least 3 times higher data hiding capacity when compared with Lo et al.'s scheme. To further increase the data hiding capacity while keeping the standard format of the AM(BTC) compressed codes, there have been several variants such as multilevel histogram shifting given by Zhao et al. [56] and Amita et al. [2], residual histogram shifting by Chang et al. [8], bi-stretch hiding by Li et al. [30], bit-plane flipping by Li et al. [29] and Lin & Liu [36]. Kim et al. [24] discussed a new data hiding scheme for AMBTC compressed images using histogram modification. Kim et al.'s scheme is able to increase the data hiding capacity quite significantly, however, it fails to retain the standard format of trios of AMBTC. Recently, Chen et al. [13] proposed a reversible data hiding method based on block classification and histogram modification strategy. The method positions the quantization levels of each block at the corners and then classifies the blocks into four categories. Each block is used to embed the secret data using a novel data hiding strategy.

3.2 Prediction error expansion based data hiding schemes

One of the most popular prediction error expansion based reversible data hiding schemes for AMBTC compressed images is discussed by Sun et al. [46]. Sun et al.'s scheme uses a joint neighbor coding technique and embeds the secret data into low-mean and high-mean tables to keep the bit-planes unchanged. Sun et al.'s method selects a prediction value according to the to-be-embedded bits and obtains the prediction errors for data embedding. Before embedding the secret data, Sun et al.'s method classifies the quantization values into referential and predictable groups. The quantization values in the first row, the first column and the last column are chosen as the referential quantization values as shown in the gray background of Fig. 5 to recover the predictable ones. Other quantization values are predictable as shown in Fig. 5 with a white background, where two bits of secret data can be carried by each quantization value. Since the embedding in higher and lower quantization values share the same procedures, only higher ones are taken for illustration. The illustrative example to explain Sun et al.'s method is given as follows. Since the data hiding process for high mean and low mean values is the same, the high mean value is considered as shown in Fig. 6.

Fig. 5 Neighboring indices of mean value H_i



$(01)_2$	$(10)_2$	$(11)_2$
$(00)_2$	H_i	

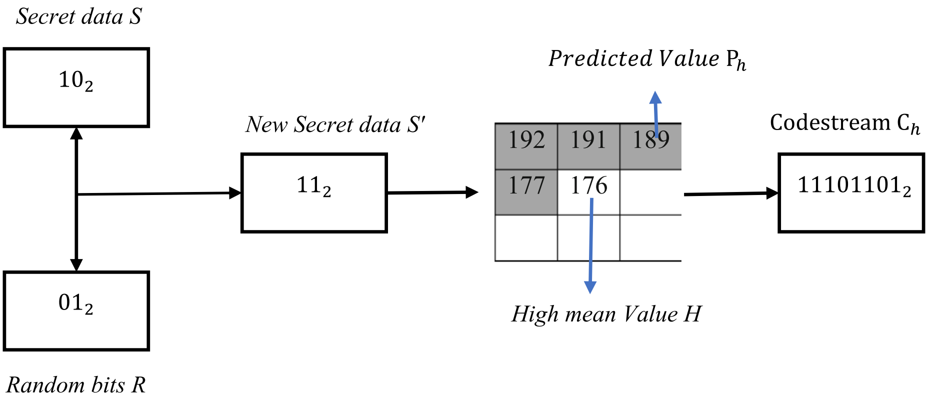


Fig. 6 An example of prediction error expansion-based method [46]

Example: Let H = 176 be the high mean value which is to be encoded, m = 4 and secret data S to be embedded is $(10)_2$. First of all, two-bit sequence R is generated at random and let $(01)_2$. Now, the sequence R is XOR operation with secret data S, S ⊕ R to get new secret data S′ to get the predicted location value P_h. Thus, $S' = (10 \oplus 01) = (11)_2$ and P_h is 189 in Fig. 5. Then, the prediction error E_h can be calculated as $H\text{-}P_h = 176–189 = −13$. The E_h is then represented into binary form as per the following cases.

- Case 1: If the E_h value is in the range of $–(2^m\text{-}1)$ to 0, then represent E_h value in m-bit binary forms concatenated by $(10)_2$.
- Case 2: If the E_h value is less than $–(2^m\text{-}1)$, then represent E_h value in m-bit binary forms concatenated by $(00)_2$.
- Case 3: If the E_h value is in the range of 0 to $(2^m\text{-}1)$, then represent E_h value in 8-bit binary forms concatenated by $(11)_2$.
- Case 4: If the E_h value is greater than $(2^m\text{-}1)$, then represent E_h value in 8-bit binary forms concatenated by $(01)_2$.

The concatenation is done on the left side so that the receiver can easily extract the original data. In this example, E_h is − 13 means E_h is represented as 10||1101 which belongs to Case 1. The final codestream C_h for high mean value H is represented as 11||10||1101, where the first two bits belong to S′. To extract the secret data S from C_h and recover high mean value H, the receiver will calculate XOR operation for the first two bits with sequence R as $S = (11 \oplus 01) = (10)_2$ and then where the predicted value P_h is located using the first two bits i.e., $(11)_2$ of $C_{h,}$. The value in the image at the location $(11)_2$ is 189 as shown in Figs. 5 and 6. The next two bits stated as D tell about the sign (positive or negative) on the prediction error E_h which can be extracted from C_h after the first four bits. If D belongs to (01) or (11) then the prediction error E_h is positive otherwise it is negative. In this example, D is 10 which means $E_h = (1101)_2 = 13$ is negative. Thus, the high mean value H can be obtained as $P_h + E_h = 189+ (−13) = 176$.

Wang et al. [51] discuss a new prediction-error expansion based reversible data hiding scheme for BTC compressed images. Wang et al.'s scheme utilizes the correlation among neighboring blocks to embed the secret data. Wang et al.'s scheme constructs the low mean value and high mean tables and embeds the secret data into both the means using prediction-error expansion given by Thodi et al. [49] only when the low mean value and the corresponding high mean value are not equal. Wang et al.'s scheme directly replaces the bit-plane by the secret data bits for the blocks having equal high mean value and low mean. Tsai et al. [50] discuss a new reversible data hiding scheme which is basically an extension of Sun et al. [46]. Tsai et al.'s scheme uses the median edge detector to predict the high mean and low mean values for neighboring blocks instead of a random predictor in Sun et al.'s scheme [46]. Tsai et al.'s scheme then embeds the secret data into the predicted difference based on the difference expansion technique. To further improve the performance of Sun et al. [46], Hong et al. [18] additionally introduce alternative prediction to minimize the prediction errors and efficient error division to minimize the number of bits for encoding the prediction errors. Thus, Hong et al.'s scheme is able to increase the data hiding capacity. Hong et al. [20] introduce a joint adaptive coding-based data hiding scheme that uses reversible integer transform to efficient predictive encoding. Thus, Hong et al.'s scheme claims to reduce the bitrate to represent the images and improves the embedding efficiency.

3.3 Type-1 based data hiding schemes

The Type-1 schemes modify the AMBTC codes/images in such a way that the ethos of compression is destroyed for hiding the secret data. Type-1 based techniques provide good data hiding capacity and maintain stego-image quality, however, fail to preserve the compression ratio as of the original image. One of the most popular Type-1 based reversible data hiding scheme for AMBTC encoded images is given by Lin et al. [37]. Lin et al.'s scheme combines the secret message bits with the bit-planes first of all. Then the combination (bit-planes and secret data bits) is used to classify the image blocks into two categories namely embeddable and non-embeddable. A block is said to be an embeddable block if the number of different combinations with secret data bit and bit-plane bits per block is more than 2, otherwise, the block is considered non-embeddable. If the block is an embeddable block, then the secret data can be considered to embed using four different combinations of average *AVG* and standard variation *var* as follows:

- If the combination is '00' where the first '0' is from the secret data and last '0' is from the bit-plane, then the corresponding pixel value for the image block will be *AVG* - *var*.
- If the combination is '01' where the first '0' is from the secret data and the '1' is from the bit-plane, then the corresponding pixel value for the image block will be *AVG* + *var*.
- If the combination is '10' where the first '1' is from the secret data and the '0' is from the bit-plane, then the corresponding pixel value for the image block will be *AVG* – *var* -1.
- IF the combination is '11', then the corresponding pixel value for the image block will be *AVG* + *var* +1.

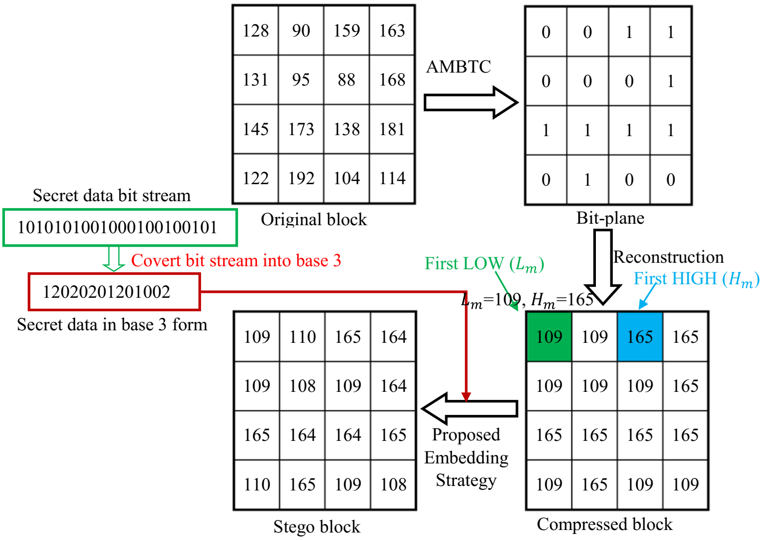
Thus, Lin et al.'s scheme embeds a decent amount of secret data if the secret data bit stream is good a mixture of zeros and ones. Sun et al. [47] extended Lin et al.'s scheme to overcome the limitation of continuous zeros or ones in secret data bit-stream. Thus, Sun et al.'s scheme significantly increases the data hiding capacity. However, since Lin et al. and Sun et al.'s schemes are not able to maintain the ethos of AMBTC compression for each embeddable block in the final stego-image, two schemes will have more than two different values. Zhang et al. [55] discuss a new reversible data hiding scheme for BTC compressed images based on the lossless coding of mean tables. Zhang et al.'s scheme constructs the mean tables both for low mean values and high mean values separately. And then partitions both the tables into 2 ×

2 blocks. The mean values are encoded into the code stream to hide the secret data. Huynh et al. [23] discuss minima-maxima preserving data hiding scheme for AMBTC compressed images. Huynh et al.'s scheme makes use of the least significant bit substitution method to hide the secret data into the compressed image. Huynh et al.'s scheme adaptively uses LSB substitution method for hiding the secret data. To provide the reversibility, Huynh et al.'s scheme does not embed the secret data into the first low and first high mean value of each image block. Therefore, the hiding capacity is significantly increased and the stego-image quality is also maintained. Chang et al. [10] discuss a joint neighborhood coding (JNC) based reversible data hiding scheme for AMBTC compressed images. The scheme makes use of JNC and XOR operators for hiding the secret data into the low and high means tables. Although the scheme provides high data hiding capacity, file/image size gets increased. The similar work is also carried out by Zhou et al. [57] by using gray code to improve performance in addition. To further increase the data hiding capacity, a new reversible data hiding scheme using pixel value adjusting strategy for AMBTC compressed images is introduced by Malik et al. [41]. Malik et al.'s scheme embeds the secret data into the AMBTC compressed image using some pre-defined embedding rules. Malik et al.'s scheme first converts the secret data bit stream into base-3 representation and then identifies the embeddable block using the difference of quantization levels (low-mean and high-mean) which should be greater than one. Each pixel of the embeddable blocks (except the first low mean and first high mean of each block) is used to embed the secret by modifying its value by one at most. The detailed illustrative example of Malik et al.'s scheme is given as follows.

Example: Let original image of size 4 × 4 pixels having intensity be 128, 90, 159, 163, 131, 95, 88, 168, 145, 173, 138, 181, 122, 192, 104, 114. The image is compressed using AMBTC compression to provide a trio having Low mean (L_m) = 109 and High mean (H_m) = 165 a bit-plane as shown in Fig. 7 for a raster scan manner. The trio is then used to construct the compressed image. Then, the secret message is embedded into the image. The scheme first converts the secret data bit stream into the base-3 format and then identifies the non-embeddable pixels from the compressed image/block. The non-embeddable pixels for every block are the first low mean and the first high mean valued pixels. The rest of the pixels are used for embedding the secret data. The scheme embeds one digit (base-3 converted) in every embeddable pixel by either incrementing/decrementing its value by one if the secret digit is '1'/'2' or leaving it unchanged if the secret digit is '0'. The illustration of the embedding process for hiding the secret data is shown in Fig.7. The non-embeddable pixels (i.e. the first Low mean pixel and the first High mean pixel) are discriminated by different colors respectively.

The extraction process at the receiving end is just reverse embedding process done at the sender side. The receiver first identifies the non-embeddable pixels from the image/block. Then uses the identified pixel values to extract the secret data from all the remaining pixels and restore the image.

- If the embeddable pixel value is equal to {Low meanm, High mean }, then the embedded secret data digit is '0' and there had been no change in the pixel value.
- If the pixel value is greater than by 1 at most than {Low mean (L_m), High mean (H_m)}, it means the embedded digit of the secret data is '1' and recovers the pixel value by decreasing by 1.
- Otherwise, the embedded digit of the secret data is '2' and recovers the pixel value by increasing it by 1.



Original block

128	90	159	163
131	95	88	168
145	173	138	181
122	192	104	114

Bit-plane

0	0	1	1
0	0	0	1
1	1	1	1
0	1	0	0

Compressed block

109	109	165	165
109	109	109	165
165	165	165	165
109	165	109	109

Stego block

109	110	165	164
109	108	109	164
165	164	164	165
110	165	109	108

Fig. 7 An example of Malik et al.'s scheme [41]

Thus, the compressed image is restored, and the secret data is completely extracted which is in the base-3 format. This secret data is then converted into binary form to get the original secret data bit stream.

3.4 Block classification based data hiding schemes

In this sub-section, some of the popular block classification-based data hiding schemes for AMBTC compressed images are described. Chuang and Chang [14] introduced block classification-based data hiding scheme for BTC compressed images in 2006. The scheme uses a user-defined threshold for classifying the BTC compresses code trios or blocks in two categories namely smooth and complex based on the difference of their quantitation levels, i.e. high mean value and low mean value. If the difference is smaller than the defined threshold, then the block is considered a smooth block otherwise it is considered a complex block. The scheme then embeds the secret data in the smooth blocks by replacing the bits of the corresponding bit-plane with the secret data bit stream. Chuang and Chang's scheme does not embed the secret data in the complex block because the replacement of bit-plane bits with the secret data bits may result in drastic deterioration on stego-image quality. Ou and Sun [43] discussed a new data hiding method to improve the quality of stego-image on Chuang and Chang's scheme. Ou and Sun's scheme basically recalculates the quantization levels, i.e. the high mean value and the low mean value of smooth blocks after hiding the secret data to reduce the difference. Additionally, Ou and Sun's scheme increases the data hiding capacity by embedding secret data bits into the quantization levels of complex blocks by changing their order and flipping the bits of the corresponding bit-plane as per the secret data bits like Chen et al. [12]. The illustrative example of Ou and Sun's scheme is given as follows.

Example: Let the secret data bit stream S be $(1110101010101 0011)_2$ and the threshold (thr) be 15. Further suppose two blocks each of size 4 × 4 pixels contain pixel values 98, 88, 83, 80, 74, 73, 70, 75, 78, 79, 78, 79, 80, 79, 77, 77 and 184, 209, 210, 203, 179, 206, 209, 200, 183, 203, 201, 191, 192, 209, 200, 188. Two blocks are compressed using AMBTC compression technique and obtain low mean values i.e. 75 & 186 and high mean values 83 & 205 along with two bit-planes as shown in Fig. 8.

Then, absolute differences D_1 and D_2 are calculated as $D_1 = |75 - 83| = 8 < \text{thr}$, which is less than the defined threshold value and as $D_2 = |186 - 205| = 19 > \text{thr}$, which is greater than the defined threshold value. It means the first block is a smooth block and the second block is a complex block. The secret data in a smooth block is embedded by replacing the bits of the bit-plane with the same number of bits of secret data. Thus, a new bit-plane is received. To minimize the caused distortion, mean values are recalculated according to the new bit-plane with some predefined equations. New mean values are 77 and 81 for the smooth block as shown in Fig.8. In the complex blocks, only one bit of secret data is embedded. If the bit to be embedded is '1', then the ordering of mean values is changed, and the bits of the bit-plane is flipped, otherwise, everything remains unchanged as shown in Fig. 8. The secret data is extracted from the received blocks by identifying their types. To identify the type, absolute differences D_1 and D_2 of mean values are calculated as follows $|77 - 81| = (3 < \text{thr})$, which is less than the defined threshold value and follows $|205 - 186| = (19 > \text{thr})$, which is greater than the defined threshold value. It means the first block is a smooth block and the second block is a complex block. The secret data from the smooth block is extracted by extracting its bit-plane bits, thus, the extracted secret data from the smooth block is 1110101010101001. In the case of a complex block, the receiver checks the mean values ordering if the first mean value is lower than the second one, and then the hidden secret bit is '0' otherwise it is '1'. Thus, complete secret data is extracted.

Huang et al. [22] improved the Ou and Sun [43] in terms of data hiding capacity. Huang et al.'s scheme utilized the difference of the quantization levels irrespective of their block type to embed the secret data. The quantization levels are modified for hiding the secret data in such a way so that their restoration to original value/form after extraction the secret data. Hong et al. [19] extended Ou and Sun's schemd to improve the stego-image quality by introducing a new

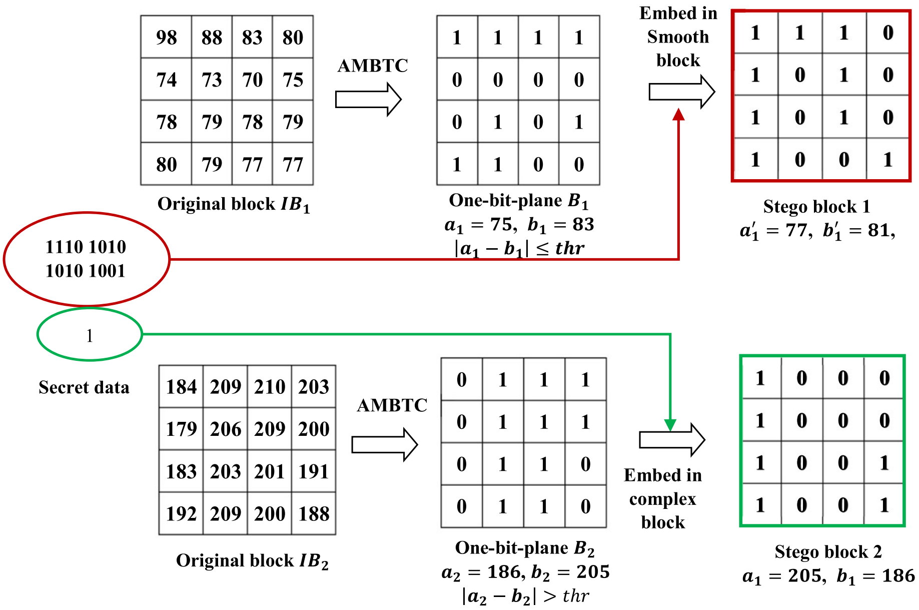


Fig. 8 An example of Ou and Sun's scheme [43]

scheme that efficiently processes the quantization levels of smooth block after hiding the secret data to reduce the caused distortion. Hong et al.'s scheme additionally uses a perturbation technique to hide two additional bits in the quantization levels. Hong [17] introduced a new data hiding method using a pixel pair matching technique. Hong' scheme further minimized the caused distortion by processing the smooth blocks efficiently while hiding the secret data. Additionally, Hong' scheme embeds a few more bits of secret data into the quantization levels of a smooth block by using adaptive pixel pair matching. Hong et al. [21] introduced a difference matching technique to extend Huang et al.'s scheme for minimizing the caused distortion while embedding the secret data into the quantization levels. The difference matching technique adjusts the quantization values to minimize the caused change and makes use of quantization level ordering for additional data embedding. Therefore, hiding capacity is increased with improved stego-image quality. Chen and Chi [11] discussed a new data the hiding scheme which categorizes the blocks into three categories namely smooth, complex-1 and complex-2 instead of two categories [22, 43]. Chen and Chi's technique uses two user-defined thresholds and classifies the blocks in three categories. The scheme embeds the secret data in smooth and complex-1 blocks in a similar fashion in Ou and Sun's scheme, however, additional two bits are embedded in complex-2 blocks. The work is further extended by Kumar et al. [27] by utilizing the complex block's bit-planes for additional embedding. Malik et al. [39, 40] discussed a new data hiding scheme that modifies the AMBTC encoding strategy. Malik et al.'s scheme produces four quantization levels and two bits bit-plane instead of two quantization levels and one-bit bit-plane [11, 22, 43]. Malik et al.'s scheme claims that additional bits used in bit-plane can be used for embedding the secret data and two extra quantization levels to help in reducing the caused distortion. Thus, Malik et al.'s scheme embeds the secret data even in complex blocks using a similar method. Therefore, the scheme provides superior stego-image quality and high data hiding capacity. However, Malik et al.'s scheme reduces the compression ratio by half.

3.5 Miscellaneous data hiding schemes

In this sub-section, some schemes that have been classified into the miscellaneous category are explained. Both lossless and lossy data hiding schemes are comprised. Tang et al. [48] worked a new AMBTC based steganography scheme using the interpolation technique. The original image is first down-scaled ¼ to its original size and then up-scaled to its original size using the interpolation technique. The up-scaled image is called as the cover image which will be used for hiding the secret data. The scheme makes use of a rotation method to improve security. Thus, high data hiding capacity along with good quality stego-image is achieved. However, the use of the interpolation technique harms the prospects of the reversibility of the original compressed image. Due to the requirement of content security, encrypted domain based reversible data hiding schemes have seen significant growth in recent years. Yin et al. [54] proposed a secure reversible data hiding scheme for encrypted AMBTC images in 2017. Yin et al.'s scheme first ciphers/encrypts the quantization levels using a random key and then uses prediction error histogram modification strategy for embedding the secret data. Thus, two keys namely encryption and hiding keys are used. The person possessing only the encryption keys will be able to decrypt the image to get an image that is similar to the original but not the hidden secret data one whereas the person with only the hiding key will be able to get the hidden secret data but not the original image. Only the person holding both the keys will be able to get both the secret data and the original image. Thus, the scheme increases the security of the contents. Chang et al. [9] discuss a novel steganography scheme for BTC-compressed images using a dynamic programming strategy. The scheme first searches for the optimal solution of the bijective mapping function for LSB substitution and then quantization values are modified to embed the secret data. The modification is done in such a way that high data hiding capacity can be achieved without making a bigger change to the image. Further, the scheme is able to maintain the same bitrate as of original block truncation coding.

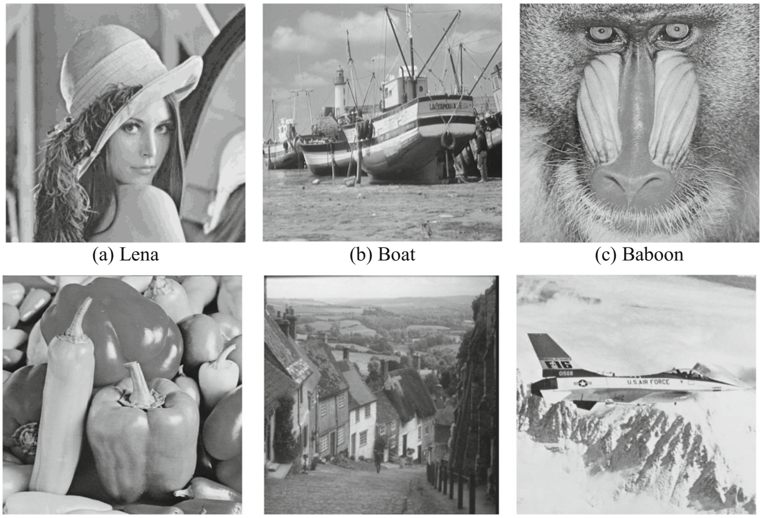
4 Performance analysis and discussion

To analyze the performance of AMBTC based data hiding schemes, three important parameters, i.e. data hiding capacity, compression ratio, and visual quality were used. Data hiding capacity refers to the total number of bits embedded in the cover image and the compression ratio is achieved by dividing the size of the original/input image with the compressed/output image. The visual quality is measured using the peak signal-to-noise ratio (PSNR), which is calculated using Eq. (5) as follows:

$$\mathrm{PSNR}=20\log_{10}\left[\frac{P_l}{\frac{1}{M\times N}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(x_{ij}-y_{ij}\right)^2}\right] \tag{5}$$

Where, P_l is the maximum peak value of the image and x_{ij} and y_{ij} are the pixels located at the i$^{\text{th}}$ row and j$^{\text{th}}$ column of cover image x and stego-image y for each of size M × N pixels, respectively.

Six grayscale cover images are used for experimental results namely Lena, Boat, Baboon, Peppers, Goldhill and F16 as shown in Fig.9. The secret data is embedded in 4 × 4 blocks for



(a) Lena (b) Boat (c) Baboon

(d) Peppers (e) Goldhill (f) F16

Fig. 9 Cover images with 512×512 size

all the methods being used in performance analysis and discussion. The comparative analysis of existing AMBTC based data hiding schemes is done according to their style of working.

The comparison of different histogram shifting based reversible data hiding schemes for visual image quality and data hiding capacity is shown in Figs. 10 and 11, respectively. In Fig.10, Kim et al.' scheme [24] is outperforming for histogram shifting based reversible data hiding schemes in terms of PSNR whereas Bi-Stretch [30] is performing better than all the other related schemes in terms of hiding capacity as shown in Fig. 11. The main reason for the performance is the non-compliance of the AMBTC format as blocks of the output image may have more than two different values rather than the usual two values. Thus, these schemes provide a good quality marked image and higher data hiding but at the cost of compression ratio.

The experimental results of PSNR and hiding capacity (in bits) for prediction based reversible data hiding schemes on AMBTC compressed images are shown in Figs. 12 and 13. In Figs.12 and 13, Hong et al. [22], Sun et al. [46] and joint adaptive coding [20] are outperforming in both PSNR and data hiding capacity for all cover images. The main reason for such superior performance is the efficient prediction method, which takes into account the median edge detector or joint adaptive coding for low and high mean values prediction. Further, adaptive embedding and efficient error division helped in minimizing the number of bits for encoding the prediction errors.

The Type-1 schemes basically embed the secret data in the compressed image by modifying every pixel value of the image. Therefore, the stego-image of Type-1 schemes will have more than two different values per each block, unlike the original AMBTC compressed image which has just two different values, low mean and high mean values. Thus, Type-1 schemes diminish the ethos of AMBTC compression which results in less compression ratio.

To analyze the performance of type-1 schemes quantitatively, the same parameters namely visual image quality and data hiding capacity are used as shown in Figs. 14 and 15. In Fig.14, Chang et al. [10] is performing better than Type-1 schemes in terms of PSNR. Malik et al.'s scheme [41] has almost 1.5 times higher data hiding capacity than other related methods such as Chang et al. [10], Lin et al. [37] and Zhang et al. [55] as shown in Fig. 15. Malik et al. [41] provides higher data hiding capacity because it is able to use every pixel of the image for secret data embedding. Further, the quality of the output image is maintained as the pixel values are changed by one at most while hiding the secret data. Specifically, Malik et al.'s scheme embeds almost 1.5 bits in every pixel except two pixels per block which has higher data hiding

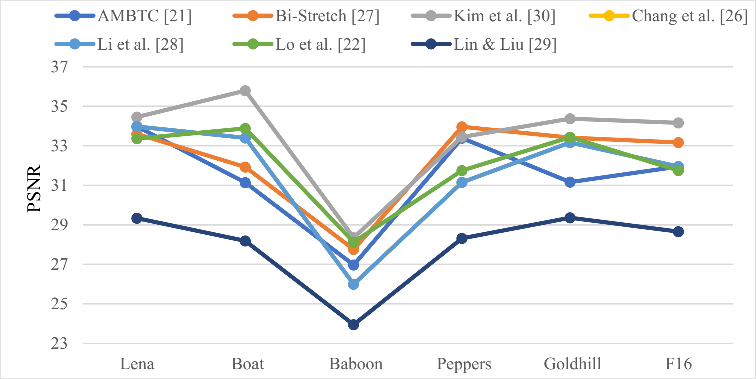


Fig. 10 Comparison of PSNR on different Histogram-based schemes

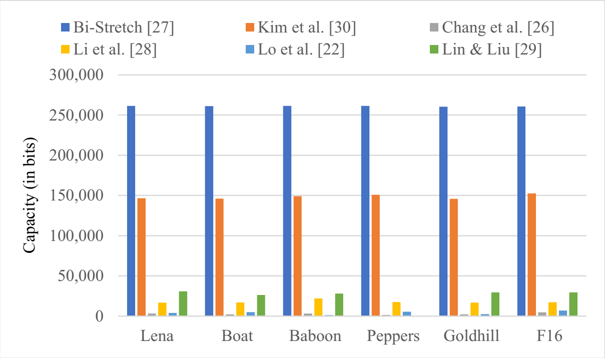


Fig. 11 Embedding capacity on different histogram-based schemes

capacity while maintaining stego-image quality. However, the achieved compression ratio is greatly diminished since every pixel value by eight bits is required to represent.

The block classification-based data hiding schemes for AMBTC compressed images are lossy data hiding schemes. These schemes embed the secret data by replacing the bit-plane with the secret data bits which cause permanent loss of original bit-plane values. There has been a lot of work on block classification-based data hiding schemes for AMBTC compressed images.

To analyse the performance, the experimental results and comparison of classification-based schemes like Malik et al. [39], Hong [17], Huang et al. [22], Ou & Sun [43], Hong et al. [19] Chuang and Chang [14] in terms of PSNR and embedding capacity are shown in Fig. 16–21. In Fig. 16, Malik et al. [39] has high data hiding capacity and has a slight edge on PSNR value as well. The good performance is its four-quantization levels and two-bit bit-plane to reduce the compression ratio by half. The second-best performance in terms of data hiding capacity is achieved by Hong [17]. The performance in terms of PSNR in Hong [17] is achieved along with data hiding capacity when compared with Huang et al. [22], Ou & Sun [43] and Hong et al. [19]. Moreover, Hong's method maintains the same compression ratio of Huang et al. [22], Ou & Sun [43], and Hong et al. [19]. Therefore, block classification-based schemes have performed quite well in the aspect of image quality and data hiding capacity. However, block classification-based schemes have a lack limitation of reversibility due to the bit-plane replacement. In future works, the researchers can work on the problem of irreversibility.

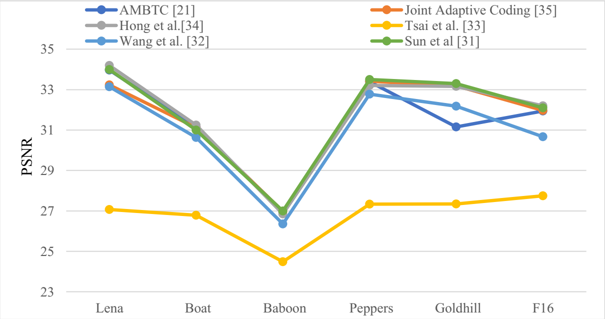


Fig. 12 PSNR on different prediction-based schemes

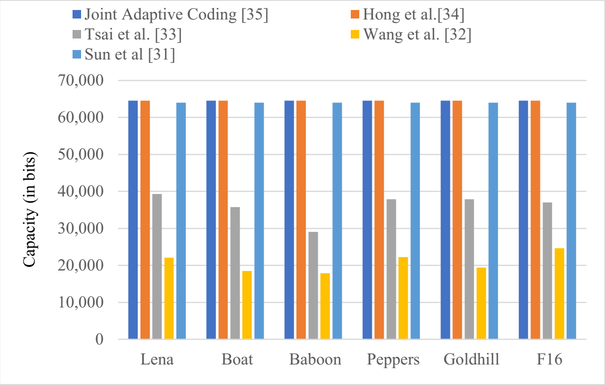


Fig. 13 Embedding capacity on different prediction-based schemes

To comprehensively evaluate the performance of AMBTC compression-based data hiding schemes, the best performing schemes from each defined category i.e., (histogram, prediction, Type-1, block classification) are cumulatively evaluated using combo graph as shown in Fig. 17 for all test images. In Fig. 17, Kim et al. [24] from the histogram-based schemes category has the highest PSNR whereas Malik et al. [41] from the Type-1 category has the highest embedding capacity. The main reason behind the outstanding performance of two schemes is the violation of the syntax of AMBTC encoding which results in a reduced compression ratio.

Thus, it can be summarized that the existing data hiding schemes (other than the block classification-based data hiding schemes) for AMBTC compressed images/codes have achieved high data hiding capacity while maintaining image quality. However, some of the schemes like Kim et al. [24] and Malik et al. [41] have violated the ethos of AMBTC compression while hiding the secret data, thus, results in a low compression ratio. The violation may invite the suspicion of the potential attackers. In the future research direction, the researchers can explore to preserve the format of AMBTC codes without compromising with the data hiding capacity and image quality.

There has been a lot of work in different directions such as histogram shifting, prediction error expansion, Type-1, block classification on AMBTC compressed images/codes for hiding the secret data. The experimental results show that the quality is being maintained for most of the schemes while hiding the secret data.

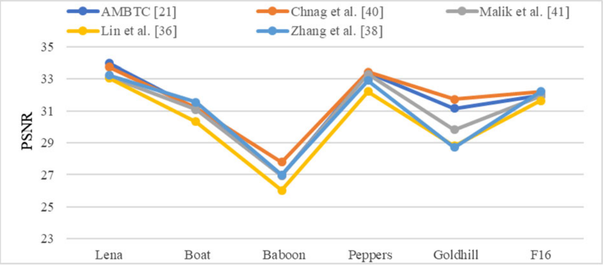


Fig. 14 PSNR on different Type-1schemes

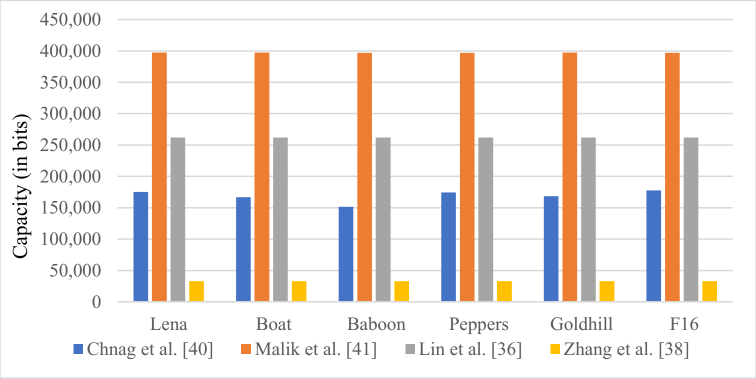
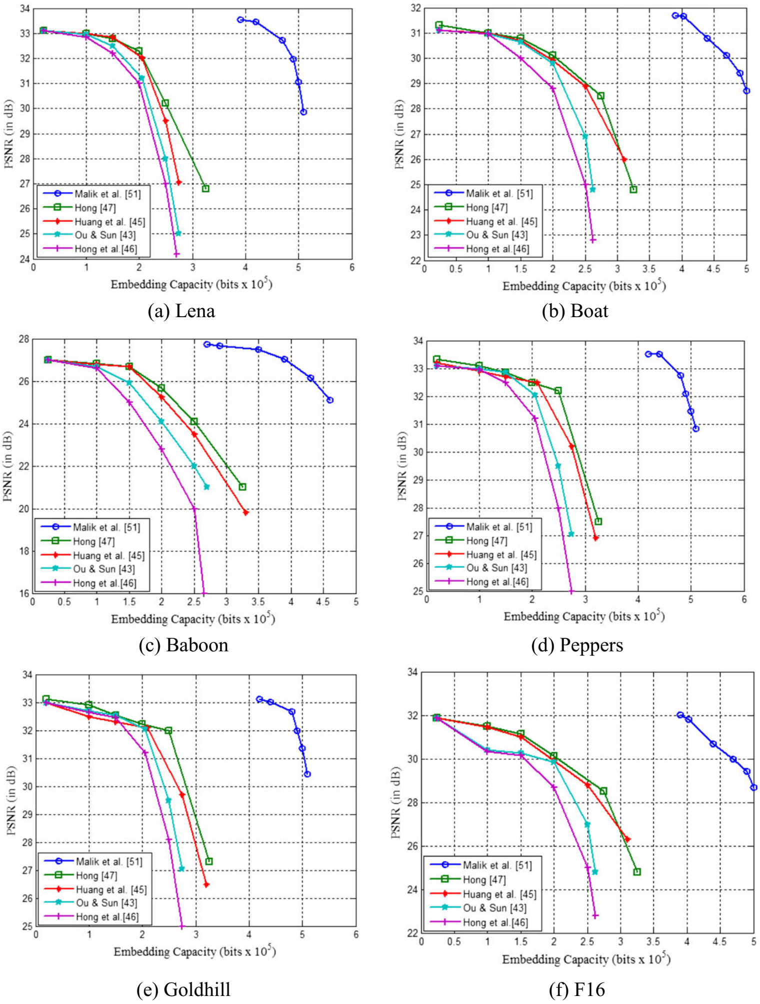


Fig. 15 Embedding capacity on different Type-1schemes



(a) Lena

(b) Boat

(c) Baboon

(d) Peppers

(e) Goldhill

(f) F16

Fig. 16 PSNR versus embedding capacity

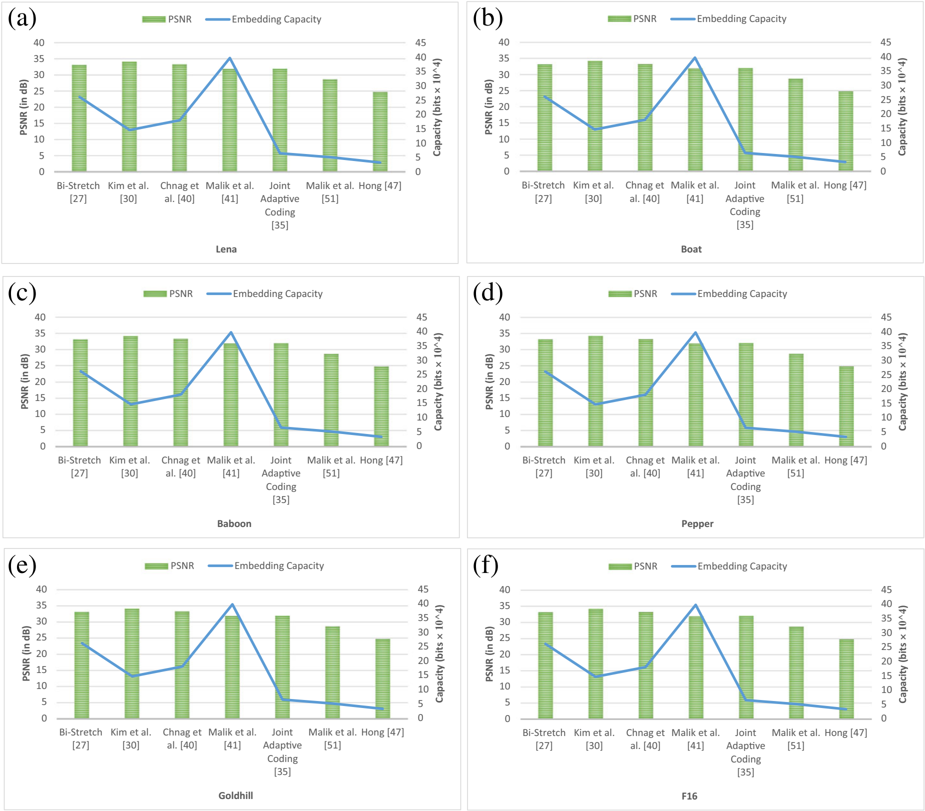


Fig. 17 Comparison of PSNR at maximum embedding capacity

5 Conclusions and summary

In this systematic survey, AMBTC based data hiding schemes have been discussed and analyzed. Existing schemes have utilized multiple popular data hiding concepts like histogram shifting, prediction error expansion and LSB replacement to hide the secret data efficiently. The performance of the previous schemes has been analyzed using hiding capacity, compression ratio, and visual quality. The main analyzed results of previous schemes are as follows: (a) Compressed image can be completely restored along with extraction of the secret data, (b) Reduction in the required bandwidth to transmit the secret data by AMBTC is a lossy compression method, (c) High data hiding capacity can be provided without distortion the stego-image quality. As a result, different data hiding schemes using the AMBTC compressed images have performed quite well. However, the stego-image quality has been limited to the quality of AMBTC compressed image for the format of AMBTC trios. To enhance the image quality further, an adaptive AMBTC based data hiding method could be required. Further, new direction works can be extended on AMBTC based data hiding schemes by interpolation and/or encryption-based data hiding schemes in some way.

Acknowledgements This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2018R1D1A1A09081842) and Brain Pool program funded by the Ministry of Science and ICT through the National Research Foundation of Korea (No. 2018H1D3A2065993).

References

1. Amirulhaqi A, Purboyo TW, Nugrahaeni RA (2017) Security on GIF images using steganography with LSB method, spread spectrum and the vigenere cipher. Int J Appl Eng Res 12(23):13604–13609
2. Amita, Kaur A, Kumar M (2018) Reversible data hiding in absolute moment block truncation coding compressed images using adaptive multilevel histogram shifting technique. Int J Inf Comput Secur 10(2): 261-275. https://doi.org/10.1504/IJICS.2018.091463
3. Chan CK, Chen LM (2004) Hiding data in images by simple LSB substitution. Pattern Recogn 37(3):469–474
4. Chang CC, Lin CY (2007) Reversible steganographic method using SMVQ approach based on declustering. Inf Sci 177(8):1796–1805
5. Chang CC, Lin CC, Tseng CS, Tai WL (2007) Reversible hiding in DCT-based compressed images. Inf Sci 177(13):2768–2786
6. Chang CC, Lin CY, Fan YH (2008) Lossless data hiding for color images based on block truncation coding. Pattern Recogn 41:2347–2357
7. Chang CC, Wu HL, Chung TF (2014) Applying histogram modification to embed secret message in AMBTC, Tenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing. https://doi.org/10.1109/IIH-MSP.2014.128
8. Chang CI, Hu CY, Chen LW, Lu CC (2015) High capacity reversible data hiding scheme based on residual histogram shifting for block truncation coding. Signal Process 108:376–388
9. Chang CC, Son Y, Nguyen T (2015) A novel data hiding scheme for block truncation coding compressed images using dynamic programming strategy. Sixth International Conference on Graphic and Image Processing (ICGIP 2014). doi: https://doi.org/10.1117/12.2179686
10. Chang CC, Chen TS, Wang YK, Liu Y (2018) A reversible data hiding scheme based on absolute moment block truncation coding compression using exclusive OR operator. Multimed Tools Appl 77(7):9039–9053
11. Chen YY, Chi KY (2017) Cloud image watermarking: high quality data hiding and blind decoding scheme based on block truncation coding. Multimedia Systems:1–13. https://doi.org/10.1007/s00530-017-0560-y
12. Chen J, Hong W, Chen TS, Shiu CW (2008) Steganography for BTC compressed images using no distortion technique. Imaging Sci J 58:177–185
13. Chen YY, Hsia CH, Jhong SY, Lin HJ (2018) Data hiding method for AMBTC compressed images. J Ambient Intell Humaniz Comput:1–9. https://doi.org/10.1007/s12652-018-1048-0
14. Chuang JC, Chang CC (2006) Using a simple and fast image compression algorithm to hide secret information. Int J Comput Appl 28:1735–1743
15. Delp E, Mitchell O (1979) Image compression using block truncation coding. IEEE Trans Commun 27(9):1335–1341
16. Dutta P, Bhattacharyya D, Kim TH (2009) Data hiding in audio signal: a review. International Journal of Database Theory and Application 2(2):1-8
17. Hong W (2018) Efficient data hiding based on block truncation coding using pixel pair matching technique. Symmetry 10(2). https://doi.org/10.3390/sym10020036
18. Hong W, Ma YB, Wu HC (2017) An efficient reversible data hiding method for AMBTC compressed images. Multimed Tools Appl 76(4):5441–5460
19. Hong W, Chen T, Yin Z, Luo B, Ma Y (2017) Data hiding in AMBTC images using quantization level modification and perturbation technique. Multimed Tools Appl 76:3761–3782
20. Hong W, Zhou X, Weng S (2018) Joint adaptive coding and reversible data hiding for AMBTC compressed images. Symmetry 10. https://doi.org/10.3390/sym10070254
21. Hong W, Li Y, Weng S (2018) A difference matching technique for data embedment based on absolute moment block truncation coding. Multimed Tools Appl:1–20
22. Huang YH, Chang CC, Chen YH (2017) Hybrid secret hiding schemes based on absolute moment block truncation coding. Multimed Tools Appl 76:6159–6174
23. Huynh NT, Bharanitharan K, Chang CC, Liu Y (2018) Minima-maxima preserving data hiding algorithm for absolute moment block truncation coding compressed images. Multimed Tools Appl 77(5):5767–5783
24. Kim C, Shin D, Leng L, Yang CN (2016) Lossless data hiding for absolute moment block truncation coding using histogram modification. J Real-Time Image Proc. https://doi.org/10.1007/s11554-016-0641-8
25. Kumar R, Malik A, Singh S, Chand S (2016) A high capacity email based text steganography scheme using huffman compression. 3rd International Conference on Signal Processing and Integrated Networks (SPIN), p 53–56
26. Kumar R, Malik A, Singh S, Kumar B, Chand S (2016) Reversible data hiding scheme for LZW codes using even-odd embedding strategy. Int. Conf. on Computing, Communication and Automation (ICCCA), p 1399–1403
27. Kumar R, Kim DS, Jung KH (2019) Enhanced AMBTC based data hiding method using hamming distance and pixel value differencing. Journal of Information Security and Applications 47:94–103
28. Lema M, Mitchell O (1984) Absolute moment block truncation coding and its application to color images. IEEE Trans Commun 32(10):1148–1157

29. Li CH, Lu ZM, Su YX (2011) Reversible data hiding for BTC-compressed images based on bitplane flipping and histogram shifting of mean tables. Inf Technol J 10(7):1421–1426
30. Li F, Bharanitharan K, Chang CC, Mao Q (2016) Bi-stretch reversible data hiding algorithm for absolute moment block truncation coding compressed images. Multimed Tools Appl 75(23):16153–16171
31. Liao X, Wen Q, Zhang J (2013) Improving the adaptive steganographic methods based on modulus function. IEICE Trans Fundam Electron Commun Comput Sci 96(12):2731–2734. https://doi.org/10.1587/transfun.e96.a.2731
32. Liao X, Li K, Yin J (2016) Separable data hiding in encrypted image based on compressive sensing and discrete fourier transform. Multimed Tools Appl 76(20):20739–20753. https://doi.org/10.1007/s11042-016-3971-4
33. Liao X, Qin Z, Ding L (2017) Data embedding in digital images using critical functions. Signal Process Image Commun 58:146–156. https://doi.org/10.1016/j.image.2017.07.006
34. Liao X, Guo S, Yin J, Wang H, Li X, Sangaiah AK (2017) New cubic reference table based image steganography. Multimed Tools Appl 77(8):10033–10050. https://doi.org/10.1007/s11042-017-4946-9
35. Liao X, Yin J, Guo S, Li X, Sangaiah AK (2018) Medical JPEG image steganography based on preserving inter-block dependencies. Comput ElectrEng 67:320–329. https://doi.org/10.1016/j.compeleceng.2017.08.020
36. Lin CC, Liu XL (2012) A reversible data hiding scheme for block truncation compressions based on histogram modification. In: Genetic and Evolutionary Computing (ICGEC), Sixth International Conference on, p 157–160
37. Lin CC, Liu XL, Tai WL, Yuan SM (2013) A novel reversible data hiding scheme based on AMBTC compression technique. Multimed Tools Appl. https://doi.org/10.1007/s11042-013-1801-5
38. Lo CC, Hu YC, Chen WL, Wu CM (2014) Reversible data hiding scheme for BTC-compressed images based on histogram shifting. Int J Secur Appl 8(2):301–314
39. Malik A, Sikka G, Verma HK (2016) A high payload data hiding scheme based on modified AMBTC technique. https://doi.org/10.1007/s11042-016-3815-2
40. Malik A, Sikka G, Verma HK (2018) A high capacity data hiding scheme using modified AMBTC compression technique. International Arab Journal of Information Technology (IAJIT)
41. Malik A, Sikka G, Verma HK An AMBTC compression based data hiding scheme using pixel value adjusting strategy. Multidim Syst Sign Process. https://doi.org/10.1007/s11045-017-0530-8
42. Ni Z, Shi YQ, Ansari N, Su W (2006) Reversible data hiding. IEEE Trans Circuits Syst. Video Technol 16(3):354–362
43. Ou D, Sun W (2015) High payload image steganography with minimum distortion based on absolute moment block truncation coding. Multimed Tools Appl 74:9117–9139
44. Paruchuri JK, Cheung SC, Hail M (2009) Video data hiding for managing privacy information in surveillance systems. EURASIP J Inf Sec: 1–18. https://doi.org/10.1155/2009/236139
45. Shiu PF, Tai WL, Jan JK, Chang CC, Lin CC (2019) An interpolative AMBTC-based high-payload RDH scheme for encrypted images. Signal Process Image Commun 74:64–77
46. Sun W, Lu ZM, Wen YC, Yu FX, Shen RJ (2013) High performance reversible data hiding for block truncation coding compressed images. SIViP 7(2):297–306
47. Sun S, Yin Z, Tang J, Luo B (2017) Improved reversible data hiding scheme based on AMBTC compression technique. International Conference on Industrial IoT Technologies and Applications Industrial IoT: Industrial IoT Technologies and Applications, p 111–118
48. Tang M, Zeng S, Chen X, Hu J, Du Y (2017) An adaptive image steganography using AMBTC compression and interpolation technique. Optik 127(1):471–477
49. Thodi DM, Rodriguez JJ (2007) Expansion embedding techniques for reversible watermarking. IEEE Trans Image Process 16(3):721–730
50. Tsai YY, Chan CS, Liu CL, Su BR (2014) A reversible steganographic algorithm for BTC-compressed images based on difference expansion and median edge detector. The Imaging science Journal 62(1):48–55
51. Wang K, Hu Y, Lu ZM (2012) Reversible data hiding for block truncation coding compressed images based on prediction-error expansion. In: 2012 Eighth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Piraeus, p 317–320
52. Wu DC, Tsai WH (2003) A steganographic method for images by pixel-value differencing. Pattern Recogn Lett 24(9):1613–1626
53. Yang B, Schmucker M, Funk W, Brush C, Sun S (2004) Integer DCT-based reversible watermarking for images using compounding technique. In: Proceedings of the SPIE, Security, Steganography and Watermarking of Multimedia Contents, vol. 5306, p 405–415
54. Yin Z, Niu X, Zhang X, Tang J, Luo B (2018) Reversible data hiding in encrypted AMBTC images. Multimed Tools Appl 77(14):18067–18083
55. Zhang Y, Guo S-Z, Lu Z-M, Luo H (2013) Reversible data hiding for BTC-compressed images based on lossless coding of mean tables. IEICE Trans Commun 96(2):624–631

56. Zhao ZF, Tang LL (2012) High capacity reversible data hiding in AMBTC-compressed images. Int J Digit Content Technol Appl 6:205–211
57. Zhou X, Hong W, Peng C, Chen TS, Zhai Y (2018) A reversible data hiding scheme for AMBTC images using gray code and exclusive-or approaches, 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Rajeev Kumar received his B.S. degree in Information Technology from Uttar Pradesh Technical University in 2007 and the M.S. degree in Computer Science and Engineering from University of Delhi in 2012. He received the Ph.D. degree in Computer Engineering from University of Delhi in 2017, India. He had been employed as a lecturer at HRIT, India from 2007 to 2009 and ESAR College of Engineering, India from 2009 to 2010. Currently, he is a senior consultant at CPA global support services, India from 2012 and works as a researcher at the Department of Cyber Security, Kyungil University and WIT from 2018 to 2019, South Korea. His current research interests are steganography and wireless sensor networks.

Ki-Hyun Jung received his B.S. degree in Computer Engineering from Kyungpook National University in 1995 and the M.S. degree in Computer Engineering from Kyungpook National University in 1997, South Korea. He received the Ph.D. degree in Computer Engineering from Kyungpook National University in 2007, South Korea. He had been employed as a senior researcher at Agency for Defense Development, South Korea from 1997 to 2003. He was a professor at the School of Computer Information, Yeungjin College, South Korea from 2003 to 2015. Currently, he is a professor at the Department of Cyber Security, Kyungil University, South Korea from 2015. He has selected his biography for inclusion in the Who's Who in the World from 2011 to 2019. His current research interests are information hiding, watermarking, cryptography, network security, game & mobile programming.