INTERNATIONAL JOURNAL OF INTEGRATED ENGINEERING
ISSN: 2229-838X e-ISSN: 2600-7916

IJIE

Vol. 16 No. 3 (2024) 290-301
https://publisher.uthm.edu.my/ojs/index.php/ijie

Block-Classification-Based AMBTC with Neural Networks for Image Compression

Aqeel Noori[1,2], Norulhusna Ahmad[1]*, Siti Armiza Mohd Aris[1], Norliza Mohd Noor[1]

[1] *Faculty of Artificial Intelligence, Universiti Teknologi Malaysia, 54100, Kuala Lumpur, MALAYSIA*
[2] *Education College of Pure Science, Computer Science Dept, Basrah University, Basrah, IRAQ*

*Corresponding Author: norulhusna.kl@utm.my
DOI: https://doi.org/10.30880/ijie.2024.16.03.026

Article Info

Received: 20 August 2024
Accepted: 24 September 2024
Available online: 9 November 2024

Keywords

Image compression, lossy image compression, absolute moment block truncation coding, block classification, artificial neural networks

Abstract

The world has recently witnessed a rapid revolution in multimedia signal processing. Images are one of the most widely used media that need a large amount of data to be represented. Because of the restrictions of limited bandwidth and storage capacity, image compression is a necessity. AMBTC is a straightforward lossy image compression scheme, and studies are still being conducted to improve its performance. This paper incorporated AMBTC with block classification and artificial neural networks to lower the bitrate and preserve the image quality. The proposed scheme was benchmarked with the recent AMBTC techniques for greyscale images. According to the results, the proposed method significantly improved over conventional AMBTC by achieving 16% bitrate reduction while preserving 99.19% of AMBTC's Peak Signal to Noise Ratio (PSNR).

1. Introduction

Millions of images are uploaded and transmitted over the internet; to represent, save, and transfer them effectively, they must be compressed. Image compression is an essential image-processing technique widely used nowadays. Joint picture expert group (JPEG), vector quantization (VQ) and absolute moment block truncation coding (AMBTC) are the well-known lossy image compression techniques. AMBTC is an uncomplicated classical image compression technique used to compress images; Lema and Mitchel invented it in 1984 [1], and it is the enhanced version of block truncation coding (BTC).

AMBTC partitions the image into a set of non-overlapping blocks; for each block, the mean is computed and used as a threshold to classify the block pixels into two groups and calculate the higher and lower means. Higher and lower means represent the quantization levels of each block. Binary bit-plane blocks are generated by setting the value '0' to pixels lower than the mean and the value '1' to pixels equal or exceed the mean. In decoding, the two quantization levels, higher mean, and lower mean, will be used to reconstruct the compressed image by substituting bit-plane zeros with equivalent lower mean and ones with equivalent higher mean. Compared to BTC, AMBTC is efficient in terms of image quality and has less computational cost [2], but they share the same bitrate. Many studies have been conducted to minimize the BTC's high bitrate [3]-[6]. These studies use different techniques like bit-plane interpolation, visual patterns, variable-size block segmentation, and block prediction using similarity.

Previously, AMBTC was integrated with neural networks for image compression to lower the bitrate in [7]. Although the integration reduced the bitrate, the Peak Signal to Noise Ratio (PSNR) value is still lower than the conventional AMBTC method which means that the quality of reconstructed images is poor. Thus, this paper aims to enhance the previous work by utilizing block classification technique. The addition of the classification technique will provide the ability to determine which data can be considered redundant and what data are crucial and need to be preserved, this will grant a trade-off between reducing the bitrate and preserving the quality of AMBTC reconstructed images. The main contribution of the paper listed below,

- This paper proposed a compression system for grayscale images that uses artificial neural networks with AMBTC.
- Adding the block classification strategy to the proposed model to classify blocks into smooth blocks(trainable) and complex blocks (non-trainable).
- Evaluate the proposed model performance using the two metrics: bitrate (BR) and peak signal to noise ratio (PSNR).

The following sections organize the paper: Section 2 presents the related work of recent advances in AMBTC and BTC. Section 3 explains the Algorithm of AMBTC for image compression. Section 4 explains the proposed method block classification- based AMBTC with artificial neural networks. Section 5 discusses the experimental results of the proposed scheme and makes a comparison with conventional AMBTC and previous study AMBTC with neural networks, Section 6 concludes the overall paper and future works.

2. Related Works

Absolute moment Block Truncation Coding (AMBTC) has evolved significantly since its inception, leading to various adaptations and enhancements aimed at improving image compression efficiency, for both grayscale and color images. This section demonstrates the recent advances in AMBTC; as well as some of the recent studies that use artificial neural networks in image compression.

Hu et al. [8] employed quadtree segmentation with AMBTC. The quadtree was utilized to adaptively segment color images into blocks of varying sizes. This method leveraged the similarity among neighboring pixels, resulting in better compression ratios without significant loss in image quality. The quadtree segmentation effectively minimized redundant information, making it a powerful tool in AMBTC-based image compression.

Hui and Zhou [9] introduced a new image compression scheme that further refined AMBTC by incorporating real-time block classification and a modified threshold for pixel grouping. This approach aimed to enhance compression efficiency by dynamically adjusting the threshold values used for block segmentation, leading to better preservation of important image features. This modification was particularly beneficial for high-contrast images, where traditional AMBTC methods often faltered.

Kumar et al. [10] presented a study that explored a Human Visual System (HVS)-based approach to enhance AMBTC for color image compression. By integrating HVS principles, the study introduced interpolation techniques that improved perceived image quality, especially in areas of high visual importance. This approach highlighted the importance of aligning compression techniques with human perceptual characteristics to achieve better results.

Ahmad et al. [11] proposed a different approach that focused on adaptive multilevel AMBTC for color image coding. This technique allowed for varying levels of compression within the same image, adapting to the complexity of different regions. The adaptive nature of this method provided a balanced trade-off between compression efficiency and image quality, making it suitable for applications where both factors are critical.

Xiang et al. [12] presented adaptive multilevel grouping AMBTC (MGAMBTC), where the image is segmented into blocks, and pixels of that block will be grouped into one to four groups, depending on the block's complexity. Smooth blocks need only one group, slightly complex blocks need two groups, while complex blocks need three or four groups.

Chaung et al. [13] enhanced Xiang's approach by presenting an improved multilevel grouping AMBTC (iMGAMBTC); this scheme depends on the concept of resemblance between nearby blocks. So, to encode any inter-block, the nearby encoded blocks will be searched to find the most similar one, and its code will be used to encode the current block. iMGAMBTC also utilized an entropy-based indicators generation mechanism to minimize the bitrate further.

Chen et al. [14] developed a hybrid encoding scheme for AMBTC-compressed images using a ternary representation technique. This method combined the strengths of AMBTC with ternary encoding to achieve higher compression ratios while maintaining image quality. The hybrid approach demonstrated that combining multiple coding strategies could lead to more robust and efficient image compression methods.

Kumar et al. [15] introduced an enhanced interpolation-based AMBTC technique using Weber's Law. This method further refined image compression by incorporating perceptual models that better aligned with human vision. The application of Weber's Law in the interpolation process led to more natural-looking compressed images, particularly in areas with gradual intensity changes.

Artificial neural networks have been used in image compression, where in most of the studies, the compression or encoding is done between the input and hidden layers. At the same time, decompression or decoding is accomplished between hidden and output layers [16]-[18]. Alshehri et al. [19] presented a study that used neural networks to estimate the removed data to reconstruct the compressed image. Another study presented by Chakib et al. [20] that used a backpropagation neural network as a classifier to assign the suitable wavelet transform to compress each image.

Table 1 *Summary of the AMBTC advances from the literature*

Authors	Method	Advantages	Limitations
Hu *et al.* [8]	Quadtree Segmentation with BTC	Efficiently reduces data redundancy; Provides adaptive block size for better quality.	Computational complexity due to quadtree segmentation.
Hui and Zhou [9]	Real-Time Block Classification, Modified Threshold technique	Real-time classification enhances speed; Modified threshold improves grouping accuracy.	May struggle with highly complex images, reducing compression efficiency.
Kumar *et al.* [10]	Human Visual System, Interpolation scheme with AMBTC	Mimics human visual system for better perceptual quality; Interpolation enhances compression without losing details.	Higher computational cost due to the interpolation process.
Ahmad *et al.* [11]	Adaptive Multilevel BTC	Offers flexibility in handling different image types; Effective in balancing quality and compression ratio.	May introduce artifacts in highly textured or detailed images.
Xiang *et al.* [12]	Dynamic Multi-Grouping with AMBTC	Improves compression efficiency; Reduces redundancy effectively.	Complexity increases with the number of groups, making it less suitable for real-time applications.
Chaung *et al.* [13]	Improved Multi-Grouping with AMBTC and entropy-based indicators generation mechanism	Enhances versatility by adapting to different grayscale levels; Optimizes compression.	Limited effectiveness for color images, restricting application to grayscale images only.
Chen *et al.* [14]	Ternary Representation in Hybrid Encoding	Improves compression while retaining image details; Effective for various image types.	Higher implementation complexity due to the ternary representation and clustering algorithm.
Kumar *et al.* [15]	Weber's Law, Interpolation with AMBTC	Incorporates Weber's Law for perceptual compression; Improved interpolation technique enhances efficiency.	May lead to higher computational costs; Effectiveness depends on accurate perception modeling.

3. Absolute Moment Block Truncation Coding (AMBTC)

The AMBTC scheme is not limited to image compression but has been used in different applications. At present, besides image compression [21,22] and video compression [23,24], AMBTC is used in various fields like steganography [25]-[29], Image authentication [30,31], data embedding [32,33], image hashing [34,35], tamper detection and recovery [36]-[40].

In AMBTC, the input image will be segmented into several square blocks with the size $(m \times m)$, where m is 4, 8, or 16. For each block, the mean σ_j is computed as

$$\sigma_j = \frac{1}{m^2} \sum_{i=1}^{m^2} P_i \quad (1)$$

where P_i is the i-th pixel in the block. After that, the block pixels will be grouped into two groups, and the two quantization levels, the lower mean L_j and the higher mean H_j, will be computed using

$$L_j = \frac{1}{m^2 - k_j} \sum_{P_i < \sigma_j}^{m^2} P_i \tag{2}$$

and

$$H_j = \frac{1}{k_j} \sum_{P_i \geq \sigma_j}^{m^2} P_i \tag{3}$$

where k_j is the number of pixels that equals or exceeds the mean value σ_j.

Finally, the bit-plane (Bitmap) blocks will be created, where each bit-plane pixel is represented by one bit only '0' or '1'. The following two rules are used to generate the bit-plane block.

- If $P_i \geq \sigma_j$, the bit-plane pixel will set to '1'.
- Otherwise, the bit-plane pixel will set to '0'.

In decoding, the compressed image is reconstructed using the two quantization levels, L_j and H_j along with the bit-plane blocks, where '0's in the bit-plane will be replaced by L_j and '1's will be replaced by H_j.
The followings points summarize the AMBTC encoding,

1- Divide the input images to non-overlapping blocks with $m \times m$ (typically $m = 4$)
2- For each block, calculate the mean of pixels σ using equation 1.
3- For each block, calculate the lower mean L and the higher mean H using equations 2 and 3 respectively.
4- For each block, generate the bit-plane block, by setting the value '0' for pixels less than the block mean σ and setting the value '1' to pixels equal or exceed the block mean σ.
5- The compressed file consists of the block codes, where each block code is represented by the lower mean L and the higher mean H as well as the bit-plane block.

The AMBTC decoding phase is summarized by the following,

1- Extract the block codes from the compressed file.
2- For each bit-plane block, reconstruct the block by replacing '0's with the lower mean L value, and replacing '1's with the higher mean H value.
3- Rearrange the reconstructed blocks to reconstruct the image.

4. The Proposed Method

As mentioned before, AMBTC is efficient in terms of image quality, but its bitrate is high, and previously an integration of artificial neural networks with AMBTC was made to reduce the bitrate further [7]. After AMBTC encoding the main mean matrix was fed to the neural networks as input, while the lower mean and higher mean matrices were fed as targets; the neural networks train on these data sets to be able to predict the approximated lower and higher means matrices that will be used in decoding phase to reconstruct the compressed image. This work managed to lower the bitrate because it represents each block code with mean value and the bit plane block rather than representing the block code by lower mean, higher mean, and the bit plane block as in the conventional AMBTC. The bitrate reduction was on the expense of the reconstructed image quality.

The purpose of this work is to preserve AMBTC reconstructed image quality and minimize the bitrate. The proposed scheme uses AMBTC with block classification and neural networks (BC-AMBTC-ANN) as they are good tools for prediction, regression and function approximation problems due to their ability to train on data sets.

The block diagram of the proposed method is shown in Fig.1. The first step involves applying AMBTC encoding to the input image to create the mean σ, lower mean L, higher mean H matrices, and the bit-plane (bitmaps) blocks. The second step is classifying the blocks into smooth and complex blocks according to the difference between the block's higher mean and lower mean. A 1-bit indicator $\boldsymbol{ind}$ will be used to distinguish the block type; if $\boldsymbol{ind} = 0$, the block is smooth; if $\boldsymbol{ind} = 1$, the block is complex. The smooth block parameters will be fed to the neural network system for training; the mean matrix σ will be provided as inputs, while the lower mean and the higher mean matrices (L, H) will be fed as targets. After training, the neural network's output will be L' and H' matrices, which are equivalent to L and H matrices, respectively.

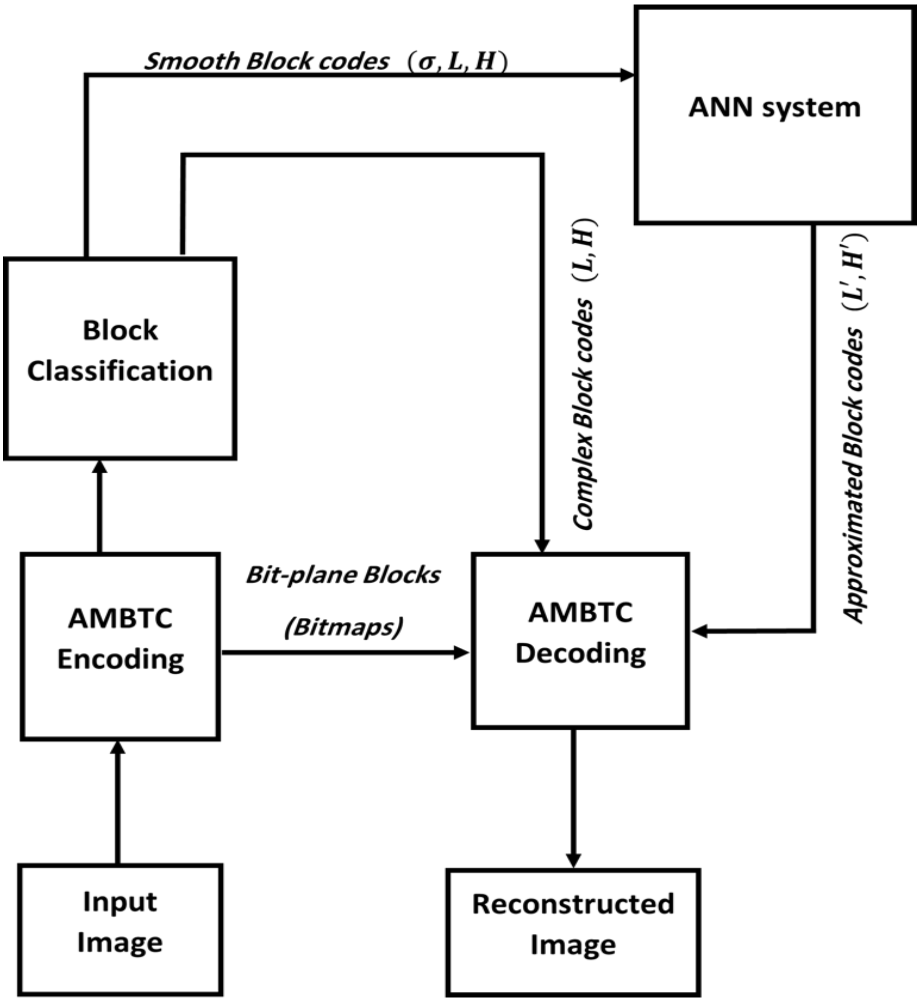


Fig. 1 *The proposed block diagram of the BC-AMBTC-ANN*

The compressed image will be constructed in the decoding stage by reconstructing all blocks using their parameters. So, the mean matrix σ of the smooth blocks will be extracted, and the neural network will make a prediction process, using the σ matrix only, to generate the approximated quantization values L' and H'; from the other hand, the complex blocks will be decoded using their L and H values. The following subsections demonstrate the steps of the proposed scheme.

4.1 Encoding

The conventional AMBTC encoding process is applied to the input image; so, it will be segmented into 4×4 non-overlapping blocks. For each block, the mean σ, the lower mean L, and the higher mean H are generated using (1), (2), and (3), respectively, as well as the bit-plane block.

4.2 Block Classification

In this step, the image blocks will be classified into two classes, smooth and complex, using the difference between H and L as follows:

if $\boldsymbol{H} - \boldsymbol{L} \leq T$;
    $\boldsymbol{ind} = 0$; "The block is smooth."
else;
    $\boldsymbol{ind} = 1$; "The block is complex."

where T is a predefined threshold. So, the block codes of smooth blocks will be represented by $\boldsymbol{ind} = 0$ and the mean value $\boldsymbol{\sigma}$, while the block codes of the complex blocks will be defined by $\boldsymbol{ind} = 1$, and the two quantization levels L and H.

4.3 ANN System Training

A system of two neural networks will be used in this work, where the training process will be done for the smooth blocks parameters, i.e., the mean σ matrix and the two quantization values L and H . The mean matrix will be fed to both neural networks as input; the target of the first neural network is L matrix, while the target of the second neural network is H matrix. The two neural networks have the same architecture, each consisting of three layers: an input layer with one input, a hidden layer comprising ten cells, and one output layer with a single neuron representing the output. A normalization process was applied to convert data values in the range $[0-1]$. For training, the Levenberg-Marquardt algorithm is used; where the weights and biases are generated randomly, and the activation functions of hidden units are the sigmoid, while the activation function of output is linear.

4.4 Decoding

In decoding, the image will be reconstructed by reconstructing all image blocks using the quantization values of each block and the bit-plane blocks. But before that, the quantization levels of smooth blocks should be generated. So, the block codes of smooth blocks, represented by mean values σ, will be extracted utilizing the 1-bit indicator $\boldsymbol{ind}$, and a prediction is made to the trained neural networks system to get the approximated quantization levels L' and H'. The complex blocks already have their quantization levels, L and H. In this way, all quantization levels for smooth and complex blocks are ready for image reconstruction as in conventional AMBTC.

5. Results and Discussion

This study selected six grayscale images with a size of 512×512 pixels to evaluate the performance of the proposed scheme: “Lenna”, “F-16”, “Boats”, “Pepper”, “Goldhill”, and “Sailboat” as shown in Fig.2.

Fig. 2 *The original grayscale testing images of* $\mathbf{512 \times 512}$ *pixels (a)Lenna;(b) F-16;(c) Boats; (d)Pepper; (e)Goldhill; (f)Sailboat*

Int. Journal of Integrated Engineering Vol. 16 No. 3 (2024) p. 290-301

The experimental results were obtained using MATLAB neural network tools, and the performance of the proposed method is evaluated using two key metrics: Peak Signal-to-Noise Ratio (PSNR) and Bitrate (BR).
The PSNR value is calculated as follows:

$$PSNR = 20 \times \log 10(\frac{255}{\sqrt{MSE}}) \tag{4}$$

Where MSE represents the mean squared error that is given by the following equation:

$$MSE = \frac{1}{m \times n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (f(i,j) - g(i,j))^2 \tag{5}$$

Where f represents the matrix data of the original image.
g *mean*s the matrix data of the reconstructed image.
m, n represent the width and the height, respectively.

The bitrate is computed as follows:

$$BR = \frac{Total\ number\ of\ bits\ of\ the\ compressed\ file}{Image\ size\ in\ Bytes} \tag{6}$$

Table 2 *Comparative results of AMBTC, MGAMBTC, IMGAMBTC, AMBTC-ANN, and the proposed scheme*

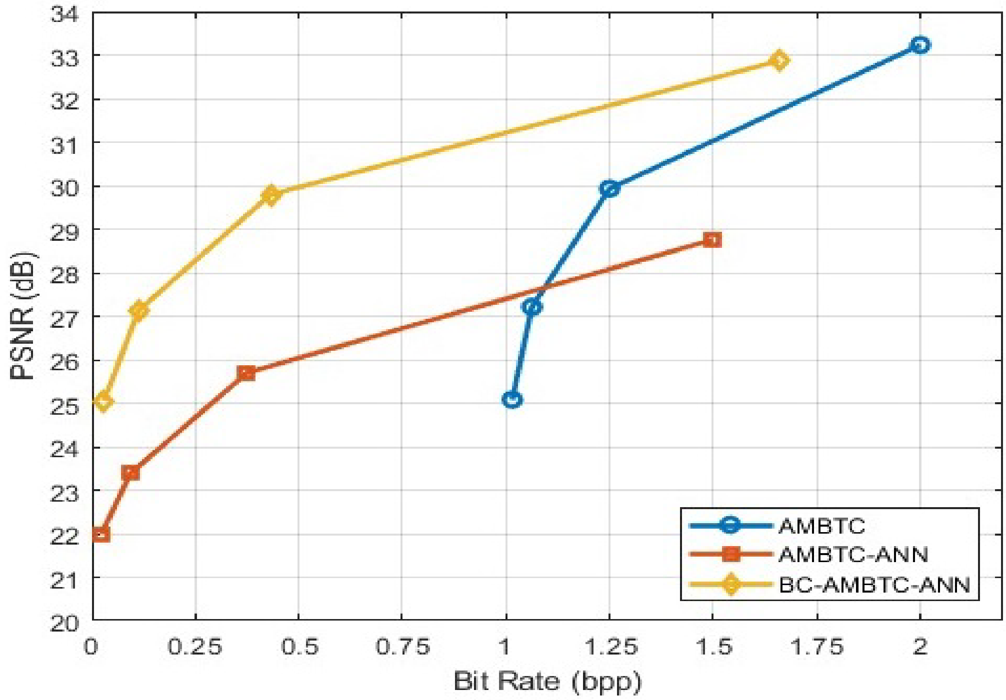
Method	AMBTC		MGAMBTC [12]		iMGAMBTC [13]		AMBTC-ANN [7]		BC-AMBTC-ANN	
Image	PSNR	BR	PSNR	BR	PSNR	BR	PSNR	BR	PSNR	BR
Lenna	33.23	2	38.6	1.8	38.8	2	28.76	1.5	**32.87**	**1.66**
F-16	31.97	2	38.9	1.7	38.9	1.6	27.22	1.5	**31.80**	**1.67**
Boats	31.97	2	37.9	1.9	37.9	1.9	28.03	1.5	**31.73**	**1.69**
Pepper	33.70	2	38.7	1.8	39.1	2	25.57	1.5	**33.31**	**1.65**
Goldhill	32.63	2	39.1	2.4	39.5	2.7	29.09	1.5	**32.32**	**1.70**
Sailboat	29.87	2	36.9	2.5	37.2	2.7	26.77	1.5	**29.77**	**1.73**
Average	**32.22**	**2**	**38.35**	**2.016**	**38.56**	**2.15**	**27.57**	**1.5**	**31.96**	**1.68**

Table 2 presents the experimental results with a block size of 4×4 pixels. The comparison shows that the methods by Xiang et al. [12] and Chuang et al. [13] significantly improve image quality, achieving the highest PSNR compared to our proposed method. However, their average bitrates exceed 2 bpp, and for images with complex details like Goldhill and Sailboat, their bitrates are nearly 1 bpp higher than our model.

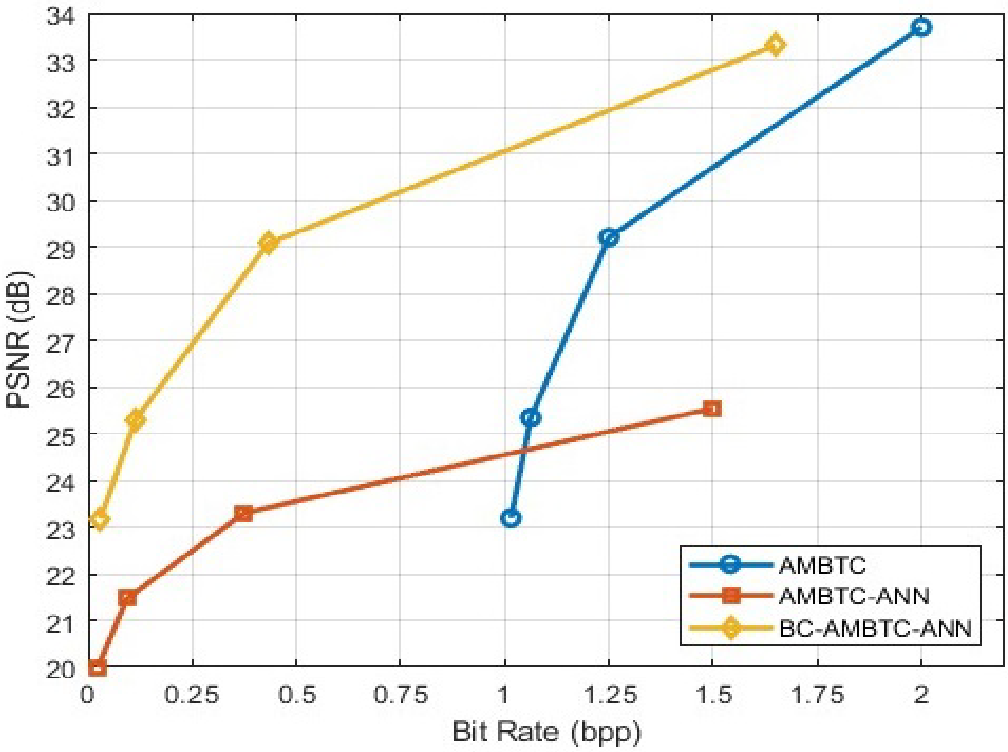
The previously proposed AMBTC-ANN [7] reduces the bitrate to 1.5 bpp by retaining only the mean value and the bit-plane for each block. Despite this, it yields a much lower PSNR compared to the original AMBTC, resulting in poor reconstructed image quality. This is due to the neural network's approximation of quantization levels, L' and H', which differ significantly from the actual L and H values, introducing distortions.

In our experiments, we found that a threshold value of $T = 19$ was optimal for classifying image blocks into smooth and complex categories. The results indicate that the bitrate (BR) of our method is lower than the average bitrate of AMBTC, MGAMBTC, and iMGAMBTC by 0.32 bpp, 0.336 bpp, and 0.47 bpp, respectively. Moreover, it achieved PSNR values close to those of AMBTC, demonstrating a balanced performance in both compression efficiency and image quality.The incorporating of the block classification technique to the previous work in [7], provides the ability to determine which image blocks are smooth and can be represented in reduced code; mean value only with the bit plane, and which blocks are complex and need to be preserved by representing them in the original AMBTC block code, i.e. lower mean, higher mean and the bit-plane.

Since this work is proposed and designed according to original AMBTC and AMBTC-ANN, and to ensure a fair comparison, the rate-distortion curves were generated for the three forementioned schemes at different block sizes, including $4 \times 4, 8 \times 8, 16 \times 16$, and 32×32. This allows for a comprehensive evaluation of the proposed method's performance relative to AMBTC and AMBTC-ANN, demonstrating its effectiveness in improving compression efficiency while maintaining image quality across various block sizes. In this paper, we present the result of two images; Lenna and Pepper; to investigate the performance of the image compression technique as shown in Fig.3.



(a) Lenna bitrate-distortion



(b) Pepper bitrate-distortion

Fig. 3 *Bitrate-Distortion Curves for (a) Lenna; (b) Pepper*

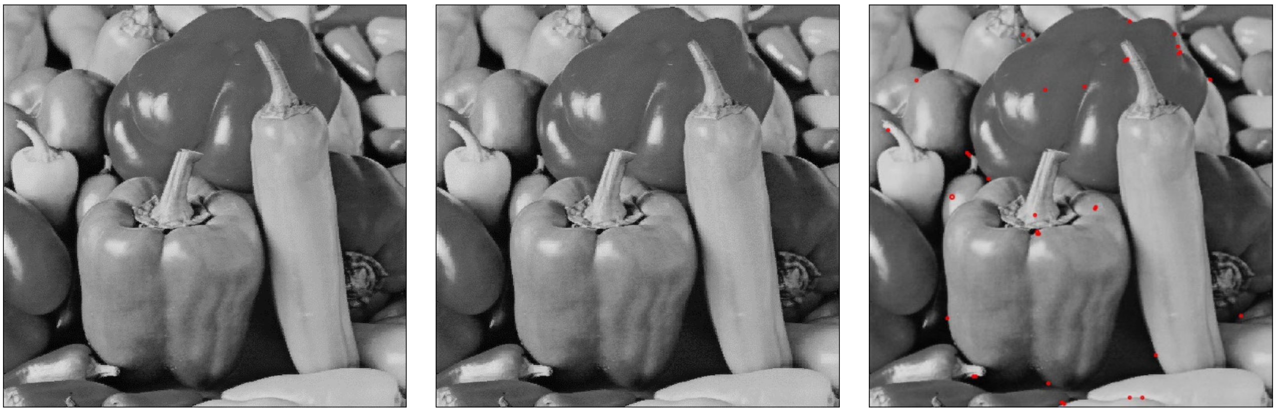
As can be seen from Fig.3., the proposed scheme BC-AMBTC-ANN strikes a balance between image quality and compression efficiency. In the comparison of Bitrate-Distortion performance for Lenna and Pepper images, BC-

AMBTC-ANN consistently outperforms AMBTC and AMBTC-ANN across all bitrates. For the Lenna image, BC-AMBTC-ANN achieves a PSNR of approximately 33 dB at 2 bpp, compared to around 32 dB for AMBTC and 24 dB for AMBTC-ANN. Similarly, for the Pepper image, BC-AMBTC-ANN reaches a PSNR of 32 dB at 2 bpp, while AMBTC achieves 31 dB and AMBTC-ANN remains at 24 dB.

These results highlight the robustness of BC-AMBTC-ANN, which not only offers higher quality at lower bitrates but also maintains a significant advantage as bitrate increases. The AMBTC method shows improvement at higher bitrates, reaching a PSNR of 31-32 dB, but it lags behind BC-AMBTC-ANN at lower bitrates. AMBTC-ANN, with PSNR values consistently around 24-29 dB, proves to be the least effective across both images, making it less suitable for applications requiring high-quality compression.

For accurate visual inspection, Fig.4. displays the reconstructed images of Lenna and Pepper, Fig.4. (a) is the AMBTC reconstructed image of Lenna, Fig.4. (b) is the proposed method BC-AMBTC-ANN reconstructed image of Lenna, Fig.4. (d) is the AMBTC reconstructed image for Pepper, and Fig.4. (e) is the proposed method BC-AMBTC-ANN reconstructed image for Pepper. It can be seen that both images are almost identical, with only minor differences that are not immediately noticeable to the naked eye.

(a) AMBTC Lenna **(b) Proposed method Lenna** **(c) Lenna - Annotated difference image**



(d) AMBTC Pepper **(e) Proposed method Pepper** **(f) Pepper- Annotated difference image**

Fig. 4 *Reconstructed images and annotated difference images for Lenna and pepper and pepper*

To accurately identify and highlight these subtle discrepancies and by neglecting the difference values less than 10, annotated difference images were generated, which are shown in Fig.4. (c) for Lenna and Fig.4. (f) for Pepper; these images' purpose is to reveal the variations between the two reconstructed images of AMBTC and the proposed scheme BC-AMBTC-ANN. It can be observed that the discrepancies, visible as tiny scattered red circles in the images, are minimal and confined to specific areas. This indicates that reconstructed images of AMBTC and the proposed scheme share a high degree of similarity, with only a few pixels differing slightly. The highlighted areas in the annotated difference images provide a clear visualization of where these small differences exist, offering valuable insight into the degree of similarity between the two reconstructed images.

Based on the preceding discussion and analysis, it can be concluded that the proposed scheme BC-AMBTC-ANN achieves comparable performance to the conventional AMBTC while utilizing a lower bitrate and demonstrating enhanced reliability. Consequently, it provides a more advantageous balance between image quality and compression efficiency, making it a preferable option for applications where both high image quality and efficient compression are crucial.

6. Conclusions

This study presents a new method for grayscale image compression using block-classification-based AMBTC with neural networks. The conventional AMBTC is applied to the image to generate the mean matrix, the block codes (lower and higher mean values), and the bit-planes. The blocks will be classified into smooth and complex blocks utilizing the difference between the higher and lower means. The role of the neural network is to predict the approximated quantization levels of the smooth blocks for decoding. In contrast, complex blocks will be decoded by their original quantization levels. The experimental results prove that the proposed method minimizes the bitrate and preserves the image quality. For future works, this approach can be expanded by trying another type of neural network with different architectures and parameters; it can also be extended by applying it to compress the color images.

Acknowledgement

This work was supported by the Ministry of Higher Education Malaysia under Fundamental Research Grant Scheme (FRGS/1/2021/ICT09/UTM/02/1) and the output of ASEAN IVO project (http://www.nict.go.jp/en/asean_ivo /index.html) entitled "Agricultural IoT based on Edge Computing" (S.K130000.7656.4X796).

Conflict of Interest

The authors would like to declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

The authors confirm contribution to the paper as follows: ***study conception and design:*** *Aqeel Noori, Norulhusna Ahmad, Siti Armiza Mohd Aris, Norliza Mohd Noor;* ***data collection:*** *Aqeel Noori;* ***analysis and interpretation of results:*** *Aqeel Noori, Norulhusna Ahmad and Norliza Mohd Noor;* ***draft manuscript preparation:*** *Aqeel Noori and Norulhusna Ahmad. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] Lema, M., & Mitchell, O. (1984). Absolute moment block truncation coding and its application to color images. *IEEE Transactions on communications*, *32*(10), 1148-1157. https://doi.org/10.1109/TCOM.1984.1095973.

[2] Hu, Y. C., Liu, J. S., Lo, C. C., Wu, C. M., & Chen, Y. (2022). Grayscale image coding using optimal pixel grouping and adaptive multi-grouping division block truncation coding. Multimedia Tools and Applications, 81(13), 17937-17958. https://doi.org/10.1007/s11042-022-12680-y.

[3] Hu, Y. C. (2003). Low-complexity and low-bit-rate image compression scheme based on absolute moment block truncation coding. *Optical Engineering*, *42*(7), 1964-1975. https://doi.org/10.1117/1.1576776.

[4] Hu, Y. C. (2003). Improved moment preserving block truncation coding for image compression. Electronics Letters, 39(19), 1. https://doi.org/10.1049/el:20030884.

[5] Hu, Y. C. (2004). Predictive moment preserving block truncation coding for gray-level image compression. Journal of Electronic Imaging, 13(4), 871-877. https://doi.org/10.1117/1.1785158.

[6] Olsen, S. I. (2000). Block truncation and planar image coding. Pattern Recognition Letters, 21(13-14), 1141-1148. https://doi.org/10.1016/S0167-8655(00)00075-1.

[7] Ali, A. N. M., Ahmad, N., Noor, N. M., & Aris, S. A. M. (2022, November). Image Compression Using AMBTC with Artificial Neural Networks. In 2022 IEEE Symposium on Future Telecommunication Technologies (SOFTT) (pp. 78-82). IEEE. https://doi.org/10.1109/SOFTT56880.2022.10009930.

[8] Hu, Y. C., Liu, Y. H., & Chang, I. C. (2018, April). Color image coding based on block truncation coding using quadtree segmentation. In 2018 3rd International Conference on Computer and Communication Systems (ICCCS) (pp. 196-200). IEEE. https://doi.org/10.1109/CCOMS.2018.8463236.

[9] Hui, Z., & Zhou, Q. (2018, November). An Image Compression Scheme Based on Block Truncation Coding Using Real-time Block Classification and Modified Threshold for Pixels Grouping. In 2018 Eighth International Conference on Image Processing Theory, Tools and Applications (IPTA) (pp. 1-5). IEEE. https://doi.org/10.1109/IPTA.2018.8608124.

[10] Kumar, R., Singh, S., & Jung, K. H. (2019, March). Human visual system based enhanced AMBTC for color image compression using interpolation. In 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN) (pp. 903-907). IEEE. https://doi.org/10.1109/SPIN.2019.8711635.

[11] Ahmad, N., Jaffery, Z. A., & Irshad. (2019). An Approach to Color Image Coding Based on Adaptive Multilevel Block Truncation Coding. In Applications of Artificial Intelligence Techniques in Engineering: SIGMA 2018, Volume 2 (pp. 597-606). Springer Singapore. https://doi.org/10.1007/978-981-13-1822-1_56.

[12] Xiang, Z., Hu, Y. C., Yao, H., & Qin, C. (2019). Adaptive and dynamic multi-grouping scheme for absolute moment block truncation coding. Multimedia Tools and Applications, 78, 7895-7909. https://doi.org/10.1007/s11042-018-6030-5.

[13] Chuang, J. C., Hu, Y. C., Chen, C. M., & Yin, Z. (2020). Adaptive grayscale image coding scheme based on dynamic multi-grouping absolute moment block truncation coding. Multimedia Tools and Applications, 79, 28189-28205. https://doi.org/10.1007/s11042-020-09325-3.

[14] Chen, T. S., Wu, J., Chen, K. S., Yuan, J., & Hong, W. (2021). Hybrid encoding scheme for AMBTC compressed images using ternary representation technique. Applied Sciences, 11(2), 619. https://doi.org/10.3390/app11020619.

[15] Kumar, R., Kumar, N., & Jung, K. H. (2022). Enhanced interpolation-based AMBTC image compression using Weber's law. Multimedia Tools and Applications, 81(15), 20817-20828. https://doi.org/10.1007/s11042-022-12634-4.

[16] Rani, M. L. P., Rao, G. S., & Rao, B. P. (2019, March). Performance analysis of compression techniques using LM algorithm and SVD for medical images. In 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN) (pp. 654-659). IEEE. https://doi.org/10.1109/SPIN.2019.8711601.

[17] Sasibhushana Rao, G., Vimala Kumari, G., & Prabhakara Rao, B. (2019). Image compression using neural network for biomedical applications. In *Soft Computing for Problem Solving: SocProS 2017, Volume 2* (pp. 107-119). Springer Singapore. https://doi.org/10.1007/978-981-13-1595-4_9.

[18] Suma. (2022). Modelling of Efficient Medical Image Compression System Based on Joint Operation of DWT and Neural Network. *Wireless Personal Communications*, *124*(4), 3129-3143. https://doi.org/10.1007/s11277-022-09505-4.

[19] Alshehri, S. A. (2016). Neural network technique for image compression. IET Image Processing, 10(3), 222-226. https://doi.org/10.1049/iet-ipr.2014.1039.

[20] Chakib, H., Minaoui, B., Salhi, A., & Badi, I. (2018). Switching of Wavelet Transforms by Neural Network for Image Compression. Journal of Electronic Commerce in Organizations (JECO), 16(1), 43-56. https://doi.org/10.4018/JECO.2018010104.

[21] Al-Azawi, S., Boussakta, S., & Yakovlev, A. (2011, May). Low complexity image compression algorithm using AMBTC and bit plane squeezing. In International Workshop on Systems, Signal Processing and their Applications, WOSSPA (pp. 131-134). IEEE. https://doi.org/10.1109/WOSSPA.2011.5931432.

[22] Anil, D., Karthik, K. V., & Kumar, K. S. (2011, June). A modified three level block truncation coding for image compression. In *2011 International Conference on Pattern Analysis and Intelligence Robotics* (Vol. 1, pp. 31-35). IEEE. https://doi.org/10.1109/ICPAIR.2011.5976907.

[23] Al-Asmari, A. K., Dave, S., & Kwatra, S. C. (1999, October). Low complexity video compression algorithm using AMBTC. In *MILCOM 1999. IEEE Military Communications. Conference Proceedings (Cat. No. 99CH36341)* (Vol. 2, pp. 1241-1245). IEEE. https://doi.org/10.1109/MILCOM.1999.821402.

[24] Vijayanagar, K. R., & Kim, J. (2012, October). Compression of residual layers of layered depth video using hierarchical block truncation coding. In 2012 3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON) (pp. 1-4). IEEE. https://doi.org/10.1109/3DTV.2012.6365476.

[25] Chang, C. C., Chen, T. S., Wang, Y. K., & Liu, Y. (2018). A reversible data hiding scheme based on absolute moment block truncation coding compression using exclusive OR operator. Multimedia Tools and Applications, 77, 9039-9053. https://doi.org/10.1007/s11042-017-4800-0.

[26] Hui, Z., & Zhou, Q. (2020). A reversible data hiding method for AMBTC compressed image without expansion inside stego format. KSII Transactions on Internet and Information Systems (TIIS), 14(11), 4443-4462. https://doi.org/10.3837/tiis.2020.11.011.

[27] Kumar, N., Kumar, R., Malik, A., & Singh, S. (2022). Low bandwidth data hiding for multimedia systems based on bit redundancy. Multimedia Tools and Applications, 1-19. https://doi.org/10.1007/s11042-021-10832-0.

[28] Kumar, R., Kim, D. S., & Jung, K. H. (2019). Enhanced AMBTC based data hiding method using hamming distance and pixel value differencing. Journal of Information Security and Applications, 47, 94-103. https://doi.org/10.1016/j.jisa.2019.04.007.

[29] Yu, Z., Lin, C. C., Chang, C. C., & Su, G. D. (2019). HBF-DH: An enhanced payload hybrid data hiding method based on a hybrid strategy and block features. IEEE Access, 7, 148439-148452. http://doi.org/10.1109/ACCESS.2019.2943505.

[30] Chen, T., Zhou, X., Chen, R., Hong, W., & Chen, K. (2021). A high fidelity authentication scheme for AMBTC compressed image using reference table encoding. Mathematics, 9(20), 2610. https://doi.org/10.3390/math9202610.

[31] Zhou, X., Chen, J., Yang, G., Lin, Z. F., & Hong, W. (2023). An authentication method for AMBTC compressed images using dual embedding strategies. Applied Sciences, 13(3), 1402. https://doi.org/10.3390/app13031402.
[32] Lai, P. C., Shen, J. J., & Chou, Y. C. (2023). High Embedding Capacity Data Hiding Technique Based on Hybrid AMBTC and LSB Substitutions. Int. J. Netw. Secur, 25, 221-234. https://doi.org/10.6633/IJNS.202303_25(2).05.
[33] Bhardwaj, R., & Niranjan, A. (2023). An improved dual image separable reversible data hiding algorithm for encrypted HAMBTC compressed images. Multimedia tools and applications, 82(3), 3335-3362. https://doi.org/10.1007/s11042-022-13209-z.
[34] Qin, C., Liu, E., Feng, G., & Zhang, X. (2020). Perceptual image hashing for content authentication based on convolutional neural network with multiple constraints. *IEEE Transactions on Circuits and Systems for Video Technology*, *31*(11), 4523-4537. https://doi.org/10.1109/TCSVT.2020.3047142.
[35] Abbas, S. Q., Ahmed, F., & Chen, Y. P. P. (2021). Perceptual image hashing using transform domain noise resistant local binary pattern. Multimedia Tools and Applications, 80(7), 9849-9875. https://doi.org/10.1007/s11042-020-10135-w.
[36] Kim, C., Shin, D., & Yang, C. N. (2018). Self-embedding fragile watermarking scheme to restoration of a tampered image using AMBTC. Personal and ubiquitous computing, 22, 11-22. https://doi.org/10.1007/s00779-017-1061-x.
[37] Qin, C., Ji, P., Chang, C. C., Dong, J., & Sun, X. (2018). Non-uniform watermark sharing based on optimal iterative BTC for image tampering recovery. IEEE MultiMedia, 25(3), 36-48. https://doi.org/10.1109/MMUL.2018.112142509.
[38] Kim, C., & Yang, C. N. (2021). Self-embedding fragile watermarking scheme to detect image tampering using AMBTC and OPAP approaches. Applied Sciences, 11(3), 1146. https://doi.org/10.3390/app11031146.
[39] Jana, M., Jana, B., & Joardar, S. (2022). Local feature based self-embedding fragile watermarking scheme for tampered detection and recovery utilizing AMBTC with fuzzy logic. Journal of King Saud University-Computer and Information Sciences, 34(10), 9822-9835. https://doi.org/10.1016/j.jksuci.2021.12.011.
[40] Lin, C. C., Lee, T. L., Chang, Y. F., Shiu, P. F., & Zhang, B. (2023). Fragile watermarking for tamper localization and self-recovery based on AMBTC and VQ. Electronics, 12(2), 415. https://doi.org/10.3390/electronics12020415.