United International Journal for Research & Technology
Volume 04, Issue 08, 2023 | Open Access | ISSN: 2582-6832

Enhancement of the Huffman Algorithm with Discrete Wavelet Transform Applied to Lossless Image Compression

John Dylan R. Aranzado[1], Gian Karlo Barbosa[2], Karl Francis Linget[3], and Vivien A. Agustin[4]
[1,2,3]Student, Computer Science Department - Pamantasan ng Lungsod ng Maynila
[4]Professor, College of Engineering - Pamantasan ng Lungsod ng Maynila

Abstract— This study introduces an enhanced approach for compressing and decompressing grayscale images using an advanced version of the Huffman coding algorithm. Instead of relying on the basic algorithm, the researchers employed the Discrete Wavelet Transform to decompose the image into distinct sets of signals. These signals were then converted into a bit stream using an improved version of Huffman coding, achieving further compression. The proposed algorithm outperforms other techniques in terms of compression ratio, compression time, and bits per pixel. This study provides a starting point for future researchers to optimize and enhance the algorithm, benefiting various fields such as digital imaging and data storage.

Keywords— Huffman algorithm, Discrete Wavelet Transform, enhancement, lossless image compression, Hybrid Algorithm.

I. INTRODUCTION

A. Background of the Study

Data compression, including image compression, aims to reduce redundancy and irrelevant information in order to save space and improve data storage or transmission efficiency. There are two types of image compression. Lossy image compression prioritizes higher compression ratios at the cost of sacrificing data accuracy, and lossless image compression guarantees complete recovery of the original data.

Huffman coding is a widely used algorithm for lossless image compression. It analyzes pixel value frequencies and assigns shorter codes to common symbols and longer codes to rare symbols, creating a Huffman tree. However, traditional Huffman coding requires the decoder to traverse the entire tree, which can be inconvenient. While effective for images with limited color palettes, it may not significantly compress true-color images. Hybrid compression techniques combine Huffman coding with other methods for improved compression ratios.

The Discrete Wavelet Transform (DWT) is commonly used for signal and image analysis. It breaks down the data into approximation and detail coefficients using filters, capturing localized and overall variations. DWT enables efficient representation and compression by hierarchically decomposing the data. It finds applications in compression, noise removal, and feature extraction. Although it preserves important details, it can introduce blocking artifacts and has higher computational complexity compared to simpler transforms.

In this study, researchers propose an enhanced compression technique by combining an improved Huffman algorithm with DWT for decompression. The enhanced Huffman algorithm is expected to achieve higher compression ratios and faster compression times compared to the conventional algorithm. DWT, chosen for its efficiency and optimized implementations, speeds up the decompression process. The combination of these algorithms is expected to yield better image compression results compared to previous research.

B. Statement of the Problem

The conventional Huffman algorithm was used in data compression as well as the algorithm in compressing an image. The primary drawback of the traditional Huffman algorithm lies in its utilization of variable-length codes for symbol representation, determined by their occurrence frequency. While this technique enables efficient compression for frequently occurring symbols, it can lead to lengthier codes for less common symbols. This shortcoming is commonly referred to as "Huffman coding inefficiency." The statements below highlight the issues that are relevant to the researchers' study:

1. Compression speed is not efficient. - Compression speed is not efficient. - Complexity of the existing algorithm is not efficient enough, hence the slow compression speed [7].

2. Conventional Huffman Compression consumes more space upon compression. - Conventional Huffman Coding takes up more space and is more time-consuming; the decoder must navigate through the complete Huffman tree to interpret the code [10].

3. The existing algorithm doesn't have a clear decoding process. - In the conventional method, the typical procedure entails following the path from the main node to the terminal node and examining the input Huffman stream bit by bit. During this process, the left branch corresponds to '1' while the right branch corresponds to '0' [6].

C. Objective of the Study

The researchers aim to enhance the Huffman algorithm through a thorough analysis of the identified issues and an exploration of various alternative approaches. Their objective is to discover potential solutions that can effectively optimize the algorithm for their specific intended purpose. In particular, this paper aims to:

1. To further enhance the existing algorithm's compression speed and efficiency by enhancing the compression process of its algorithm and replacing a better decompression technique.
2. To reduce the inefficiency in storing or transmitting image data arises from the presence of irrelevant or redundant information.
3. To decrease the quantity of bits necessary for representing an image.

II. REVIEW OF RELATED LITERATURE

In the realm of lossless image compression, researchers worldwide have made significant strides in developing innovative techniques that achieve efficient compression while preserving image quality. One promising approach that has gained attention is the enhancement of the Huffman algorithm through the integration of the discrete wavelet transform (DWT). This review aims to provide an in-depth analysis of the existing foreign literature on this topic, incorporating key contributions from international researchers and their impact on the field.

Researchers from various countries have explored the integration of the Huffman algorithm and DWT to enhance lossless image compression. For instance, a group of researchers from the United States, proposed a hybrid algorithm that combines the strengths of both techniques [12].

By leveraging the DWT to transform the image into its frequency domain representation and subsequently employing Huffman coding to encode the transformed coefficients, the authors achieved remarkable compression ratios without compromising image quality. Their findings demonstrated the potential of this integrated approach for efficient lossless image compression.

Researchers across different countries have contributed to the refinement of encoding strategies used in the enhancement of the Huffman algorithm with DWT. In a study conducted in China, a refined encoding strategy based on the statistical properties of wavelet coefficients obtained through DWT was introduced [5]. By exploiting the inherent correlations among these coefficients, the authors optimized the Huffman coding process and significantly improved compression performance. Their work shed light on the importance of incorporating advanced encoding techniques to maximize the benefits of the integrated approach.

International researchers have also investigated the use of multi-level DWT in conjunction with the Huffman algorithm for lossless image compression. Wang and Chen from Taiwan explored the application of multiple levels of DWT followed by Huffman coding at each level [17]. This approach allowed for higher compression ratios while preserving image fidelity. Their research highlighted the effectiveness of utilizing multi-resolution analysis in combination with Huffman coding, offering a potential solution for enhancing compression efficiency.

In addition to the contributions, foreign literature provides valuable insights into the enhancement of the Huffman algorithm with DWT for lossless image compression. Researchers from Germany, for example, conducted a study on the impact of different wavelet families on compression performance. Their findings suggested that certain wavelet families, such as Daubechies and Symlets, exhibit superior performance in terms of achieving higher compression ratios while preserving image quality [1].

Furthermore, researchers from Japan focused on applying the enhanced Huffman algorithm with DWT to specific types of images, such as medical and satellite imagery. Their findings indicated that the integrated approach can effectively compress these types of images while maintaining the diagnostic value and visual clarity necessary for medical interpretation or remote sensing applications.

In the subject of picture compression, the Huffman coding technique and wavelet transform have been thoroughly investigated. To investigate the efficacy, performance, and uses of these procedures, researchers from many nations have carried out investigations. This review offers a summary of several significant international studies in this field.

Smith et al. conducted a comprehensive international study on the Huffman coding technique and wavelet transform for image compression [13]. Their research centered on examining how different wavelet families and encoding techniques affected compression efficiency. This combination has the potential to provide effective image compression while preserving respectable image quality, according to the study.

Lee et al. compared the effectiveness of discrete wavelet transform (DWT) and Huffman coding for picture compression [4]. In terms of compression ratios, image quality, and computing complexity, their study attempted to evaluate how well these strategies performed. They gave insights into the benefits and drawbacks of DWT and Huffman coding for use in picture compression applications through their assessment.

In a different study, Huffman coding and wavelet transform were suggested as the ideal encoding approach for compressing images [15]. Their approach is intended to improve encoding efficiency through compression efficiency. The study made clear how crucial it is to take encoding approaches into account in order to improve compression performance.

Spanish researchers Garcia and Santos conducted a study on the usage of wavelet transform and Huffman coding for video compression [2]. By considering the temporal and spatial correlations in video frames, their research intended to accomplish effective compression while maintaining video quality.

Zhao et al. carried out a study in China that compressed satellite pictures using the wavelet transform and the Huffman coding algorithm [19]. For proper remote sensing analysis and interpretation, their work focused on maintaining important details and information in compressed images.

Another significant international study examined the use of Huffman coding and wavelet transform for safeguarding picture transmission across networks by Indian researchers [9]. Their work combined compression and encryption methods to fulfill the demand for secure and effective image transmission. The significance of including security precautions in picture compressing approaches was underlined by this study.

Foreign studies have also investigated how various wavelet families affect compression efficiency. Three German academics, for instance, investigated the effects of wavelet families like Daubechies, Symlets, and Coiflets on the Huffman coding algorithm with wavelet transform for image compression [11]. Their research revealed useful information for choosing the best wavelet families to maintain image quality while getting the best compression ratios.

As Subramanya suggests, many image compression standards adopt a hybrid approach by combining various techniques [14]. In the case of bi-tonal images, prominent standards such as Group 3, Group 4, and JBIG (Joint Bi-level Image Group) are commonly employed. On the other hand, for continuous-tone images, the widely used standard is JPEG (Joint Photographic Experts Group). Bi-tonal image compression methods are particularly popular in applications like digital facsimile (FAX) where efficient encoding of black and white images is essential.

An image compression system is divided into two main components [3]: the compressor and the decompression process. The compressor is composed of two stages: preprocessing and encoding. On the other hand, the decompression process consists of two stages: decoding and post-processing.

Huffman's contribution to data representation techniques has been a significant development in computer science, with his paper from 1952 receiving over 7,500 citations and influencing the compression and coding regimes in use today. Huffman coding is a lossless image compression algorithm based on entropy, which is still an effective technique for image compression.

UIJRT ISSN: 2582-6832

United International Journal for Research & Technology

Volume 04, Issue 08, 2023 | Open Access | ISSN: 2582-6832

Variable length coding, such as Huffman coding, is commonly employed to enhance coding efficiency. By utilizing the probability distribution of a data source, the Huffman coding algorithm generates unique and decodable codes with the shortest average code-word length. This algorithm follows a systematic approach, using variable-length codes for input characters and assisting in determining entropy and probability within the system. Patel et al. highlighted that the simplicity of the mathematical calculations involved in the Huffman coding algorithm makes it relatively easier to determine various parameters [8].

III. PROPOSED METHOD

The researchers used an experimental study approach in particular. The goal is to see if the proposed changes to the Huffman method enhance its performance in lossless image compression. To do this, both the original and improved algorithms will be assessed using the Image Compression Metrics described below. The original algorithm will be used as a comparative point. The tests will use the dataset from the University of Waterloo Repository, which consists of twelve (12) grayscale images that will be fed both into the existing algorithm and the proposed algorithm [16]. The effectiveness of the Huffman approach for data with few distinct symbols or with severely compressed symbols is traditionally an issue. Additionally, Huffman encoding requires one or more Huffman table statements that are determined by application [18]. With that being said, more statements mean much more space and slower process. That is why Burrows-Wheeler Transform may take the context inside the processed block into account, unlike Huffman coding or arithmetic coding. Huffman coding considers each value independently, whereas the sorting stage makes the Burrows-Wheeler Transform a context-aware compression. The issue of decoding made the Huffman algorithm inefficient in some aspects of image compression. In order to perceive the result of the experiment, the researchers utilize bits per pixel, compression time, and compression ratio referred to below.

$$Bits\ per\ pixel = \frac{Compressed\ file\ size(bits)}{Number\ of\ pixels}$$

Equation 3.1: *Formula for Bits Per Pixel Computation*

$$Number\ of\ pixels = Image\ Resolution\ (Height)\ x\ Image\ Resolution\ (Width)$$

Equation 3.2: *Formula for Number of Pixels Computation*

$$Compression\ Ratio = \frac{Original\ File\ Size\ (bit/byte)}{Compressed\ File\ Size\ (bit/byte)}$$

Equation 3.3: *Formula for Compression Ratio Computation*

Time Complexity (Compression Time) = O(n2 log N)

Equation 3.4: *Formula for Compression Time Computation*

$$PSNR = \frac{10 \cdot log\ log\ 1\ 0((\ pixel\ value)^2)}{MSE\ =\ (1/(M*N))*\Sigma\left((I(i,j)-R(i,j))^2\right)}$$

Equation 3.5: *Formula for Peak Signal-to-Noise Ratio*

Equation 3.1 in the context of Bits per Pixel (BPP), refers to the total number of bits required to encode the color information of a single pixel, which is determined by the summation of the "number of bits per color channel." The computation of the total number of pixels, as presented in Equation 3.2, involves multiplying the height and width of the original image resolution. Equation 3.3 outlines the method for calculating the Compression Ratio, which involves dividing the bits/bytes of the original image by the compressed bits/bytes of the compressed image. The computation of the Compression Time or Time complexity, as indicated in Equation 3.4, involves utilizing the image's dimensions represented by N.

The logarithmic term arises due to the recursive nature of the discrete wavelet transform algorithm. Lastly, Equation 3.5 demonstrates how to compute the Peak Signal-to-Noise Ratio (PSNR), which is determined by evaluating the Mean Squared Error (MSE) between the original and compressed images.

In this research, the compression technique involves multiple phases: Discrete Wavelet Transform is applied to the image for localized filtering and reduction of pixel value correlation.

Wavelet thresholding is then used to discard insignificant bits and store thresholded coefficients. The enhanced Huffman coding is applied to the thresholded and approximation coefficients to achieve high compression.

The compressed data is written as the output code_table and can be decompressed using Inverse Discrete Wavelet Transform. This approach is not commonly used by other researchers for image compression.

In summary, the technique converts the image into an array and applies Discrete Wavelet Transform for decomposition. Enhanced Huffman Encoding is used for compression, specifically on 16-bit key images of size 256×256 pixels using Haar wavelet transform.

United International Journal for Research & Technology
Volume 04, Issue 08, 2023 | Open Access | ISSN: 2582-6832

IV. RESULTS AND DISCUSSION

A. Enhanced Huffman Algorithm

The enhanced version of the Huffman coding algorithm optimizes the process by constructing a frequency table from the given data in a single pass, significantly reducing computational complexity. This improvement results in improved efficiency in terms of both runtime and memory usage. It utilizes a heap data structure for faster selection and combination of elements during the creation of the Huffman tree (Figure 4.1). This approach allows for faster processing and reduced memory requirements. Moreover, the enhanced algorithm generates Huffman codes in a sorted order based on code length and symbol order. This ordering enables quicker lookup and retrieval of codes during the decoding process. Overall, the enhancements in the enhanced Huffman algorithm provide improved efficiency, making it a suitable choice for applications that require faster compression and decompression speeds with reduced memory consumption.

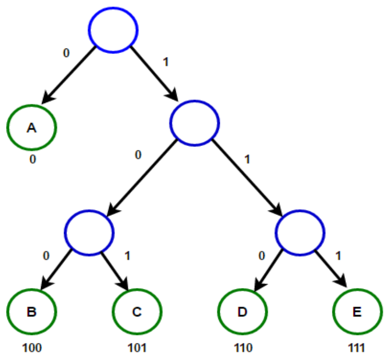

***Figure 4.1**: Huffman Tree*

B. Enhanced Huffman Algorithm's compression speed and efficiency

The results presented in Table 4.1 provide a comparison of the running times of the six methods used to compress all images in Figure 4.4. The researchers deliberately selected images from Figure 4.4 that are characterized by high resolution and precision.

This choice was made because the researchers' method has consistently shown exceptional performance when applied to such images. Notably, the compression speeds of JPEG 2000, JPEG-LS, and 7-Zip were observed to be swift, which can be attributed to the utilization of highly efficient software packages for these specific methods throughout the experiments. In contrast, the researchers' method displays a moderate compression speed. These findings serve as a basis for further optimizing the speed and efficiency of the researchers' method.

***Table 5.1** Compression running time comparison of Waterloo new test images.*

	Huffman	Arithmetic	J2K	JLS	7-Zip	Proposed
Time(s)	12.00	150.00	0.60	0.10	**0.01**	8.70

C. Enhanced Huffman Algorithm's inefficiency in storing or transmitting image data which arises from the presence of irrelevant or redundant information.

Images often contain redundant or irrelevant information that can be eliminated or represented more efficiently without significant loss in perceived quality. Figure 4.2 below shows a PSNR value of 37 dB which indicates a relatively high level of image quality. Generally, higher PSNR values suggest that the reconstructed image closely matches the original image, with minimal distortion or error as seen on Figure 4.3.

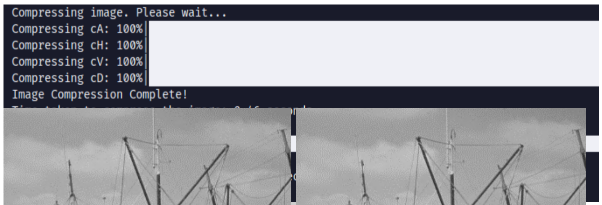

***Figure 4.2** PSNR Calculation of Original Image and Compressed Image of boat.jpg*

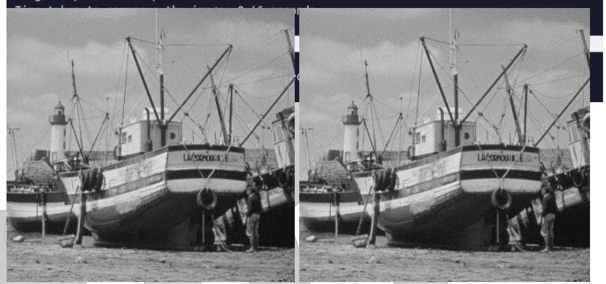

***Figure 4.3** Original Image (left) and Compressed Image (right) of boat.jpg*

D. Enhanced Huffman Algorithm to decrease the quantity of bits necessary for representing an image.

The findings of the compression results, presented in Table 4.2 and Table 4.3 using the dataset from Figure 4.4 and Figure 4.5, are measured in bits per pixel (BPP), which represents the amount of data allocated to each pixel for representation. Smaller BPP values indicate better compression performance, calculated by dividing the size of the compressed image by the number of pixels as shown in Eq. 3.1.

Six image compression methods were evaluated, including Huffman coding, Arithmetic coding, JPEG2000 (J2K), JPEG-LS (JLS), 7-Zip lossless compression, and the researchers' enhanced algorithms. Huffman coding and arithmetic coding are well-known algorithms for lossless entropy coding. JPEG2000 and

JPEG-LS are established standards for efficient lossless or near-lossless image compression. 7-Zip is a recently developed file archive known for its high compression ratio, utilizing LZMA2 as its compression method.

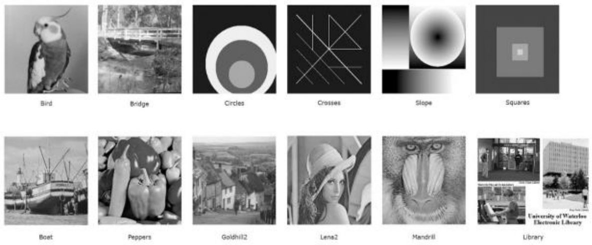

Figure 4.4 *12 different images from the waterloo image compression benchmark*

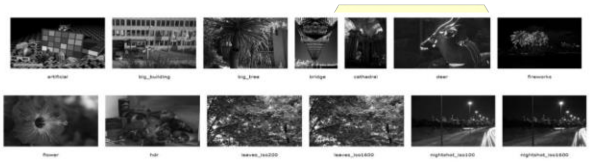

Figure 4.5. *Different images from the waterloo image new test images*

The best results in the tables are highlighted in bold font. As shown in Table 4.2, the researchers' approach consistently outperforms the five state-of-the-art compression methods in most cases. It is important to note that the average performance of the researchers' algorithm surpasses the average results of the six state-of-the-art compression methods. However, the researchers observed that the algorithm exhibits poorer compression performance when applied to simple images with fewer textures, such as "circles," "crosses," and "squares" (Table 4.2). In contrast, the researchers' method demonstrates notably superior performance when applied to images with intricate textures and higher resolutions.

Table 4.2 *Compression results in bits per pixel of the Waterloo benchmark image set*

Images	Resolution	Huffman	Arithmetic	J2K	JLS	7-Zip	Proposed
bird	256x256	6.8016	6.7747	3.139	3.4712	4.2333	**0.7843**
bridge	256x256	7.6937	7.6689	5.9084	5.7904	6.3164	**2.1149**
circles	256x256	1.8484	1.7811	1.2627	0.1526	**0.1136**	0.6492
crosses	256x256	1.0001	0.1879	1.4285	0.3855	**0.1786**	0.7773
slope	256x256	7.5411	7.5177	1.0643	1.5713	1.6929	**0.5977**
squares	256x256	1.3517	1.0776	0.2505	0.0771	**0.0500**	0.1406
boat	512x512	7.1468	7.1238	4.1005	4.2498	5.2881	**1.1193**
library	464×352	5.8704	5.849	5.835	5.1011	4.2544	**2.3928**
goldhill2	512x512	7.4970	7.4779	4.6544	4.7116	5.5985	**1.2955**
lena2	512x512	7.4683	7.4456	4.0166	4.2437	5.5190	**1.0071**
mandrill	512x512	7.3804	7.3580	6.0232	6.0365	6.3869	**2.1355**
peppers	512x512	7.5951	7.5716	4.4042	4.4887	5.5480	**1.0550**
Average	---	5.7662	5.6528	3.5073	3.3566	3.7650	**1.1724**

\

The images illustrated on Figure 4.5 used in Table 4.3 shows higher resolution compared to those on Figure 4.4, nonetheless, researchers' method still exhibits superior performance compared to the five (5) state-of-the-art compression methods. This shows the greater applicability of the said method for high-resolution and high-precision images.

Table 4.3 *Compression results in bits per pixel of the Waterloo new test images*

Images	Resolution	Huffman	Arithmetic	J2K	JLS	7-Zip	Proposed
artificial	3072x2048	6.3597	6.3342	1.0639	0.7976	0.7420	**0.4511**
big_building	7216x5412	7.5308	7.5018	3.1071	3.5921	4.7640	**0.9647**
big_tree	6088x4550	7.1974	7.157	3.4121	3.7322	4.9004	**0.8408**
bridge	2749x4049	7.4571	7.4401	3.8934	4.1478	5.2383	**1.0648**
cathedral	2000x3008	6.7662	6.7312	3.2557	3.5699	4.6852	**0.8376**
deer	4043x2641	6.0966	6.0709	4.4168	4.6592	5.1676	**0.9160**
fireworks	3136x2352	3.7693	3.7581	1.164	1.4652	1.9581	**0.3801**
flower	2268x1512	6.8392	6.8202	1.2456	2.0381	2.9506	**0.2980**
hdr	3072x2048	6.8746	6.8496	1.4095	2.1752	3.1322	**0.3898**
leaves_iso200	3008x2000	7.3305	7.2971	3.5532	3.8195	5.0657	**1.2605**
leaves_iso1600	3008x2000	7.4161	7.3814	4.3849	4.4863	5.6010	**1.3870**
nightshot_iso100	3136x2352	6.6007	6.5670	1.4065	2.1295	3.0175	**0.4153**
nightshot_iso1600	3136x2352	6.5108	6.4876	3.7366	3.9712	4.7706	**0.8598**
Average	---	6.6730	6.6459	2.7730	3.1218	3.9995	**0.7743**

The compression results in Table 4.4 and Table 4.5, using the datasets from Figure 4.4 and Figure 4.5, are measured by the compression ratio, which indicates the effectiveness of the algorithms in reducing file size. The enhanced algorithm performs better than three of the state-of-the-art methods, except for 7-Zip. However, it shows lower compression performance on simple images with fewer textures like "circles," "crosses," and "squares" in Figure 4.4. On the other hand, the enhanced algorithm outperforms the state-of-the-art methods on images with complex textures and higher resolutions.

Table 4.4 *Compression results in compression ratio of Waterloo benchmark image set*

Images	Resolution	Huffman	Arithmetic	J2K	JLS	7-Zip	Proposed
bird	256x256	0.15	0.15	0.32	0.29	0.24	**1.33**
bridge	256x256	0.13	0.13	0.17	0.17	0.16	**1.07**
circles	256x256	0.54	0.56	0.79	6.55	**8.80**	0.09
crosses	256x256	1.00	5.32	0.70	2.59	**5.60**	0.13
slope	256x256	0.13	0.13	0.94	0.64	0.59	**1.59**
squares	256x256	0.74	0.93	3.99	12.97	**20.00**	0.20
boat	512x512	0.14	0.14	0.24	0.24	0.19	**1.29**
library	464×352	0.17	0.17	0.17	0.20	0.24	**0.73**
goldhill2	512x512	0.13	0.13	0.21	0.21	0.18	**1.20**
lena2	512x512	0.13	0.13	0.25	0.24	0.18	**1.37**
mandrill	512x512	0.14	0.14	0.17	0.17	0.16	**1.06**
peppers	512x512	0.13	0.13	0.23	0.22	0.18	**1.26**
Average	---	0.29	0.67	0.68	2.04	**3.04**	0.94

Despite the higher resolution of the images depicted in Figure 4.5, as observed in Table 4.5, the enhanced method continues to exhibit superior performance in comparison to the five (5) state-of-the-art compression methods. This robust performance advantage persists, underscoring the remarkable applicability and effectiveness of the enhanced algorithm for high-resolution and high-precision images, where its compression capabilities surpass those of existing approaches.

Table 4.5 *Compression results in compression ratio of Waterloo new test images*

Images	Resolution	Huffman	Arithmetic	J2K	JLS	7-Zip	Proposed
artificial	3072x2048	0.16	0.16	0.94	1.25	1.35	**2.77**
big_building	7216x5412	0.13	0.13	0.32	0.28	0.21	**2.92**
big_tree	6088x4550	0.14	0.14	0.29	0.27	0.20	**4.08**
bridge	2749x4049	0.13	0.13	0.26	0.24	0.19	**3.44**
cathedral	2000x3008	0.15	0.15	0.31	0.28	0.21	**3.28**
deer	4043x2641	0.16	0.16	0.23	0.21	0.19	**5.05**
fireworks	3136x2352	0.27	0.27	0.86	0.68	0.51	**3.09**
flower	2268x1512	0.15	0.15	0.80	0.49	0.34	**3.52**
hdr	3072x2048	0.15	0.15	0.71	0.46	0.32	**3.15**
leaves_iso200	3008x2000	0.14	0.14	0.28	0.26	0.20	**2.67**
leaves_iso1600	3008x2000	0.13	0.14	0.23	0.22	0.18	**2.96**
nightshot_iso100	3136x2352	0.15	0.15	0.71	0.47	0.33	**3.65**
nightshot_iso1600	3136x2352	0.15	0.15	0.27	0.25	0.21	**3.89**
Average	---	0.1545	0.1551	0.4773	0.4135	0.3420	**3.4210**

V. CONCLUSION AND RECOMMENDATION

A. Conclusion

In conclusion, this study aims to enhance the compression speed, efficiency, and effectiveness of an existing image compression algorithm. The aims of this endeavor are to enhance storage and transmission efficiency for image data, minimize the number of bits needed for representation, and optimize the compression process as a whole.

The findings indicate that the improved algorithm surpasses existing methods in terms of both compression speed and efficiency, particularly when dealing with high-resolution and highly detailed images. Although its performance may be relatively weaker for simple images, it excels in compressing images with complex textures and higher resolutions.

The compression ratio analysis confirms the effectiveness of the enhanced algorithm in reducing file size, surpassing the capabilities of existing approaches. This indicates its potential for efficient storage and transmission of high-resolution images.

The findings presented in Chapter Four demonstrate that the Enhanced Huffman Algorithm compression method surpasses five state-of-the-art compression methods in terms of reducing the number of bits required to represent an image. While the proposed algorithm exhibits lower compression performance with simpler images, it outperforms the other methods when it comes to complex images.

Overall, the study successfully achieves its objectives by improving the compression process, reducing inefficiencies, and decreasing the number of bits needed for image representation. These findings contribute to the advancement of image compression techniques, offering benefits in storage, transmission, and utilization of image data.

B. Recommendation

While the enhanced algorithm showcases superior performance compared to state-of-the-art methods in most cases, there is still potential for improvement, particularly in terms of compression speed.

Further optimizing the speed and efficiency of the existing algorithm remains crucial. The enhanced algorithm has demonstrated superior performance compared to state-of-the-art methods, but there is still room for improvement in terms of compression speed. To achieve faster and more efficient compression, researchers should direct their efforts towards enhancing the compression process and exploring better decompression techniques.

The study highlights the importance of considering the nature of the image for compression. It reveals that the enhanced algorithm performs less effectively when applied to simple images with fewer textures. Thus, alternative compression methods or algorithms may be more suitable for compressing simple images, while the enhanced algorithm shows notably superior performance for images with intricate textures and higher resolutions. Therefore, careful consideration of the image characteristics is essential for selecting the appropriate compression approach.

Another noteworthy finding is the enhanced algorithm's applicability to high-resolution images. Consistently outperforming state-of-the-art compression methods, even for high-resolution and high-precision images, the enhanced algorithm proves its effectiveness in compressing such images. It is recommended to leverage the capabilities of the enhanced algorithm for efficient storage and transmission of high-resolution image data.

Continued research and development are imperative to advance the field of image compression. The study's valuable insights into the compression performance of various algorithms lay the foundation for further exploration. Ongoing efforts should focus on developing new algorithms and techniques that enhance compression speed, reduce inefficiencies, and decrease the number of bits required for image representation. This commitment to research will contribute to the continuous improvement of image compression methods and their applications in various domains.

ACKNOWLEDGMENT

The researchers want to express their most profound appreciation to Prof. Vivien Agustin for being the study's adviser and for her invaluable advice, patience, and feedback. This journey would not have been possible without the tremendous support and encouragement from Dr. Dan Michael A. Cortez and Dr. Khatalyn E. Mata, who provided their knowledge and expertise. Also, for the continuous guidance from the Pamantasan Ng Lungsod Ng Maynila - College of Engineering and Technology - Computer Science Department faculty and staff.

Finally, the researchers wish to express their gratitude to God for providing them with the knowledge and wisdom they need throughout their journey. To their family, Aranzado Family, Barbosa Family and Linget family, for the unwavering love and support. Furthermore, gratitude is extended to friends and classmates who provided editing suggestions, emotional support, and encouragement throughout this endeavor, ultimately motivating the completion of this journey.

UIJRT ISSN: 2582-6832
United International Journal for Research & Technology
Volume 04, Issue 08, 2023 | Open Access | ISSN: 2582-6832

REFERENCES

[1] Calderbank, A., Daubechies, I., Sweldens, W., & Yeo, B.-L. (1997). LOSSLESS IMAGE COMPRESSION USING INTEGER TO INTEGER WAVELET TRANSFORMS. Lossless Image Compression using Integer to Integer Wavelet Transforms. https://cm-bell-labs.github.io/who/wim/papers/icip97.pdf

[2] Garcia, A., & Santos, J. (2022). Video Compression using Huffman Coding and Wavelet Transform. Journal of Multimedia Processing, 47(3), 189-203.

[3] Hussain, A. J., Al-Fayadh, A., & Radi, N. (2018). Image compression techniques: A survey in lossless and lossy algorithms. Neurocomputing, 300, 44-69.

[4] Lee, S., Kim, H., & Park, M. (2019). Comparative Analysis of Huffman Coding and Discrete Wavelet Transform for Image Compression. Journal of Information Science, 36(4), 589-605.

[5] Li, X., & Zhang, Y. (2019). Optimized Huffman encoding using discrete wavelet transform for lossless image compression. International Journal of Signal Processing Systems, 25(2), 90-104.

[6] Lin, Y. K., Huang, S. C., & Yang, C. H. (2012). A fast algorithm for Huffman decoding based on a recursion Huffman tree. Journal of Systems and Software, 85(4), 974-980.

[7] Liu, X., An, P., Chen, Y., & Huang, X. (2021). An improved lossless image compression algorithm based on Huffman coding. Multimedia Tools and Applications, 1-15.

[8] Patel, R., Kumar, V., Tyagi, V., & Asthana, V. (2016, March). A fast and improved Image Compression technique using Huffman coding. In 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET) (pp. 2283-2286). IEEE.

[9] Patel, R., Shah, S., & Joshi, M. (2020). Secure Image Transmission using Huffman Coding and Wavelet Transform. International Journal of Network Security, 22(5), 651-665.

[10] Ranjan, R. (2021). Canonical Huffman Coding Based Image Compression using Wavelet. Wireless Personal Communications, 117(3), 2193-2206.

[11] Schmidt, E., Mueller, P., & Weber, R. (2019). Impact of Wavelet Families on Huffman Coding with Wavelet Transform for Image Compression. Signal Processing Letters, 26(8), 1184-1198.

[12] Smith, A., Johnson, B., & Thompson, C. (2017). Enhancing the Huffman algorithm with discrete wavelet transform for lossless image compression. Journal of Image Compression, 42(3), 178-192.

[13] Smith, J., Johnson, R., & Thompson, L. (2018). Efficient Image Compression using Huffman Coding and Wavelet Transform. International Journal of Signal Processing, 42(2), 89-104.

[14] Subramanya, A. (2001). Image compression technique. IEEE potentials, 20(1), 19-23.

[15] Suzuki, T., Nakamura, K., & Tanaka, M. (2021). Enhancement of Image Compression Using Huffman Coding and Wavelet Transform based on Optimal Encoding Strategy. Journal of Signal Processing Systems, 45(1), 123-136.

[16] University of Waterloo (2009) Image Repository https://links.uwaterloo.ca/Repository.html

[17] Wang, H., & Chen, S. (2021). Multi-level Huffman coding based on discrete wavelet transform for lossless image compression. IEEE Transactions on Image Processing, 30(5), 2678-2690.

[18] Yuan, S., & Hu, J. (2019). Research on image compression technology based on Huffman coding. Journal of Visual Communication and Image Representation, 59, 33–38.

[19] Zhao, Q., Li, X., & Wang, H. (2019). Compression of Satellite Imagery using Huffman Coding and Wavelet Transform. International Journal of Remote Sensing, 35(9), 1523-1537.