SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V12I12P104

Volume 12 Issue 12, 29-43, December 2025

Original Article

Block-Based Lossless Image Coding through Image Quality Improvement using the Prediction by Partial Matching Algorithm

P.R. Rajesh Kumar[1], M. Prabhakar[2]

[1,2]*School of Computer Science and Engineering, REVA University, Karnataka, India.*

[1]*Corresponding Author : r21pcs08@reva.edu.in*

Received: 04 October 2025 Revised: 06 November 2025 Accepted: 05 December 2025 Published: 27 December 2025

***Abstract** - Applications requiring exact image reconstruction always need lossless image coding. A novel method of block-based lossless image coding based on the use of the Prediction by Partial Matching (PPM) algorithm combined with two-channel coding and adaptive Huffman coding is presented in this paper. In this work, images are segmented into non-overlapping blocks, and pixels are efficiently predicted through the application of context modeling using PPM. To improve coding efficiency, a Two-channel coding is employed to separate bit and data streams. The encoded streams are further compressed by a Huffman coding scheme, adaptively adjusting symbol probabilities to local data statistics. The experimental results demonstrate an improvement in the compression ratio while maintaining image quality. Working with statistical and predictive models, the integration of PPM and two-channel and adaptive Huffman coding has created a flexible and robust coding framework. Finally, the proposed method is compared with previous state-of-the-art lossless coding techniques and evaluated in terms of compression efficiency as well as computational behavior, and found to be superior in both aspects. The proposed method demonstrates an average improvement of 46.97% in CR, 32.62% in BPP, and 2.05% in entropy compared to the TIFF, BMP, and LZW methods. This has shown a bright technology in high-fidelity image storage and transmission devices.*

***Keywords** - Adaptive Huffman Coding, Compression Ratio, Loss Image coding, Prediction by Partial Matching (PPM), Two-Channel coding.*

1. Introduction

The recent increase in the application of digital imaging to fields such as medical diagnostics, satellite remote sensing, surveillance, and archival systems has placed a significant strain on the requirement for highly efficient and lossless image compression methods. Unlike lossy compression, lossless image coding [1] preserves every bit of the original image. It can thus be applied to restore the original image bit-identically, which is a crucial quality in applications where any amount of information loss would compromise diagnostic quality, legal, or other forms of scientific validity. Traditional lossless coding techniques studied widely include entropy-based Huffman coding, arithmetic coding, and dictionary-based coding [2]; however, these methods have several constraints: they are generally unable to adapt to the properties of various images, they are unable to exploit spatial redundancies, and they are unable to achieve high compression efficiency at a low computing cost. Despite the significant advances, a critical research gap remains. Current lossless compression algorithms cannot simultaneously achieve three key objectives, namely: prediction accuracy, compression efficiency, and computational scalability. Prediction by Partial Matching (PPM) [3] has good statistical modeling properties, but has not found application in image compression due to its computational complexity and failure to localize redundancy. Images are also treated holistically by many methods, and block-wise variability and regional patterns, which play a significant role in predictive accuracy, are ignored. In addition, the majority of current methods are based on single predictive or entropy coding schemes, rather than a combined framework that can leverage their advantages.

To overcome these weaknesses, this paper proposes a single block-based lossless image compression system that combines Prediction by Partial Matching (PPM), Two-Channel Coding, and Adaptive Huffman Coding. The approach divides images into non-overlapping blocks and is therefore better at capturing region-specific redundancy, as well as enabling PPM to adopt its statistical modeling to localized patterns. Two-channel coding is used to separate bit and data streams, improving representational compactness.

Adaptive Huffman coding [4] is employed to reduce entropy by dynamically adjusting the probability of symbols. All these features, together, explicitly fill the outlined research gap, providing better prediction accuracy, higher compression rates, and flexibility in dealing with different image types and resolutions.

1.1. Motivation

Medical imaging, forensic analysis, remote sensing, and digital archiving are some of the applications that require compression solutions that ensure flawless reproduction of image data. In such areas, a single pixel change can cause an error in interpretation or a aura of degrading the quality of stored data. Despite the numerous techniques suggested, achieving high compression efficiency, low computational complexity, and visual fidelity has been simultaneously difficult to attain.

A promising direction for enhancing prediction-based compression is Prediction by Partial Matching (PPM), which is characterized by its adaptive statistical modeling. Its direct implementation with full-resolution images, however, has a problem of scalability and lower efficiency. A block-based processing approach can be used to overcome these problems by enabling localized predictions and minimizing CPU usage. Moreover, a combination of PPM and two-channel coding, along with adaptive entropy coding, may potentially yield significant benefits by leveraging the synergies of predictive coding and dynamic redundancy elimination. These are the reasons why it is desirable to develop an integrated, adaptable, and scalable lossless compression model that can address the demands of modern imaging without any loss in accuracy at reconstruction.

1.2. Problem Statement

Lossless image compression cannot be neglected in critical imaging-related areas, such as medical diagnostics, scientific research, surveillance, and digital archiving, as the integrity of an image must be preserved in all cases.

Though there has been a significant improvement, the current lossless image compression methods have significant limitations:

1. Poor adjustment to local image properties leads to poor prediction and low compression rates.
2. Poor standalone predictive/entropy coding performance, since these methods cannot be used to leverage complementary performance when applied independently.
3. Computational inefficiency of more complex statistical models, including PPM, for the full high-resolution images.
4. Lack of sufficient integration of multi-stage compression mechanisms results in fragmented solutions, which fail to optimise redundancy removal.

To overcome these shortcomings, this study presents a single block-based compression architecture that leverages the advantages of PPM to achieve better local prediction, Two-Channel Coding to enhance data representation, and Adaptive Huffman Coding to achieve dynamic entropy reduction. It aims to offer a high-performance, scalable, and domain-independent lossless compression algorithm that can significantly improve the performance of traditional algorithms while retaining precise image fidelity.

Detailed implementation of the proposed method is discussed in the remainder of this paper. Section 2 discusses the literature review, Section 3 discusses the proposed methodology, Section 4 presents system setup and database details, Section 5 presents experimental results compared against a number of existing lossless coding techniques, and finally Section 6 discusses the conclusion and an outlook for future developments.

2. Literature Review

The literature review is fundamental in any research study because it gives a precise explanation of what has been done, how it has been done, and what challenges have been experienced. It sheds light on the development of predictive coding, entropy coding, transform-based, and modern deep-learning models in lossless image compression. It analyzes classical algorithms, such as Huffman coding, LZW, JPEG-LS, and arithmetic coding, and neural network-based systems, and their advantages and drawbacks. The most common problems are low compression ratios, high computational costs, limited adaptability to a wide range of image types, and the inability to exploit localized redundancies. The review avoids duplication of research as it critically reviews past studies, reveals research gaps, and helps in steering the innovation path. It also serves the purpose of making sure that the proposed technique will not go against what is known, but will provide some improvements. Altogether, the literature review creates a solid theoretical background and justifies the necessity of the present study, as well as the choice of the practical approaches to attaining efficient image compression and its complete reversibility.

JPEG XS is a new international standard created by Antonin Descampe, Thomas Richter, and the JPEG Committee as a code allowing images to be visually lossless, with low latency and low weight, to meet various needs in the AV industry. It is compatible with video delivery, real-time storage, and sensor-based compression applications. The Core Coding System contains a color transform, wavelet transform, and a new entropy encoder, which was developed to provide a better quality with no visual distortions at moderate compression ratios. The work in question describes the characteristics, profiles, format, performance, and continuous standardization achievements of the standard. The experimental findings demonstrate that JPEG XS is highly multi-generation robust and maintains the quality when it is encoded and decoded repeatedly. Subjective ratings suggest that it can offer visual lossless quality as far as a compression factor of 6:1, except in the case of very complicated images. Further studies are necessary to refine the subjective evaluation and enhance visual quality assurance. Additionally, it is necessary to investigate the compatibility and optimal performance of JPEG XS with new video transport technologies and protocols.

Regarding the research problem of a bit error of wireless transmission, Jungan Chen, Jean Jiang et al. [6] propose a bit-error-conscious lossless image compression algorithm with bi-level coding of gray images. It introduces a novel programming variable-size 2D block extraction and encoding algorithm that can be applied to achieve higher compression ratios by leveraging statistical correlation in the 2D context. An RGB-to-YCrCb lossless color transformation reduces the decorrelations between color components, and built-in bi-level coding is employed in a manner that is resistant to bit errors. The experimental findings demonstrate higher compression rates than the present ones, and the image quality is high even in the bit-error environment. The specified solution will not always be superior to state-of-the-art methods in terms of compression ratio, but it is less susceptible to bit errors. The article draws our attention to the need to improve the generalization of deep learning methods in the context of applying them to novel data.

Shinichi Yamagiwa et al. [7], Wenjia Yang et al., suggested a novel system of adaptive lossless compression of image data by Deep Neural Networks (DNN) in order to deduce the entropy of the data. It addresses the issue of selecting an optimal compression algorithm without trial and error, which is typically NP-hard. DNN separates the original data into smaller blocks and learns the data trends and the optimal compression algorithm to apply in each block, thereby enhancing the compression efficiency. The experimental evaluations demonstrated that the method provides better compression ratios with an increment of 8 to 15 times when the worst case scenario is that one program is run on the entire data. However, the technique is also associated with a number of limitations, including the fact that the training images and data block sizes are fixed. Further studies will be conducted on the subject of variable block size and its implementation to other multimedia data forms.

Tungshou Chen, Xiaoyu Zhou et al. [8] provided a high-quality picture authentication technique in their research results by applying Absolute Moment Block Truncation Coding (Ambtc) to compressed pictures. It highlights the weaknesses of the existing AMBTC authentication systems, specifically the ability to overlook certain types of manipulations and the impact of this issue on image quality. Among the provided techniques, there is the division of image blocks into smooth and complex blocks, and the enhancement of the quality by switching bits in the smooth blocks to generate authentication codes. It employs the matching (PPM) method of inserting codes with a small error, which provides improved detection performance, and the image quality is better than that of earlier methods. The efficiency of the method is also due to the fact that the Pixel Pair Matching (PPM) technique is utilized to encode the codes, which in turn would lead to the lowest error possible. Overall, the results of the experiments indicate that the quality of performance in terms of detection and image quality enhancement is higher. It is possible to conduct the subsequent study with the help of a recoverability functionality that allows for the restoration of tampered parts of AMBTC compressed images, thereby enhancing the overall strength of the authentication process. Another potential area of research would be to examine the enrichment techniques of embedding that could further improve the quality of the image without compromising the level of detectability.

Tassnim Dardouri, Mounir Kaaniche, et al. [9] propose an alternative method for encoding images using a dynamic neural network, which is expected to improve models of lossy-to-lossless compression. It aims to become capable of learning lifting operators with an architecture based on a Fully Connected Neural Network (FCNN), thereby improving prediction and updating filters. Two adaptive learning approaches are presented to enhance the model's optimization based on the content of the input image, thereby improving the encoding process and yielding favorable experimental results. The findings reveal that the following generation of wavelets to be employed in compression can be proposed by integrating neural networks in the lifting schemes. Future research may involve extending the advised Fully Connected Neural Network (FCNN)-based schemes to more sophisticated lifting schemes, particularly vector lifting schemes, and enhancing the compression of color images. Image compression techniques may also be improved through research on the use of state-of-the-art neural network architectures.

One of the Deep Lossy and Residual (DLPR) coding structures for lossless and near-lossless image coding is introduced by Yuanchao Bai, Xianming Liu, et al. [10], and it is not mentioned in the existing literature that provides both. This framework operates with lossy compression of the data in lossless mode and lossless coding of the residuals, which are further advanced using Variational Autoencoders (VAEs) and autoregressive context modeling. The near-lossless operation limits quantization to support set error targets, allowing for universal compression without requiring additional networks. The suggested design of the context coding and adaptive residual interval scheme is significantly superior in terms of coding speed. Literature tests prove that the DLPR coding system achieves the most recent compression performance and the fastest coding speed among all resolution image activities. Future studies can deal with the optimization of the DLPR coding scheme to the point of increasing the efficiency and rate of the lossless and near-lossless image compression.

The study by Gangtao Xin and Pingyi Fan [11] introduces a new lossless image compression technique known as soft compression, and it seeks to exploit the absence of coding and spatial redundancy by using shapes to encode images. It adds a compressible indicator feature that ascertains the mean number of bits to be used in representing images. The soft compression algorithm yields better results than classical standards, such as PNG and JPEG2000, producing higher compression ratios. The technique can be applied to various image types, including binary, grayscale, and multi-component images, and is likely to reduce bandwidth and storage requirements during transmission and storage. Tests using experiments have proven that soft compression is a better option than classical standards such as PNG and JPEG2000, especially when the images have high values of compressible indicators. The methodology is flexible and can be integrated with other transformation techniques, thereby increasing its usability in critical areas such as medical imaging.

Malgorzata Frydrychowicz and Grzegorz Ulacha propose a two-step prediction error encoding scheme, which combines adaptive Golomb encoding and CABAC proposed in [12], to offer a high level of compression, resulting in a significant enhancement to the encoding performance. It explains why some contextual partition would be required to group data of various nature, and therefore, more effective compression would be attained.

The Conditional Move To Front (CMTF) approach is introduced as one that is beneficial to apply to images with high noise content. It can be concluded from the paper that the Minimum Mean Absolute Error (MMAE) method is superior to the Minimum Mean Square Error (MMSE) method on linear predictive models, particularly in low-noise images. Overall, it will be found that the Blend-28 codec achieves an 11 percent bit reduction compared to the WebP codec.

Their study, by M. Sri Raghavendra, Pasuluri Bindu Swetha et al. [13], presents a new algorithmic model for lossless image compression that efficiently applies intelligent partitioning, selective coding, and wavelet coefficient analysis. The proposed model exhibits superior compression efficiency while maintaining image quality, outperforming established standards such as JPEG2000 and PNG in terms of compression ratios, PSNR, and SSIM scores. Future research directions include combining machine learning automation with the partitioning process, determining the broader applicability of images, and developing adaptive encoding strategies for real-time applications. The developments are designed to support the increasing demand for efficient and cost-effective digital image management solutions. The proposed lossless image compression model has primarily been based on hypothetical conditions and theoretically generated data, without experimental verification using real-life data, which restricts its practical application.

The Deep Lossless Image Coding (DLIC) algorithm, proposed by Benjamin Lukas Cajus Barzen, Fedor Glazov et al. [14], effectively accelerates lossless image encoding in deep neural networks. It can be seen that smaller neural networks can compete with state-of-the-art compression algorithms on 2D images.

This parallelism of the encoding and decoding process leads to run times in practice, which apply to the real world. The ability of DLIC to be adapted to domain-specific (e.g., MRI scans) data is not without difficulty, especially regarding the size of networks and training to compete with the current state of the art.

Xiaoxiao Liu, Ping An et al. [15] present an improved lossless compression scheme for images, including lossless linear prediction, coefficient processing of LeGall's integer wavelet transforms, and Huffman coding. The proposed algorithm essentially eliminates redundancy between the adjacent pixels, which has the effect of reducing the entropy value and improving the compression ratio.

There is empirical evidence that the algorithm has been discovered to outperform the state-of-the-art techniques, especially in high-resolution and complex images. Subsequent work on the technique should focus on adjusting it to a broader range of image types and on making the algorithm more complex and efficient.

Their work, Grzegorz Ulacha, Ryszard Stasiński et al. [16] proposed an Extended Multi WLS (EM-WLS) algorithm, which is the most efficient process of lossless image coding in terms of data compaction as compared to other algorithms. It emphasizes that the EM-WLS algorithm is computationally less complex than its major competitors, allowing for easier use.

The design of the EM-WLS algorithm was based on a range of observations regarding pixel prediction and dependencies between prediction error and coder average data rate. The presented new binary context arithmetic coder is significantly less complex compared to past data modeling stages and can be applied more widely in image compression techniques. A summary of the discussed literature is shown in Table.1.

Table 1. Summary of literature review

S.No	Title	Author	Methodology	Limitations
1	"JPEG XS - A new standard for visually lossless low-latency lightweight image coding"	Antonin Descampe, Thomas Richter, et al. [5]	Quantizes by using a dead-zone quantizer and data-dependent uniform quantizer, and splits the already quantized wavelet coefficients into coding groups.	JPEG XS does not perform a fully-fledged rate-distortion optimization. The quantization process may affect the precision of the reconstruction.
2	"Bit-Error Aware Lossless Image Compression with 2D-LayerBlock Coding"	Jungan Chen, Jean Jiang et al. [6]	Lossless color transformation, prediction-based approach, variable-size 2D-block extraction, and encoding approach proposed to enhance compression efficiency, preserve image quality, as well as eliminate bit errors.	The optimization of 2D-block start bits and its impact on compression rates requires further exploration to enhance the overall efficiency of the method.
3	"Adaptive Lossless Image Data Compression Method Inferring Data Entropy by Applying Deep Neural Network "	Shinichi Yamagiwa, Wenjia Yang et al. [7]	A novel method utilizes DNN to predict the entropy of data blocks, PCA, alongside the DNN to enhance the prediction of data entropy.	Dependency of the DNN inference on the specific images used for training, which may limit the generalizability of the method. Use of fixed data block sizes for training and inference, which may not be optimal for all scenarios.
4	"A High Fidelity Authentication Scheme for AMBTC Compressed Image Using Reference Table Encoding"	Tungshou Chen , Xiaoyu Zhou et al. [8]	Presents a high-quality image authentication method using AMBTC, enhancing image quality by classifying smooth or complex blocks, generating codes, and using PPM and reference tables.	AMBTC authentication methods struggle to detect malicious tampering, embed limited-length codes, and may not consider flipped bitmaps, limiting image quality enhancement and embedding performance.
5	"Dynamic Neural Network for Lossy-to-Lossless Image Coding"	Tassnim Dardouri, Mounir Kaaniche et al. [9]	Presented various image coding methods using lifting schemes and neural networks, including FCNN-LS, D1-FCNN-LS, D2-FCNN-LS, and H-FCNN.	Discusses the complexity of optimizing update filters in lifting schemes, highlighting the need for further research to improve performance and compression efficiency.
6	"Deep Lossy Plus Residual Coding for Lossless and Near-lossless Image Compression"	Yuanchao Bai, Xianming Liu et al. [10]	Introduced a unified Deep Lossy Plus Residual (DLPR) coding architecture of lossless and near-lossless compression of images, which uses Variational Autoencoders to achieve higher performance.	Traditional lossless image codecs' efficiency is limited by 2:1 compression ratios, slow coding speeds, and limited research on near-lossless compression, leading to biased probability models.
7	"Soft Compression for Lossless Image Coding Based on Shape Recognition"	Gangtao Xin and Pingyi Fan [11]	Presents a new soft compression algorithm which is better than classical standards, such as PNG and JPEG2000.	Must be used together with other transformation techniques to increase accuracy, must also be used together with channel coding in order to enhance the impact of joint source-channel coding.
8	"Two-Stage Golomb - Context-Adaptive Binary Arithmetic Coders Coding in Lossless Image Compression"	Małgorzata Frydrychowicz, Grzegorz Ulacha [12]	New predictive coding methods such as Conditional Move To Front, adaptive Golomb coding and context-adaptive binary arithmetic coders to image coding were introduced.	It may be too complex to be practical in some applications because of its computational complexity.

9	"Enhancing Lossless Image Compression through Smart Partitioning, Selective Encoding, and Wavelet Analysis"	M. Sri Raghavendra, Pasuluri Bindu Swetha et al. [13]	Proposes a new algorithmic model of lossless compression of images, combining intelligent partitioning, selective encoding, and wavelet coefficient.	Lacking empirical validation with real-world datasets, which limits its applicability. Adoption of color images.
10	"Accelerated Deep Lossless Image Coding with Unified Paralleleized GPU Coding Architecture"	Benjamin Lukas Cajus Barzen, Fedor Glazov et al. [14]	Introduces Deep Lossless Image Coding (DLIC), a neural network and entropy encoder for efficient coding, speed improvements, and 3D window optimization for MRI scans.	Optimization of long runtimes. Inclusion of metadata in MRI images can improve compression rates.
11	"An improved lossless image compression algorithm based on Huffman coding"	Xiaoxiao Liu, Ping An et al. [15]	Introduces a lossless compression algorithm of images that utilizes linear predictive coding, integer wavelet transformations, coefficients processing and Huffman coding.	Need to accommodate wide range of image types. Complexity and speed of the algorithm is to be optimized.
12	"Extended Multi WLS Method for Lossless Image Coding"	Grzegorz Ulacha, Ryszard Stasi´nski et al. [16]	Introduces the Extended Multi Weighted Least Squares (EM-WLS) and Locally Adaptive Ordinary Least Squares (LA-OLS) algorithms of lossless image coding and enhance the prediction and prediction efficiency.	Proposed algorithm faces computational complexity, processing speed, and resource requirements challenges, including determining image entropy and optimizing modeling and entropy coding stages.

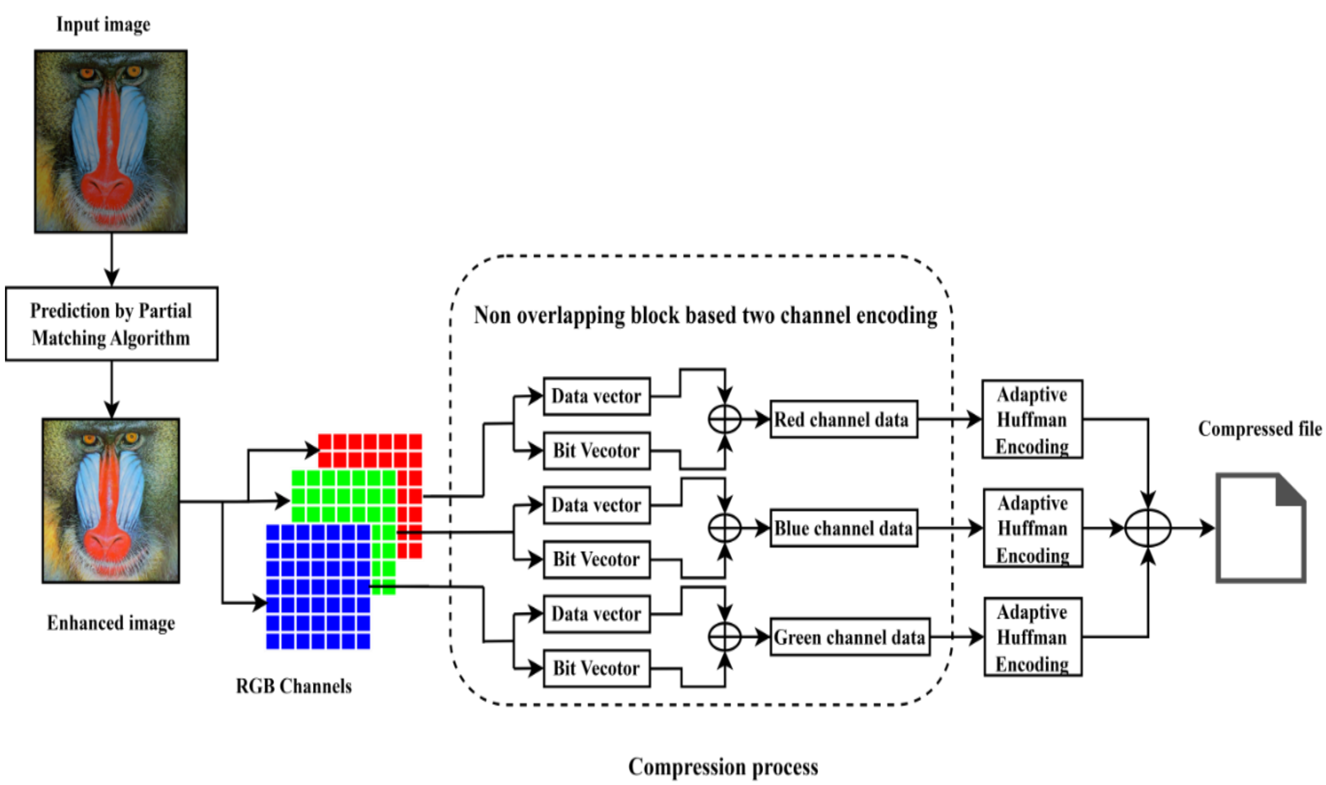

Fig. 1 Encoding process of the proposed method

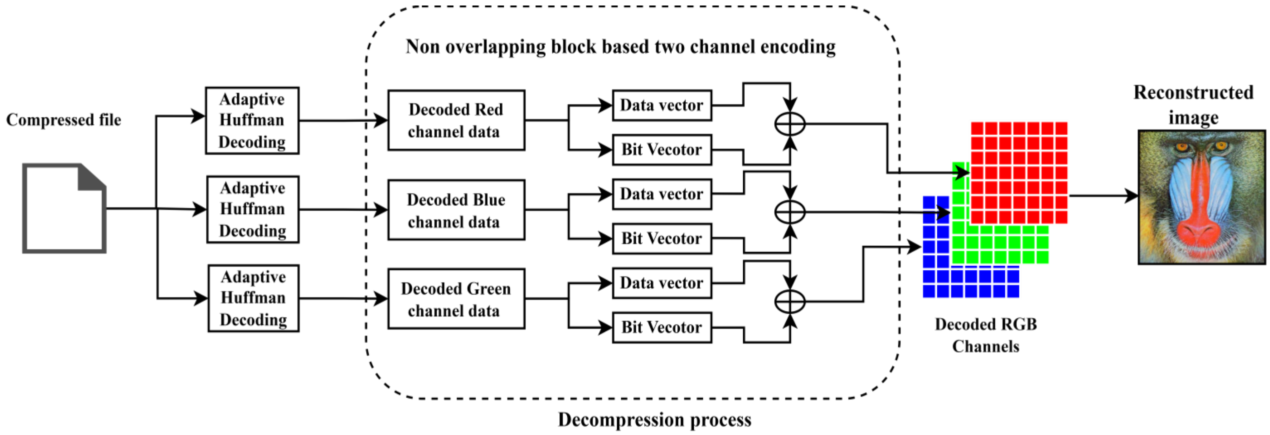

Fig. 2 Decoding process of the proposed method

3. Proposed Methodology

The proposed methodology for lossless image coding combines Prediction by Partial Matching (PPM), block-based two-channel coding, and adaptive Huffman coding, achieving efficient and accurate compression. Figures 1 and 2 show the detailed implementation of the encoding and decoding processes of this work.

Brief outline of the proposed methodology encoding and decoding steps as follows.

1. Image Acquisition and Preprocessing: The desired RGB image is input to the beginning of the process. The Prediction by Partial Matching (PPM) algorithm is then applied to enhance content and accelerate subsequent processing efficiency. The algorithm utilizes statistical modeling to reduce redundancy and incorporate the content of the image.

2. Channel Separation: The enhanced image is split into its RGB channels.

3. Block Division: Since each color channel divides into non-overlapping blocks of fixed size, such as 8x8 or 16x16 blocks of pixels.

4. Two-Channel Coding: Each block is encoded with two-channel coding. This process involves:

- Data Splitting: The pixel data of each block is broken up into two streams -- a bit stream and a data stream.
- Stream Merging: The block of information is effectively represented as an encoded stream that is merged from the bit and data streams.

5. Channel Data Encoding: All the blocks in a channel are encoded individually, and the encoded data of these blocks are combined to form a complete encoded representation of this channel.

6. Adaptive Huffman Coding: The combined encoded channel data is applied to adaptive Huffman coding. This entropy-based method achieves compression by dynamically selecting a coding scheme that matches the statistical characteristics of the data to produce the best possible compression performance.

7. Compressed file generation: Finally, the encoded data is merged from the red, green, and blue channels to produce the final compressed file.

The steps of the decompression process are represented below.

1. Compressed File Reading: Load the compressed file containing the encoded data and necessary metadata, including block size, two-channel coding parameters, and the adaptive Huffman coding tree.

2. Adaptive Huffman Decoding: Decode the compressed data using the provided adaptive Huffman coding tree to reconstruct the encoded channel data for the red, green, and blue channels.

3. Channel Data Decoding: For each channel (Red, Green, and Blue): Split the decoded channel data into individual blocks based on the block size (e.g., 8x8 or 16x16).

4. Two-Channel Decoding: For each block in a channel:

- Stream Splitting: Separate the merged bit stream and data stream for the block using metadata parameters.
- Data Reconstruction: Reconstruct the pixel data for the block from the separated streams using the inverse of the two-channel coding process.

5. Block Assembly and Channel Merging: Combine all the reconstructed blocks to recreate the complete channel data for each channel (Red, Green, and Blue). Merge the reconstructed Red, Green, and Blue channels to form the enhanced image data.

6. Reconstructed Image Output: Output the fully reconstructed RGB image, which is identical to the original input image.

3.1. Prediction by Partial Matching (PPM)

The prediction by Partial Matching (PPM) [17] is a statistical data compression technique. Given a sequence of previously observed symbols, it predicts the next symbol in that sequence. Lossless data compression algorithms are usually based on PPM. The working process of PPM is discussed below.

Context Modeling: The PP managing build model is the basis of it, and it is based on contexts, that is, substrings of previous data seen. If enough data has been observed, the longer the context, the more accurate the prediction.

Prediction: Each new symbol is checked in PPM for past contexts of different lengths. It predicts the probability distribution of the following symbol by using these contexts.
Encoding: An entropy coder, such as arithmetic coding, encodes the symbol once the probabilities have been computed.

Escape Mechanism: PPM uses an “escape” symbol if a symbol is not found in a given context, allowing the model to back off to a shorter context.

3.1.1. PPM in Image Enhancement

Traditionally, PPM is used in the context of text and data compression and can be adapted to image enhancement tasks [18]. It is capable of predicting pixel or feature values based on surrounding information.

Context Modeling in Images

- An image is represented as a 2D or 3D grid of pixels.
- PPM gives the predicted value of a target pixel based on the values of neighboring pixels (context).
- However, the context window size and shape (such as a 3x3 or 5x5 grid) are crucial for effective modeling.

Prediction Process

- Pixel Prediction: In the context, the patterns are observed, and PPM is proposed to predict pixel intensities or color values.
- Feature Prediction: PPM can predict features such as edges or textures for high-level enhancements.

Iterative Enhancement

- As we parse more of the image, the context models used by PPM improve iteratively in order to refine predictions.
- Matching patterns in context with observed data allows the method to fill in missing details, smooth the noise, or sharpen the edges.

Image enhancement using PPM is represented in Algorithm 1.

3.2. Two-Channel Coding Process

This technique divides the information into two streams: a bit stream and a data stream. Image intensity information is stored in the form of 0 or 1 in a bit stream. The intensity level is a piece of the data stream. Furthermore, the bit stream is combined with the data stream to form the encoded stream. Figure 3 shows the two-channel encoding process.

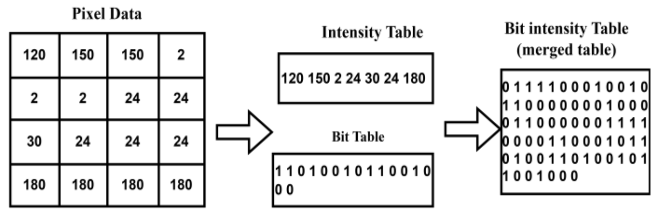

Fig. 3 Two-Channel encoding process

The detailed process of encoding and decoding of two-channel coding is represented in Pseudocode 1 and 2.

Algorithm 1. PPM_Image_Enhancement

Algorithm: PPM_Image_Enhancement

Input: *Image*
Output: *Enhanced image*
# Step 1: Initialize the PPM model
context_model = Create a PPM model with the specified order
enhanced_image = Initialize Image with the same size
# Step 2: Process the image
For each pixel in input_image:
context = Extract the context (neighboring pixels) around the given position
# Step 3: Predict the pixel value
predicted_value = Predict the next pixel value based on the context model
# Step 4: Calculate residual and enhanced value
Residual= actual_value - predicted_value
enhanced_pixel = predicted_value + residual
# Step 5: Update the model with the actual pixel value
Context_model Update the context model with the actual value
# Step 6: Store the enhanced pixel value
SetPixelValue= enhanced_pixel
# Step 7: Finalize processing
Generate enhanced_image

Pseudocode 1. Definition of function wo_channel_encoding

Function: Two-channel_encoding (Dblock)

Input: Dblock
Return: Data stream Ds and Bit stream Bs

1. InitializeBs=1 and Ds=first value of *Dblock*
2. Read next value, *Pv*, from *Dblock i*n sequential order
3. If current *Pv== Previous Pv* then
   Bs {Bs, 0}
   else
   Ds={Ds, Pv}
   Bs {Bs, 1}
4. Repeat steps 2 and 3 for all the values of *Dblock*
5. Return *Ds, Bs* vectors

end function

Pseudocode 2. Definition of function Two-channel_decoding

```
Function: Two_channel_decoding (Dstream,
Bstream )
Input: Dstream and Bstream
Return: Dblock
1. Initialize Dblock={}
CB first bit Bstream,
Cd=first value of Dstream
2. Assign Cd to datablock Dblock
      Dblock={Dblock, Cd}
3. Read the next bit from Bstream
   if Cb=1 then read next value from Dstream and
append to Dblock
       Cd=Dstream
       Dblock={Dblock, Cd}
   else if Cb=0 assign the already existing value in
Cd to Dblock
        Dblock={Dblock, Cd}
4. Repeat steps from 3 for all bits in the bit stream
Bstream
5. Return Dblock
end function
```

3.3. Adaptive Huffman Coding

Adaptive Huffman coding [19] is a dynamic Huffman coding, in contrast to static Huffman, which builds the tree depending on the pre-calculated symbol frequencies, but constructing the tree on the fly makes it a good fit for data streams with unknown or fluctuating statistics. The Huffman coding tree for adaptive Huffman coding changes dynamically in response to incoming data.

3.3.1. Adaptive Huffman Coding Working Process

Initialization:

- Begin with an initial Huffman tree (sometimes with all symbols having frequency zero).
- The symbols that have not been seen yet are handled with the special 'Not Yet Transmitted' (NYT) symbol.

Encoding Process:

- When a symbol is encountered:
  - If the symbol is new (not in the tree), output its code as the NYT symbol followed by the symbol's binary representation.
  - If the current tree has been constructed and a Huffman code for the current tree does not exist, then output the code for the current tree from this table.

Tree Update:

- Increase the symbol frequency count.
- Maintain the properties of the Huffman tree (closer to the roots are nodes with a higher frequency).
- Siblings' property maintenance helps keep the tree balanced through techniques like tree restructuring.

Decoding Process:

- The same adaptive Huffman algorithm is used. While decoding incoming data, it builds and updates its tree.

Algorithms 2 and 3 represent the proposed scheme encoding and decoding process.

Algorithm 2. Proposed lossless image coding- encoding process

```
Algorithm: Block-based lossless image coding
encoding process
Input: RGB Image Img
Output: Compressed File
1.   Preprocess Img using PPM -> Enhanced
     Image
2.   Split Enhanced Image into Channels: I_R,
     I_G, I_B
3.   For each channel I_C:
      a. Divide I_C into blocks of size 8x8 or
         16x16.
      b.Apply Two-Channel Coding to each block.
4.   Merge Two-Channel Coded Streams
5.   Apply Adaptive Huffman Coding on the
     merged stream
6.   Combine the compressed data of all channels.
7.   Save the compressed file.
```

Algorithm 3. Proposed lossless image coding- decoding process

```
Algorithm: Block-based lossless image coding
and decoding process
Input: Compressed File
Output: Decompressed RGB Image I.
1. Load compressed file and extract metadata.
2. Decode compressed stream using Adaptive
   Huffman Coding.
3. Split the decoded data into blocks based on
   the block size
4. For each block:
   a.   Apply inverse two-channel coding to
        reconstruct channels
5. Merge reconstructed blocks for I_R, I_G,
   I_B
6. Combine I_R, I_G, and I_B to reconstruct
   the RGB image I.
7. Return the decompressed image I.
```

4. System Setup and Database

The efficiency of image processing in this research is ensured by using an Intel Core i7-7500U dual-core 2.7 Ghz CPU and 8GB RAM. The software is developed using Java 22. The study utilizes the widely used USC-SIPI Image Database [20] for experimentation and evaluation of the proposed system. The database contains color images from various categories, grouped into volumes based on the fundamental characteristics of the images. These images vary in resolution, ranging from 256x256 to 1024x1024 pixels. Black and white images are stored at 8 bits per pixel and color images at 24 bits per pixel. The available volumes include: Textures - Texture mosaics, related images, and the Brodatz textures; Aerials - Aerial photographs of high altitudes; Miscellaneous - Mandrill and pepper images are popular; Sequences - Moving heads, flyovers, and moving vehicles. The database stores all of its images in TIFF format. In the study at hand, the TIFF files were converted into raw binary using TIFF software. Uncompressed and unencoded pixel intensity values can be read into the RAW format and are available for direct processing. This research aims to achieve significant image compression quantitatively while maintaining the visual quality and integrity of the contained content.

Fig. 4 Color images of size 512x512SIPI dataset

5. Results and Discussion

The proposed method has been evaluated using the color images available in the raw format of the dataset [20]. Sample test images are shown in Figure 4. The popular performance evaluation measures for the recommended and associated works are analyzed. The performance evaluation measures Compressed Size (CS), Compression Ratio (CR), Bits Per Pixel (BPP), Space Saving (SP), and Entropy are the factors considered to assess the performance of the proposed work [2, 21, 22].

1. *Compressed Size (CS)= The size of the image after it has undergone compression*.
2. $Compression\ \text{Ratio(CR)} = \frac{\text{Size before compression}}{\text{Size after compression}}$ (1)
3. $Space\ \text{Saving(SS)} = (1\text{-}\frac{\text{Size after compression}}{\text{Size before compression}}) \times 100$ (2)
4. $\text{Bits per pixel(BPP)} = \frac{\text{Number of bites}}{\text{Number of pixels}} \times 8$ (3)
5. $\text{Entropy} = \sum_{i=0}^{255} \mathrm{P_i} \log_2 \mathrm{P_i}$ (4)

Figure 5 shows a comparative study of the resulting compressed sizes when four different image compression techniques, i.e., TIFF format, BMP format, LZW method, and the proposed method, were applied to different input images. The graph illustrates the level of performance against the compressed size of each method, with the proposed method consistently achieving compressed sizes lower than those of the other methods. Remarkably, this advantage proves to be stronger when interruptions occur in other methods, particularly in terms of compressed size, which lends support to the notion that the proposed method is more responsive to the different attributes of images, thereby saving more space.

All methods perform relatively equally on medium-sized images, whereas on the input with higher resolution, the proposed method performs the best compared to others. Based on the analysis, compressed sizes are seen to be -12.38 percent, -0.0094 percent, and -27.43 percent in the case of TIFF, BMP, and LZW, and +26.04 percent in the proposed method. The minus values in TIFF, BMP, and LZW indicate that, in most instances, these methods will result in a file that is even larger than the original, but propose a real reduction in image size, which is evidence of better compression.

Figure 6 provides the comparative analysis of the Compression Ratio (CR) using different image compression techniques on corresponding input images, i.e., TIFF format, BMP format, LZW method, and the proposed method. The compression ratio is also a performance measure, indicating the effectiveness of a compression method, with a higher CR meaning greater compression effectiveness. It is evident from Figure 6 that the proposed method yields a better compression ratio compared to other methods, indicating its effectiveness in compressing data more efficiently while maintaining image quality.

Table 2. Analysis of CS and CR between the proposed method and TIFF, BMP, and LZW methods

S.No	Image Name	Size (in Bytes)	Compressed size (CS) in Bytes				Compression Ratio (CR)			
			TIFF format	BMP format	LZW method	Proposed method	TIFF format	BMP format	LZW method	Proposed method
1	airplane.raw (512x512)	7,86,432	7,04,550	7,86,488	8,85,264	4,90,788	1.1162	0.9999	0.8884	1.6024
2	baboon.raw (512x512)	7,86,432	10,08,078	7,86,488	10,44,408	6,84,183	0.7801	0.9999	0.7530	1.1494
3	couple.raw (256x256)	1,96,608	1,70,346	1,96,664	2,00,977	1,45,057	1.1542	0.9997	0.9783	1.3554
4	girl.raw (256x256)	1,96,608	1,90,264	1,96,664	2,21,842	1,47,138	1.0333	0.9997	0.8863	1.3362
5	lena.raw (512x512)	7,86,432	9,49,194	7,86,488	10,98,610	5,55,738	0.8285	0.9999	0.7158	1.4151
6	peppers.raw (512x512)	7,86,432	9,54,910	7,86,488	10,58,590	5,94,465	0.8236	0.9999	0.7429	1.3229
Average		5,89,824	6,62,890	5,89,880	7,51,615	4,36,228	0.9560	0.9999	0.8274	1.3636

Table 3. Analysis of BPP and SS between the proposed method and TIFF, BMP, and LZW methods

S.No	Image Name	Bits Per Pixel (BPP)				Space Saving (SS)			
		TIFF format	BMP format	LZW method	Proposed method	TIFF format	BMP format	LZW method	Proposed method
1	airplane.raw (512x512)	7.1671	8.0006	9.0054	4.9926	10.4118	-0.0071	-12.5671	37.5931
2	baboon.raw (512x512)	10.2547	8.0006	10.6243	6.9599	-28.1837	-0.0071	-32.8033	13.0016
3	couple.raw (256x256)	6.9314	8.0023	8.1778	5.9024	13.3575	-0.0285	-2.2222	26.2202
4	girl.raw (256x256)	7.7419	8.0023	9.0268	5.9871	3.2267	-0.0285	-12.8347	25.1617
5	lena.raw (512x512)	9.6557	8.0006	11.1756	5.6533	-20.6963	-0.0071	-39.6955	29.3343
6	peppers.raw (512x512)	9.7138	8.0006	10.7685	6.0472	-21.4231	-0.0071	-34.6067	24.4099
Average		8.5774	8.0011	9.7964	5.9237	-7.2178	-0.0142	-22.4549	25.9535

Table 4. Analysis of entropy of proposed method vs TIFF, BMP, LZW methods

S.No	*Image Name*	*Entropy*			
		RAW format	*TIFF format*	*BMP format*	*Proposed method*
1	airplane.raw (512x512)	6.2139	6.2139	6.2139	6.3559
2	baboon.raw (512x512)	7.7522	7.7522	7.7522	7.8421
3	couple.raw (256x256)	5.9313	5.9313	5.9313	6.2151
4	girl.raw (256x256)	6.3810	6.3810	6.3810	6.5042
5	lena.raw (512x512)	6.9685	6.9685	6.9685	7.0457
6	peppers.raw (512x512)	7.0583	7.0583	7.0583	7.1709
Average		6.7175	6.7175	6.7175	6.8557

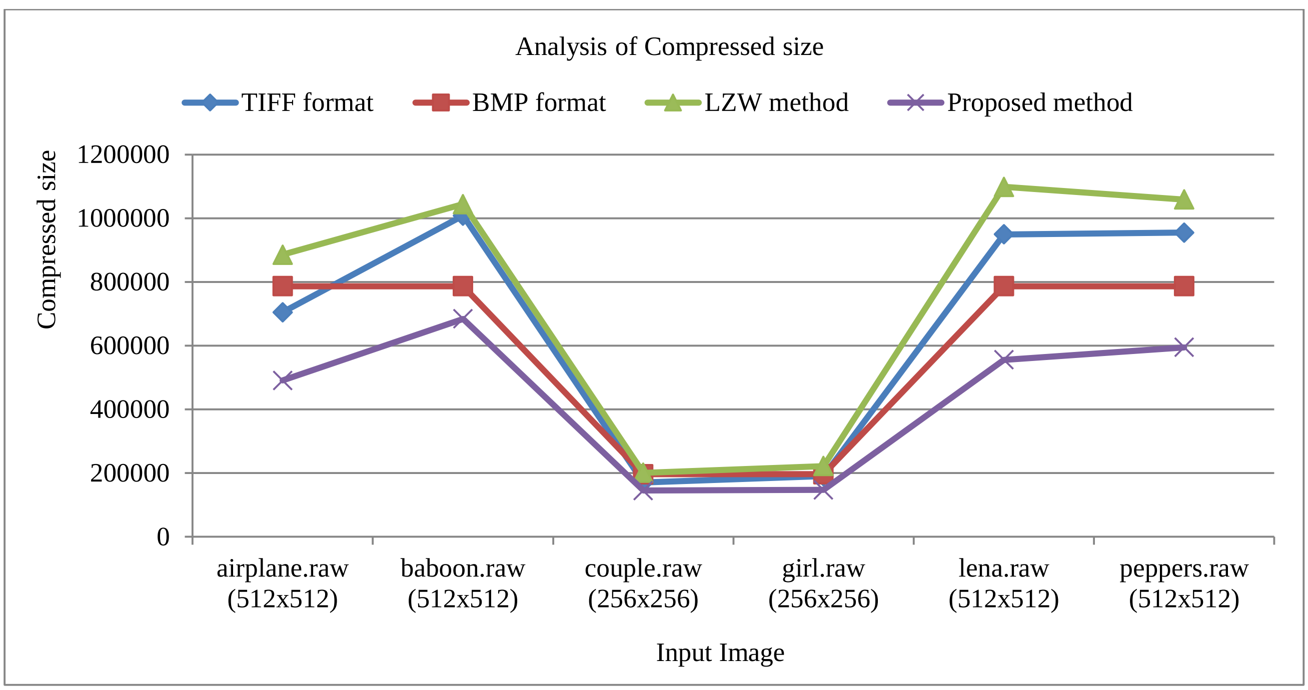

Fig. 5 Analysis of CS between the proposed method and TIFF, BMP, and LZW methods

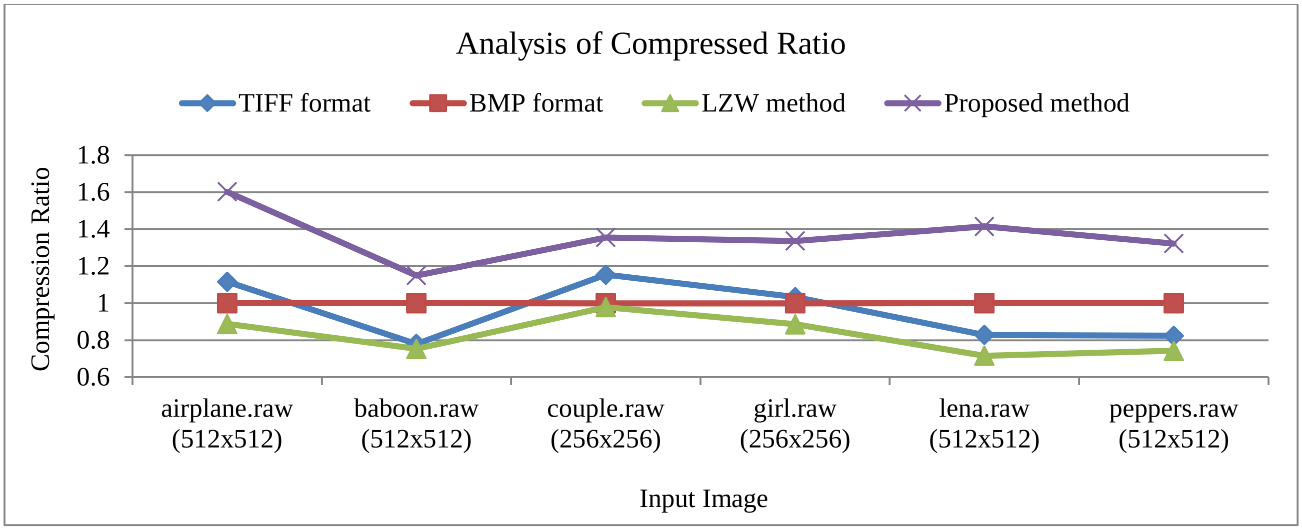

Fig. 6 Analysis of CR between the proposed method and TIFF, BMP, and LZW methods

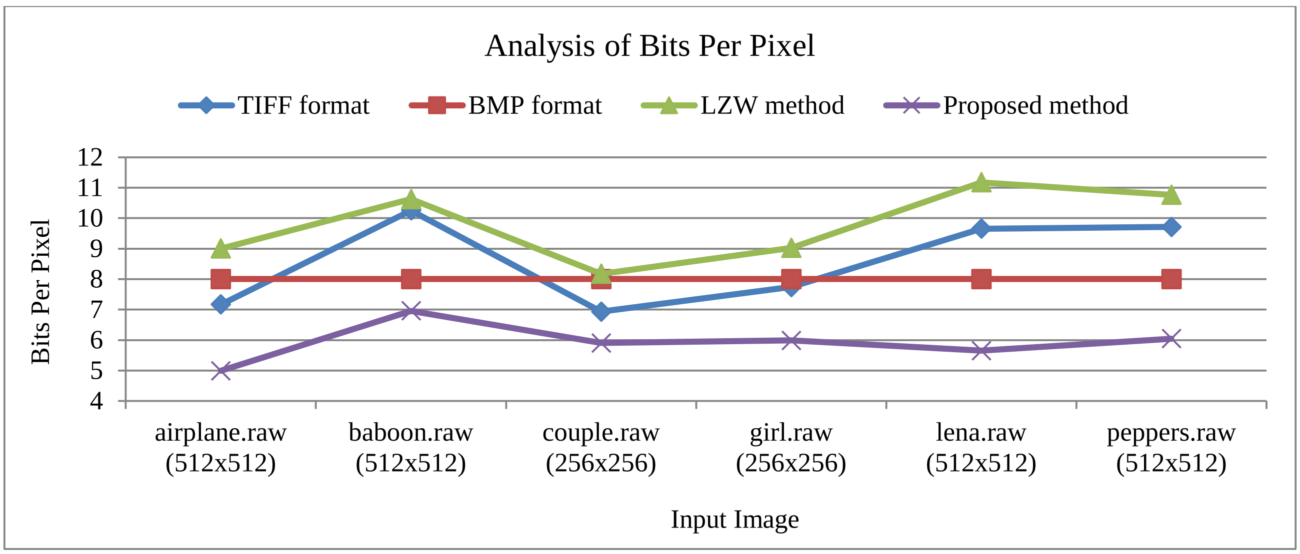

Fig. 7 Analysis of BPP between the proposed method and TIFF, BMP, and LZW methods

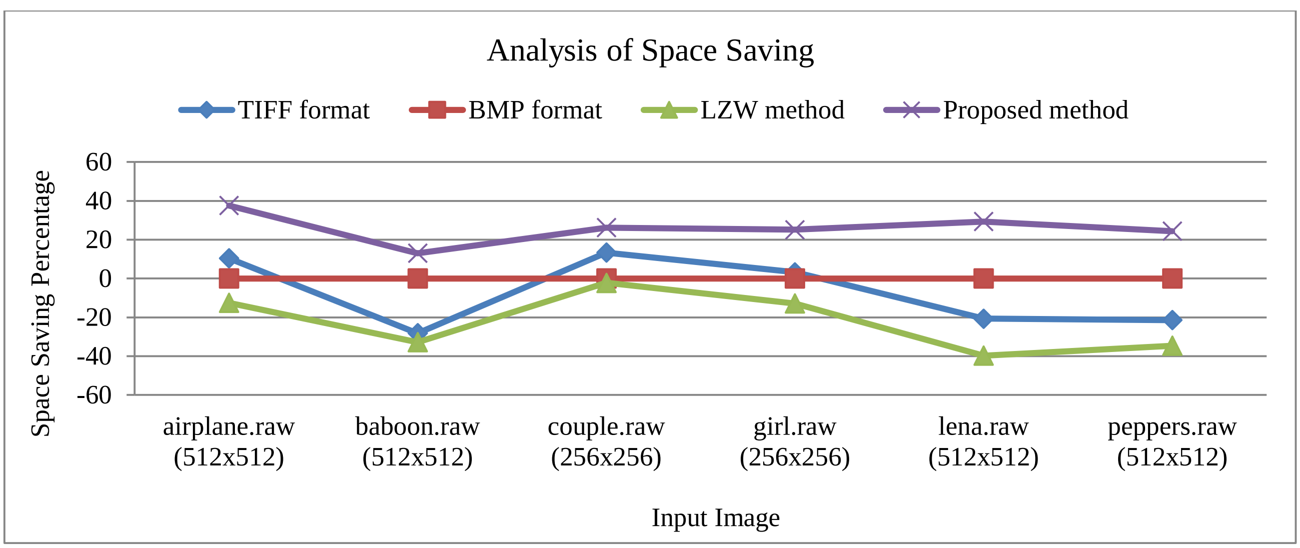

Fig. 8 Analysis of SS between the proposed method and TIFF, BMP, and LZW methods

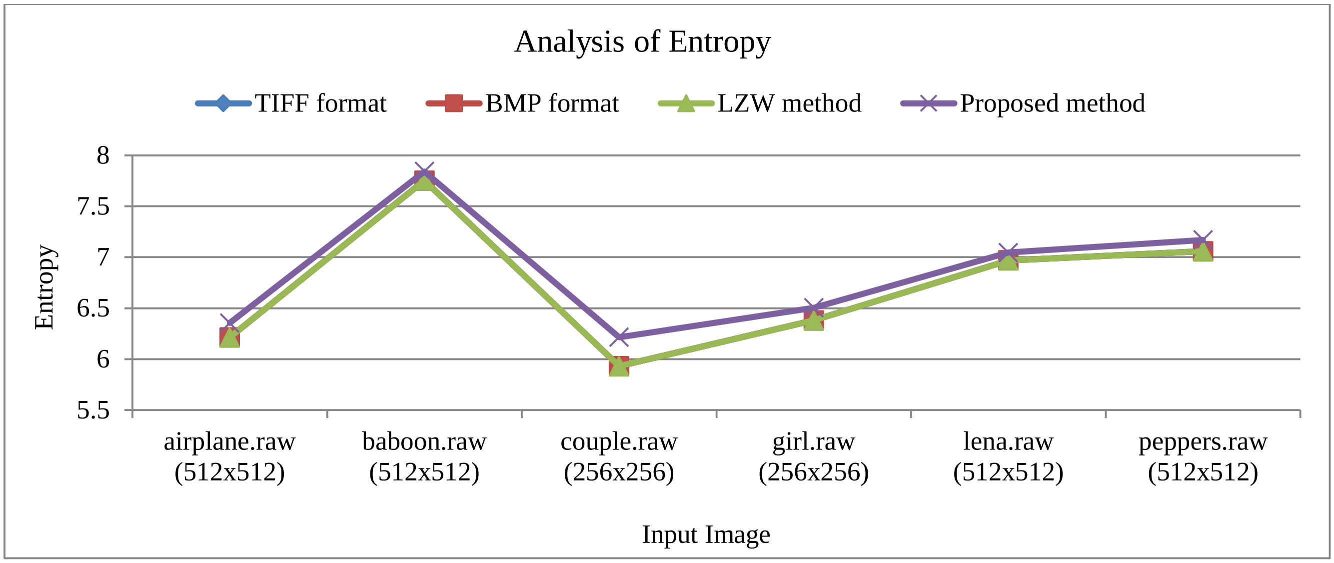

Fig. 9 Analysis of entropy between proposed method and TIFF, BMP, and LZW methods

Although the BMP method is consistent with all inputs, this is not the case with TIFF and LZW, which have variations; however, the proposed method shows its receptiveness and resilience in different images. It obtains 42.63%, 36.38%, and 64.80% greater CR than TIFF, BMP, and LZW methods, respectively, on average, which proves that it outperforms all of them overall in terms of compression.

To further examine the Bits Per Pixel (BPP) of various image compression algorithms, a few images such as TIFF, BMP, LZW and proposed algorithm are analyzed with respect to the output upon the input images of different size in Figure 7 BPP is a highly crucial factor in image compression as it is used to determine the average of the number of bits that one would need to represent the individual pixel in the compressed image and the low values of the BPP would mean a higher efficiency of the compression. The result of the proposed method has minimum values of BPP that are consistent and demonstrate the ability of the method to decrease the size of the data significantly without degrading the quality of the images. BMP on its part has a fixed BPP in all the inputs, as compared to TIFF and LZW, which have varying BPP that relies on images. In the analysis, it is possible to notice that the proposed method achieves 30.94, 25.96, and 39.53 percent smaller BPP against TIFF, BMP, and LZW methods, respectively, and thus is efficient and more adaptable to the characteristics of different types of image data.

As shown in Figure 8, a comparative study of the space saving percentage of different forms of image compression-TIFF, BMP, LZW, and the proposed one can be drawn on a variety of input images. Space saving is an important performance measure and expresses the space saving as a percentage based on the reduction in storage the compression gives in comparison to the space without compression. It is a measure of how well a step achieves proper file size while maintaining image quality. As the graph depicts, the maximum space savings using the suggested approach are always the highest, and on all the tested images, the approach performs much better in comparison with other techniques. Although BMP does not exhibit a significant decrease (flat line), results on TIFF and LZW are not consistent, and the suggested approach outperforms them in efficacy, more prolonged, and especially with regard to high-resolution or intricate images. On average, it complies with 459.57%, 600.56%, and 215.58% additional space economy over and above TIFF, BMP, and LZW, respectively, which touts its efficiency in terms of jurisdiction of resources and stewardship of information transmission.

The analysis shown in Figure 9 indicates that entropy values for different image compressions, i.e., TIFF, BMP, LZW, and the proposed approach, are subject to multiple input images. Another measurement widely used in image processing is entropy, which is used to determine the level of information or randomness in an image; the higher the entropy, the more complex and informative it is, and the lower it is, the simpler and more uniform it is. As we can see on the graph, none of the methods, including the suggested one, have significantly different entropy levels, which means that the compression algorithms do not lose the basic informational aspects of the images. Despite the fact that certain input images have obvious ripples and drops that can be observed because of the nature of variability, the proposed approach is very compatible with other formats, and the image integrity has not been lost anywhere. Moreover, the outlined approach illustrates a 2.06% increase in entropy that enhances its capacity to preserve the complexity of an image and even provides better compression rates.

6. Conclusion

In this paper, a lossless scheme of image compression is suggested that is based on Prediction by Partial Matching (PPM), two-channel block coding, and adaptive Huffman coding. The key aim is to reduce the duplication rate in digital images and retain whole image fidelity preservation. The Preprocessing Phase (using PPM) exploits the spatial correlation amongst the neighboring pixels that create an enormous amount of redundancy before the encoding phase. Then, a picture is divided into 8x8 non-overlapping blocks in order to permit a localized operation. The segmentation allows selective compression of smaller areas of the image, including the intra-block patterns and the inter-block dependencies. These blocks are then subjected to a channel coding mode, and they can be handled differently as far as structural components are concerned. This increases the compression in the achievement of inter-block redundancies that are majorly disregarded in the previous methods. Adaptive Huffman coding is then applied to entropy encoding the symbol probabilities, which are updated as time goes by. The results of the experiments indicate that the proposed method can perform better than the conventional lossy methods with respect to the Compression Ratio (CR), Bits Per Pixel (BPP), and Entropy, with an average of 46.97, 32.62, and 2.05 percent better than the most common lossless methods, such as TIFF and BMP, and LZW. Notably, the technique provides a perfect replica of the original image, therefore, satisfies the conditions of an authentic lossless compression. The method was tested on a range of various images, resolutions, and contents of the USC-SIPI Image Database. Its results are positive on its flexibility and regularity among image classes. The resultant solution to this study is omnivorous, adaptable, and domain-free, and can be implemented in conjunction with elements of in-medicine imaging, satellite imaging, and archiving facilities. It also has the possibility of future improvements by allowing the placement of content-aware or machine learning-based models of prediction.

Some further developments of the suggested method may include the subsequent introduction of deep-learning-based predictive networks to exploit the complex spatial structure of images further. Moreover, the real-time compression and addition of a parallel processing technique would be used to assist in increasing scalability. Adaptation of compressing multi-spectral and High Dynamic Range (HDR) images is another direction that is worth pursuing.

Data Availability

The images utilized during the implementation of the proposed work are available in the "USC-SIPI Image Database" https://sipi.usc.edu/database/database.php.

Funding Statement

This research work has not received any specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

References

[1] A.J. Hussain, Ali Al-Fayadh, and Naeem Radi, "Image Compression Techniques: A Survey in Lossless and Lossy Algorithms," *Neurocomputing*, vol. 300, pp. 44-69, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[2] Md. Atiqur Rahman, and Mohamed Hamada, "Lossless Image Compression Techniques: A State-of-the-Art Survey," *Symmetry*, vol. 11, no. 10, pp. 1-22, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[3] Yong Zhang, and Donald A. Adjeroh, "Prediction by Partial Approximate Matching for Lossless Image Compression," *IEEE Transactions on Image Processing*, vol. 17, no. 6, pp. 924-935, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[4] Muhammad Usama et al., "An Efficient Secure Data Compression Technique Based on Chaos and Adaptive Huffman Coding," *Peer-to-Peer Networking and Applications*, vol. 14, pp. 2651-2664, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] Antonin Descampe et al., "JPEG XS-A New Standard for Visually Lossless Low-Latency Lightweight Image Coding," *Proceedings of the IEEE*, vol. 109, no. 9, pp. 1559-1577, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[6] Jungan Chen et al., "Bit-Error Aware Lossless Image Compression with 2D-Layer-Block Coding," *Journal of Sensors*, vol. 2021, pp. 1-18, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] Shinichi Yamagiwa, Wenjia Yang, and Koichi Wada, "Adaptive Lossless Image Data Compression Method Inferring Data Entropy by Applying Deep Neural Network," *Electronics*, vol. 11, no. 4, pp. 1-13, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[8] Tungshou Chen et al., "A High Fidelity Authentication Scheme for AMBTC Compressed Image using Reference Table Encoding," *Mathematics*, vol. 9, no. 20, pp. 1-18, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[9] Tassnim Dardouri et al., "Dynamic Neural Network for Lossy-to-Lossless Image Coding," *IEEE Transactions on Image Processing*, vol. 31, pp. 569-584, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] Yuanchao Bai et al., "Deep Lossy Plus Residual Coding for Lossless and Near-Lossless Image Compression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 5, pp. 3577-3594, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[11] Gangtao Xin, and Pingyi Fan, "Soft Compression for Lossless Image Coding Based on Shape Recognition, *Entropy*, vol. 23, no. 12, pp. 1-22, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[12] Małgorzata Frydrychowicz, and Grzegorz Ulacha, "Two-Stage Golomb – Context-Adaptive Binary Arithmetic Coders Coding in Lossless Image Compression," *Advances in Science and Technology Research Journal*, vol. 18, no. 6, pp. 62-85, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[13] M. Sri Raghavendra et al., "Enhancing Lossless Image Compression through Smart Partitioning, Selective Encoding, and Wavelet Analysis, *International Journal of Electronics and Communication Engineering*, vol. 11, no. 5, pp. 207-219, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[14] Benjamin Lukas Cajus Barzen et al., "Accelerated Deep Lossless Image Coding with Unified Paralleleized GPU Coding Architecture," *2022 Picture Coding Symposium (PCS)*, San Jose, CA, USA, pp. 109-113, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] Xiaoxiao Liu e al., "An Improved Lossless Image Compression Algorithm Based on Huffman Coding," *Multimedia Tools and Applications*, vol. 81, pp. 4781-4795, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] Grzegorz Ulacha, Ryszard Stasinski, and Cezary Wernik, "Extended Multi WLS Method for Lossless Image Coding," *Entropy*, vol. 22, no. 9, pp. 1-24, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[17] Francisco Dantas Nobre Neto, Cláudio de Souza Baptista, and Claudio E.C. Campelo, "Combining Markov Model and Prediction by Partial Matching Compression Technique for Route and Destination Prediction," *Knowledge-Based Systems*, vol. 154, pp. 81-92, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[18] F. Irem Eyiokur et al., "Exposure Correction Model to Enhance Image Quality," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, New Orleans, LA, USA, pp. 676-686, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[19] Phyela Mbewe, and Sampson D. Asare, "Analysis and Comparison of Adaptive Huffman Coding and Arithmetic Coding Algorithms," *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, Guilin, China, pp. 178-185, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[20] Sijian Liang et al., "Joint Scrambling and Diffusion Image Encryption based on RNA Theory and Row–Column–Level Operation," *Journal of Electronic Imaging*, vol. 33, no. 5, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[21] Francisco F. Cunha et al., "Lossy Image Compression in a Preclinical Multimodal Imaging Study," *Journal of Digital Imaging*, vol. 36, pp. 1826-1850, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[22] Zhaoyang Jia et al., "Generative Latent Coding for Ultra-Low Bitrate Image Compression," *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, pp. 26088-26098, 2024. [CrossRef] [Google Scholar] [Publisher Link]