Poisson Wavelet Quantized Piecewise Regressive Distributed Coding for Image Compression and Transmission

[1]Ms. V.Prabavathi [2]Dr. M. Sakthi

[1]Research Scholar, Department of Computer Science, Nallamuthu Gounder Mahalingam College, Affiliated to Bharathiar University

Pollachi, Tamil Nadu,India.

ORCID:0000-0003-4325-1152

[2]Associate Professor, Department of Computer Science, Nallamuthu Gounder Mahalingam College, Affiliated to Bharathiar University

Pollachi, TamilNadu,India

Abstract:- Image compression is the process of minimizing the size of a digital image file while attempt to preserve its visual quality as much as possible. This file size reduction process is achieved by eliminating redundant or unnecessary information in an image. Therefore image compression is essential for various applications, including reducing storage space requirements, improving data transmission efficiency, and enhancing the user experience in multimedia applications. The existing compression method faces most important challenges to enhance the quality of the reconstructed image with minimum time and higher compression ratio. A novel technique called Poisson Wavelet Quantized Piecewise Regressive Distributed Coding (PWQPRDC) technique for achieving higher compression ratio. First, the numbers of natural images are collected from the dataset. The PWQPRDC first performs the Camargo's indexive gamma map filtering technique to preprocess the input image and to remove the noisy pixels from input image. Then the PWQPRDC technique performs the image compression to obtain the storage efficiency and transmission. The image compression process includes three different process namely Poisson Wavelet Transformation, Dead-Zone Quantization and Absolute piecewise regressive Geometric Distributed Coding. In PWQPRDC techniques, Poisson wavelet transformation is used for decomposing the input image into low and high frequency coefficients. After that, Dead-Zone Quantization is applied to quantize the low frequency coefficients and effectively rounding to the nearest discrete value. Finally, the Absolute piecewise regressive Geometric Distributed Coding is applied for mining the size of the quantized image with higher compression ratio. This compressed image is transmitted through the network channel. The decompression is performed as a reverse process of the compression technique such as Decoding, Dead-Zone De-Quantization and inverse Poission wavelet transformation. Experimental evaluation is carried out on factors such as peak signal-to-noise ratio (PSNR), compression ratio, compression time and Storage saved, with respect to image size. The analyzed results indicates that the superior performance of our proposed PWQPRDC techniques model when compared with existing methods.

Keywords: : *Image compression, Camargo's indexive gamma map filtering, Poisson wavelet transformation, Dead-Zone Quantization, Absolute piecewise regressive Geometric Distributed Coding*

1. Introductions

Image compression is the fundamental processes in the field of digital image transmission through the wireless channel. They play a crucial role in various applications, ranging from multimedia content delivery to medical imaging and remote sensing. The compression process involves reducing the size of a digital image while preserving its essential visual information. The main aim is to reduce the data required to represent an image efficiently. This reduction in data size is achieved through various mathematical techniques and algorithms that exploit redundancies and eliminate redundant details in the image. Compressed images occupy less storage space and manage huge image collections more efficiently. Therefore, the smaller size of image files is transmitted over networks more quickly and reducing latency. Several compression methods have been developed.

An integration of Discrete Wavelet Transform (DWT) and Huffman coding was developed in [1] to enhance image compression and facilitate image transmission over a channel. However, the method did not incorporate machine learning algorithms to optimize the encoding process for minimizing image size. A simultaneous image fusion and compression scheme (SFC) was developed in [2] employing quantization and Huffman encoding to improve visual image quality. However, it did not achieve higher compression ratios and minimum storage consumption.

In [3], a Lifting Scheme (LS) was employed with a Fully Connected Neural Network (FCNN) to encode input images. However, a significant challenge was the need to minimize the time consumed during image compression. The Edge-Aware Extended Star-Tetrix Transform was introduced in [4] for the compression of raw camera images. However, this approach did not improve the peak signal-to-noise ratio performance.

A Deep Neural Network (DNN)-based image compression method was developed in [5] to partition an image into multi-scale sub-bands using the Laplacian pyramid technique. But, it failed to optimize the image compression model for a specific visual analysis task. An Arabic Unicode text was compressed using the Lempel-Ziv-Welch (LZW) technique was designed in [6] to achieve efficient compression. But the multi-level encoding scheme with other compression techniques was not applied to improve the overall compression ratio.

An end-to-end learnt lossy image compression approach was designed in [7] with higher peak signal to noise ratio. But the lesser compression time was a major issue. A reversible data hiding method was developed in [8] for efficient image compression. But the machine learning based encoding was not applied to further enhance the compression ratio. A learned multi-resolution image compression method was introduced in [9] to divide the latent representations into the high-resolution as well as low-resolution (LR) bands. But the complexity of the method was not optimized.

An optimized JPEG-XT method (OPT_JPEG-XT) was developed in [10] to achieve improved compression ratio of medical images using discrete cosine transform (DCT) coefficients. However, it failed to develop a novel compression approach to enhance the performance.

1.1 Major contributions

The major contribution of the proposed PWQPRDC technique is discussed in the following lines,

- A novel PWQPRDC technique is proposed to enhance compression by applying Camargo's indexive gamma map filtering, Poisson wavelet transformation, Dead-Zone Quantization, and Absolute piecewise regressive Geometric Distributed Coding

- To enhance the peak signal to noise ratio, Camargo's indexive gamma map filtering technique is employed in the PWQPRDC technique improve image quality.
- To improve the image compression ratio, wavelet transformation based image decomposition, dead-zone quantization, and absolute piecewise regressive geometric distributed coding are applied to the PWQPRDC technique.
- To minimize compression time, Poisson wavelet transformation is applied to decompose the input image, resulting in various range blocks.
- Finally, a number of quantitative tests are performed using the proposed PWQPRDC techniqueand existing compression algorithms along with different performance metrics.

1.1 Paper outline:

The paper is organized into different sections. Section 2 discusses related works. Section 3 briefly describes the proposed PWQPRDC technique for natural image compression with a neat diagram. Section 4 provides information on the experimental settings and implementation details. In Section 5, quantitative analyses are presented using different performance metrics. Finally, Section 6 concludes the paper.

Related Works

Binary Inpainting Network (BINet) was developed in [11] for improved patch-based image compression. However, binary error correction was not performed. A hybrid coding framework was developed in [12] that integrate entropy coding and deep learning for image compression. An autoencoder-based lossy hyperspectral image compression method was developed in [13]. However, the performance of storage-saving capacity was not analyzed.

A lossless and lossy compression of disparity (depth) images were evaluated in [14] with lower resolution. But it failed to improve the compression performance of the model due to reduced spatial correlation. Context-based convolutional networks (CCNs) were developed in [15] for improved and valuable entropy modeling. However, the rate-distortion performance in image compression was not minimized. A content-weighted encoder-decoder model was designed in [16] for deep image compression. Image compression was performed in [17] based on optimized Toeplitz sensing matrices. But it did not minimize the computational complexity efficiently.

An efficient technique was developed in [18] for image compression and quality retrieval using matrix completion. A bit-error-aware lossless image compression approach was introduced in [19] with the help of bi-level coding for grayscale images. A general generative model was developed in [20] for image compression using an optimization encoder.

3. Methodology

With advancements in digital technology, the image acquisition capacity of devices has significantly increased. These images possess specific characteristics, such as sufficient spatial information in each pixel, high dimensionality, rapid velocity, various scales, large volume, and temporal information. The extensive size of these images poses a significant risk to their practical use in real-time transmission due to the wealth of information they contain. Therefore, there is a crucial need to develop and enhance data compression techniques to reduce both storage space and transmission time over the network, without any loss of data. Motivated by this, this paper introduces a novel PWQPRDC techniqueaimed at improving image compression performance with minimal time requirements.

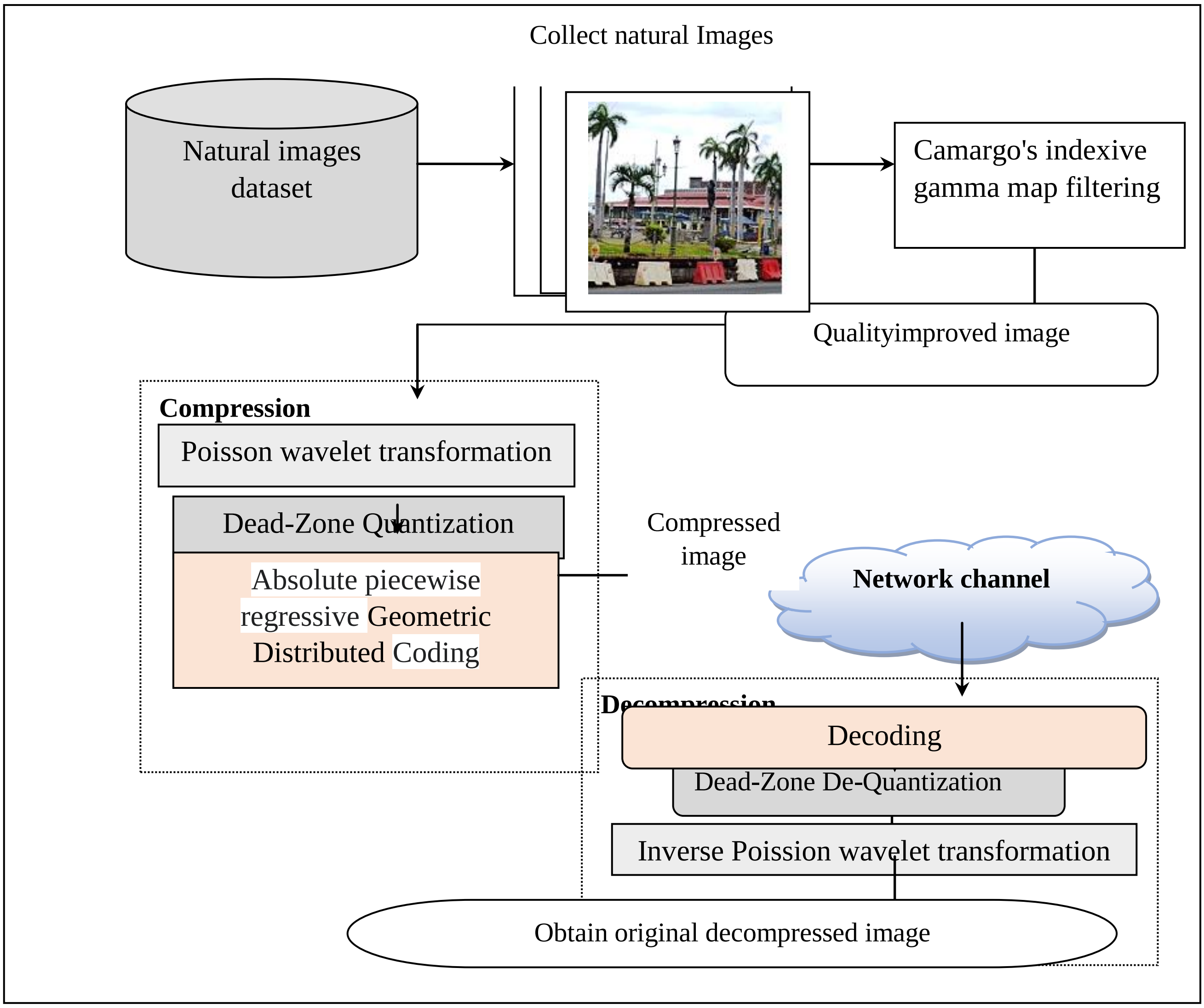


Figure 1 architecture of PWQPRDC technique for image compression and transmission

Figure 1 given above illustrates the architectural diagram of proposed PWQPRDC technique, which comprises five major processes namely image acquisition, preprocessing, transformation, quantization and encoding. As depicted in the figure, a number of natural images denoted as $NI_1, NI_2, \ldots, NI_n$ are collected from the image dataset. After the image acquisition, proposed technique performs the preprocessing using Camargo's indexive gamma map filtering technique refers to a series of operations to enhance the quality of the image, remove noise. In the third process,Poisson wavelet transformation is carried out to decompose the images into the different sub blocks. Followed by, Dead-Zone Quantization is a technique used in image and data compression. It involves dividing a decomposed image or data into a family of data compression codes. This method is particularly useful for reducing the size of data while maintaining an acceptable level of quality. Geometric Distributed Coding (GDC) is indeed used for image compression with the aim of minimizing the overall size of the image while achieving a higher compression ratio.

The detailed explanation of different process of the proposed PWQPRDC techniqueis presented in the following sub sections.

3.1 Image preprocessing

Image preprocessing refers to a series of operations applied to an input image before it performs a specific task. The fundamental process of image preprocessing is to enhance the quality of the image by eliminating the noise artifacts. Therefore, the proposed PWQPRDC techniqueuses the Gamma map filtering technique to enhance the image quality.

Gamma map filtering technique is used to correct the brightness or contrast of images. It involves applying a nonlinear adjustment to the pixel values in an image.

Let us consider the natural images $NI_1, NI_2, \ldots, NI_n$ and the pixelsare denoted by $x_1, x_2, x_3, \ldots x_m$. These extracted pixel values are assembled in a filtering window based on the gamma distribution. A filtering window, also known as a kernel is a small matrix used in various image processing operations to apply specific transformations to an image.

$$P = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} \quad (1)$$

Form (1), 3×3filter window or matrix 'P' is designed in which the image pixels are positioned in terms of rows (i) and columns (j). The pixels are organized in the ascending order and find the center pixel in the filtering window. If any even number of pixels in the center, then take average of these two values are considered as center value. Then the Camargo's indexive gamma map filtering technique is applied to window for removing the noise pixels.

The gamma map filtering technique is applied that describes the relationship between the pixels using Camargo's index. Camargo's index is a qualitative method to measure the likelihood between the pixels

The Camargo's index function is formulated as follows,

$$CI = 1 - \sum_{j=1}^{m} \left(\frac{|x_j - x_c|}{m} \right) \quad (2)$$

Where, CI indicates an index function, x_j denotes a neighboring pixels in filtering window, x_c denotes a center pixels, m indicates a number of pixels. From the analysis, the pixels that deviated from the center value are filtered. This process enhances the quality of image. The algorithm of Camargo's indexive gamma map filtering technique based image preprocessing is illustrated as follows,

Algorithm 1 : Camargo's indexive gamma map filtering technique

Input: Natural ImageDataset, Number of natural images$NI_1, NI_2, NI_3, \ldots, NI_n$

Output: preprocessed natural image

Begin

Step 1: Collect number of natural images$NI_1, NI_2, NI_3, \ldots, NI_n$

Step 2:For each input natural images (NI_i)

Step 3: Assemble the pixels in an $\mathbf{3 \times 3}$filtering window

Step 4: **Take the center pixel 'x_c'**

Step 5: Compute the likelihood byapplyingCamargo's index using (2)

Step 6: **Find noisy pixels**

Step 7: Remove the **noisy** pixels from the filtering wndow

Step 8:End for

End

Algorithm 1 describes the preprocessing of natural images using Camargo's index-based gamma map filtering technique. Firstly, a set of natural images is gathered from the dataset. Subsequently, the individual image pixels are placed within a filtering window, and the central pixel is identified. The Camargo's index is then applied to evaluate the similarity between the neighboring pixels and the central pixel's value. Pixels that exhibit significant deviations from the central value are identified as noisy and subsequently removed from the image. Consequently, the preprocessing results are used to obtain the quality improved image.

3.2 Image compression

Digital image compression minimizes an image file's size without notably reducing the image's quality. Transmission of such compressed images through communication channels requires lesser resources. The proposed PWQPRDC techniqueperforms three processes for image compression includes as follows,

- Poisson wavelet transformation based image decomposition,
- Dead-Zone image Quantization
- Absolute Moment Block Geometric Distributed Coding

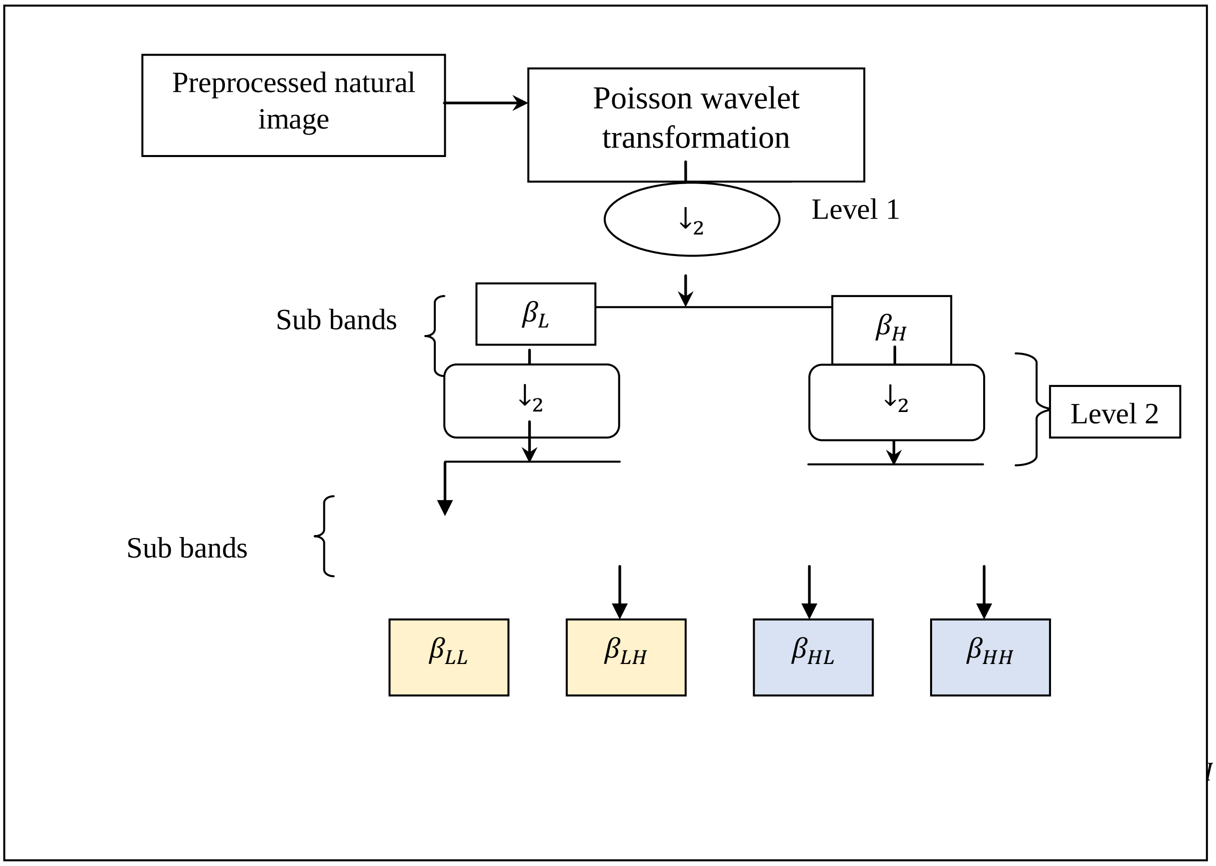
3.2 Image compression

Digital image compression minimizes an image file's size without notably reducing the image's quality. Transmission of such compressed images through communication channels requires lesser resources. The proposed PWQPRDC techniqueperforms three processes for image compression includes as follows,

- Poisson wavelet transformation based image decomposition,
- Dead-Zone image Quantization
- Absolute Moment Block Geometric Distributed Coding

3.2.1 Poisson wavelet transformation based image decomposition

Image decomposition is a process of breaking down a given preprocessed natural image into its different subbands, typically to analyze or process them separately. The decomposition process involves splitting the input image into a sequence of low-pass and high-pass filtered iterations, often referred to as subbands, which represent various frequency components of the image. This decomposition process is repeated multiple times to create a multiresolution representation of the image. The most common wavelet transform is widely used to remove the high-frequency components of the image, which are less important for the human visual system. The remaining low-frequency components are then used for the encoding process, helping to minimize storage consumption.



transform is applied as follows,

$$Y_T = \frac{1}{\sqrt{s}} \sum NI\ (t) * \varphi\ (t) \quad (3)$$

$$\varphi\ (t) = \frac{1}{2\pi}(1 - it)^{k+1} \quad (4)$$

Where, Y_T denotes a wavelet transformed output, s denotes a scaling function, $NI(t)$ denotes a input preprocessed image at time 't', $\varphi\ (t)$ denotes a mother wavelet,

Based on the transformation process, the input preprocessed natural image are decomposed into sequence of two levels such as low (β_L) and high(β_H) pass filters in both horizontal and vertical directions. For next consecutive levels, downsampling (i.e.$\downarrow_2$) is carried out and the output of each level generates four sub-blocks $\beta_{LL}, \beta_{LH}, \beta_{HL}, \beta_{HH}$. As a result, the different sub band images are obtained after the image decomposition process. The output of the decomposition is given to the next quantization process.

3.2.2 Dead-Zone Quantization

Quantization in image processing is carried out after decomposition. It is an essential step in reducing the amount of data needed to represent the image. Once the decomposition is applied, the image is split into various frequency subbands, each represented by a set of wavelet coefficients. These coefficients are positive or negative and it has a wide range of values. Quantization involves mapping the continuous transformed coefficients to a finite set of discrete values. This process aims to reduce the storage requirements during the transmission which directly impacts the perceived image quality in compressed images.

When a coefficient falls within a specific interval, it is mapped or quantized to the representative value of that interval. The main aim is to reduce the precision of the coefficients, effectively rounding them to the nearest discrete value within that interval.

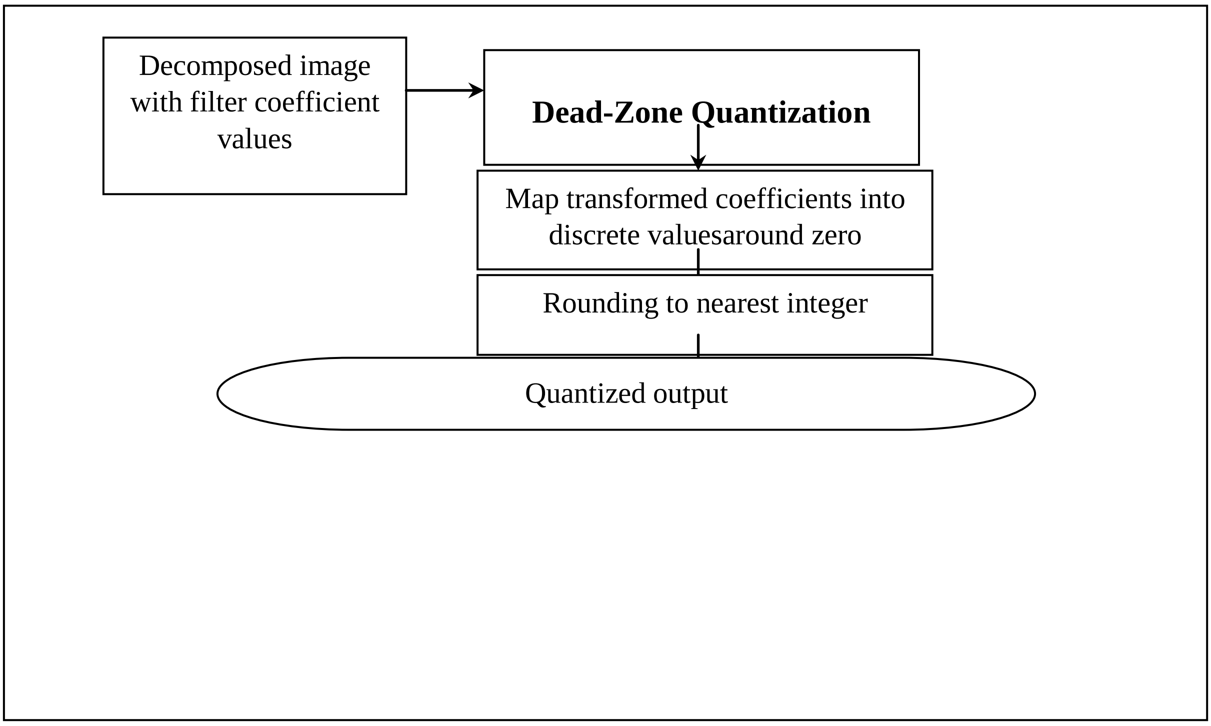


Figure 3 illustrates the block diagram of natural image quantization using Dead-Zone Quantization, wherein the coefficients' values are mapped into discrete values around zero. This process involves removing the high-frequency components of the image and quantizing the low-frequency components, which represent the image's general structure and major features.

$$Q_o = \vartheta_s(W(NI)).max\left(0,\left\lfloor \frac{|W(NI)|-\left(\frac{\delta}{2}\right)}{\Delta}+1\right\rfloor\right) \quad (5)$$

Where, Q_o denotes a Quantization outcomes, ϑ_s denotes a sign function of the wavelet coefficients' $W(NI)$' as given below,

$$\vartheta_s = \begin{cases} -1, & W(NI) < 0 \\ 0, & W(NI) = 0 \\ +1, & W(NI) > 0 \end{cases} \quad (6)$$

From (),max represents the maximum allowable quantized value i.e. around '0', δ denotes a dead zone width, $\lfloor \quad \rfloor$ symbol denotes a floor function used to give the output less than or equal to input coefficient value in the nearest round process, Δ denotes a quantization step size (i.e. Δ= 1 for rounding to the nearest integer]. In this way, the quantized outcomes are obtained

3.2.3 Absolute Moment Block Geometric Distributed Coding

Finally, the quantized coefficients are encoded to minimize the size of the images using Absolute Moment Block Geometric Distributed Coding. It is a types of lossy compression for grayscale images while maintaining the mean and deviation from the current pixels.

For each block the mean and standard deviation of the pixel values are computed. These analysis results are changes from block to block of original image. The compression is performed as follows:

$$f(x_{ij}) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x_{ij}-M}{\sigma}\right)^2\right) \quad (7)$$

The piecewise regression is a machine learning technique for estimating the relationships between pixels and the mean. Based on relationship, the two possible outcomes (i.e. 0 or 1) are obtained by setting the breakpoint i.e. mean value. The breakpoint value is defined beyond or below which desired effect occurs.

$$Y_{com} = \begin{cases} 1, & x_{ij} > M \\ 0, & x_{ij} < M \end{cases} \quad (8)$$

Where, $f(x_{ij})$ denotes a Geometric Distribution between the mean 'M'of the pixels 'x_{ij}' in the quantized results. Y_{com}denotes a compressed outcomes, if the pixels values greater than mean value, then is assigned the value 1, otherwise 0. Finally, the compressed outcomes are obtained with lesser size. The algorithmic process of image compression is given below,

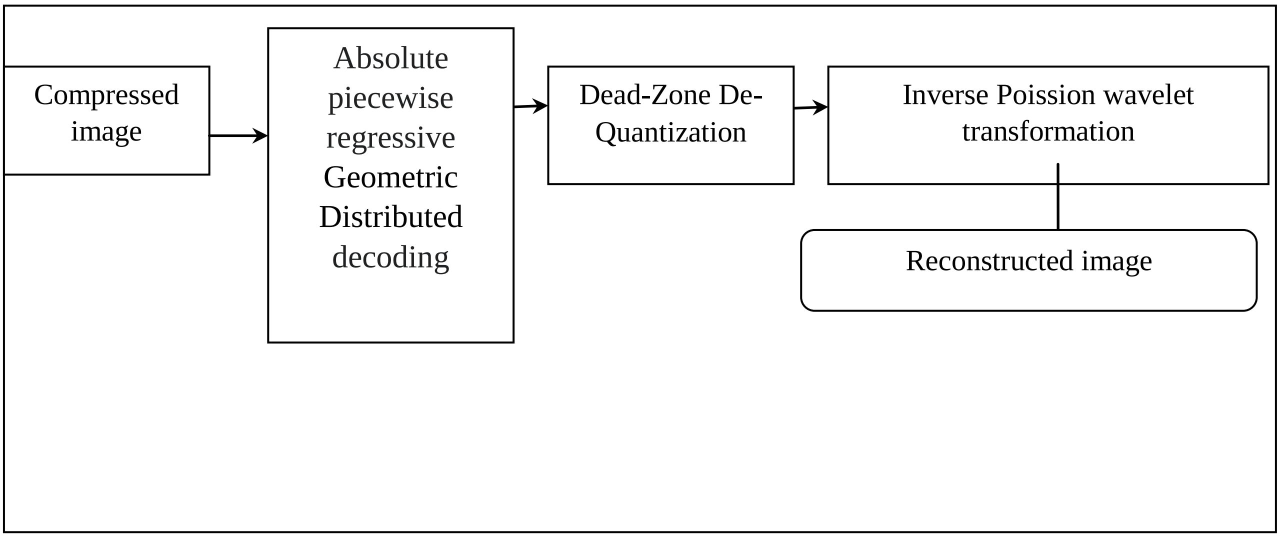
Algorithm 1: Image compression
Input: Natural image dataset, preprocessed images $NI_1, NI_2, NI_3, \dots NI_n$
Begin **1.** **Number of** preprocessed images $NI_1, NI_2, NI_3, \dots NI_n$ **taken as** input **<u>\\ Poisson wavelet transformation</u>** **2.** **For each** image NI_i **3.** Apply Poisson wavelet transform '$W(NI)$' **4.** Decompose the image into low and high frequency coefficients $\beta_{LL}, \beta_{LH}, \beta_{HL}, \beta_{HH}$ **5.** **End for** **<u>\\ Dead-Zone Quantization</u>** **6.** **For each** coefficients **7.** Perform Dead-Zone Quantization process using (5) **8.** Remove the high-frequency components **9.** Quantize the low-frequency components **10.** Rounding nearest integer value **11.** **End for** **<u>// Absolute piecewise regressive Geometric Distributed Coding</u>**

```
      **For each** pixels '$x_{ij}$' quantized image
   Measure mean '$M$' and deviation '$\sigma$'
      **If** ($x_{ij} > M$) **then**
        **Coded** the binary value as '1'
     **else**
        **Coded** the binary value as '0'
   **End if**
   End for
   **Return** (compressed image)
**End**
```

As depicted in algorithm 1, the image compression process consists of three steps, as outlined below. Firstly, the preprocessed image is taken as input and subjected to a Poisson wavelet transform to decompose the input image into its range subbands. Next, Dead-Zone Quantization is applied to quantize the low-frequency components while removing the high-frequency components. Finally, the Absolute Moment Block Geometric Distributed encoding is employed to encode the images, thereby reducing the overall size of the image. Consequently, compressed images are generated with a higher compression ratio.

3.3 Image decompression

In the proposed technique, image decompression is performed in order to obtain the reconstructed natural image. The image decompression is the reverse process of image compression.



taken as input.

3.3.1 Absolute moment block geometric distributed decoding

To reconstruct the image, elements assigned a 0 are replaced with the 'u' and elements assigned a 1 are replaced with 'v 'value. First, absolute moment block geometric distributed decoding is performed as given below,

$$x_{ij} = \begin{cases} p \,, & Y_{com} = 0 \\ q \,, & Y_{com} = 1 \end{cases} \quad (9)$$

$$u = M - \sigma \sqrt{\frac{p}{m-p}} \quad (10)$$

$$v = M + \sigma \sqrt{\frac{m-p}{p}} \quad (11)$$

Where, x_{ij} denotes a pixels, σ indicates a deviation, m denotes a total number of pixels in the block, 'p' denotes a number of pixels greater than the mean 'M'. The decoder's replaces $1's$ and $0's$ with the estimated value, while the encoder is additionally responsible for calculating the mean, standard deviation, and the two values for correction.

3.3.2 Dead-Zone De-Quantization

The dequantization refers to a process where quantized data is converted back into a high-frequency and low-frequency components. The general reconstruction of dead-zone quantizer is given blow,

$$D_Q = \vartheta_s(Q_o)\,.\,max\left(\frac{\delta}{2} + \Delta.\ (|Q_o| - 1 + R_s)\right) \quad (12)$$

Where, D_Q dequantization results, ϑ_s denotes a sign function of the wavelet coefficients' $W(NI)$', Q_o denotes a quantized outcome, δ denotes a dead zone width, Δ denotes a quantization step size, R_s denotes a reconstruction offset value in the range of 0 to 1 as a fraction of the step size.

3.3.3 Inverse Poisson wavelet transforms

The Inverse Poisson Wavelet Transform is a mathematical operation that involves reversing the Poisson wavelet transform. In order to get the original image

$$NI = \frac{1}{\varphi}\sum C_f(t) * \frac{1}{\sqrt{s}}\ \varphi\ (t) \quad (13)$$

Where, NI denotes a original preprocessed image, s denotes a scaling function, $C_f(t)$ denotes a Poisson wavelet coefficients obtained during the forward wavelet transform, φ denotes a constant, $\varphi\ (t)$ denotes a mother wavelet. In this way, the image decompression is obtained. The algorithmic process of Image Decompression is shown below,

Algorithm 2: Image decompression
Input: compressed natural images $CNI_1, CNI_2, CNI_3, \ldots CNI_n$
Begin **Step 1:** compressed natural images $CNI_1, CNI_2, CNI_3, \ldots CNI_n$ taken as input **<u>Absolute moment block geometric distributed decoding</u>**

Step 2: for each compressed image CNI_i

Step 3: Perform decoding using (9)

Step 4: Reconstruct the quantized image

Step 5: end for

<u>Dead-Zone De-Quantization</u>

Step 6: for each quantized **image**

Step 7: Perform **dequantization** using (12)

Step 8: Reconstruct the wavelet components

Step 9: end for

<u>Inverse Poisson wavelet transforms</u>

Step 10: for each wavelet components

Step 11: Perform **Inverse transform** using (13)

Step 12: Reconstruct the original preprocessed image

Step 13: end for

End

Algorithm 2 describes the process of image decompression to get original image at receiver end. The algorithm initially takes the input as a compressed natural image. Then it performs a decoding process for converting an encoded image into the original quantized image. Followed by, De-Quantization is performed to reconstruct the low and high frequency wavelet components. After that, the inverse Poisson wavelet transform is applied to reconstruct the original preprocessed image. As a result, the reconstructed quality improved natural image is obtained.

4. EXPERIMENTAL SETUP

This section describes the evaluation campaign of the proposed PWQPRDC technique and the existing DWT and Huffman coding method [1] and SFC scheme [2].

4.1 Implementation details

In this section, experimental assessment of the PWQPRDC technique and the existing DWT and Huffman coding method [1] and SFC scheme [2] and is implemented in Python. In order to perform the simulation, Nature Images dataset are collected from the https://www.kaggle.com/datasets/prasunroy/natural-images. Theimage dataset consists of Natural Scenes around the world. The numbers of images are taken from the training folder. In this folder, 14034 150x150 Images in seg_train folder for training spread evenly across 6 categories such as buildings' -> 0, 'forest' -> 1, 'glacier' -> 2, 'mountain' -> 3, 'sea' -> 4, and 'street' -> 5. For each categorizes contains the more number of images.

Table 1 image details in seg_train folder

S.No	categories	Number of images
1	buildings	2191
2	forest	2271
3	glacier	2404
4	mountain	2512
5	sea	2274
6	street	2382

5. Performance Evaluation Measures

This section provides reports the different performance metrics such as Peak signal-to-noise ratio, compression ratio, compression time, and storage saving.

Peak signal-to-noise ratio: it used as a metric for evaluating the compression capability of algorithms. It quantifies the quality of noise removal by assessing the mean square error (MSE), which is calculated as the difference between the pixel values in the original image and those in the accurately preprocessed image.

$$PSN = 10 * \left[log_{\,10}\left(\frac{255^2}{MSE}\right)\right] \qquad (14)$$

$$MSE = \sum(NI_{UC}(size) - NI_C(size))^2 \quad (15)$$

Where PSN denotes a Peak signal-to-noise ratio, MSE denotes a mean square error, $NI_{UC}(size)$ indicates original or uncompressed natural images, $NI_C(size)$denotes the compressed natural images. The peak signal-to-noise ratio is measured in decibels (dB).

Table 2 Comparison of peak signal to noise ratio

Natural images	Original Image Sizes (KB)	Peak signal to noise ratio (dB)		
		PWQPRDC	DWT and Huffman coding method	SFC scheme
Image 1	11.09	65.85	60.52	58.88
Image 2	16.8	58.58	55.06	52.86
Image 3	23.3	60.17	56.30	53.32

Image 4	19.7	56.08	52.56	49.54
Image 5	18.46	58.30	54.87	48.48
Image 6	14.12	61.28	56.76	54.68
Image 7	12.72	63.52	58.30	55.87
Image 8	15.09	62.11	53.97	49.34
Image 9	13.76	59.18	55.06	52.42
Image 10	20.61	62.55	57	54.15

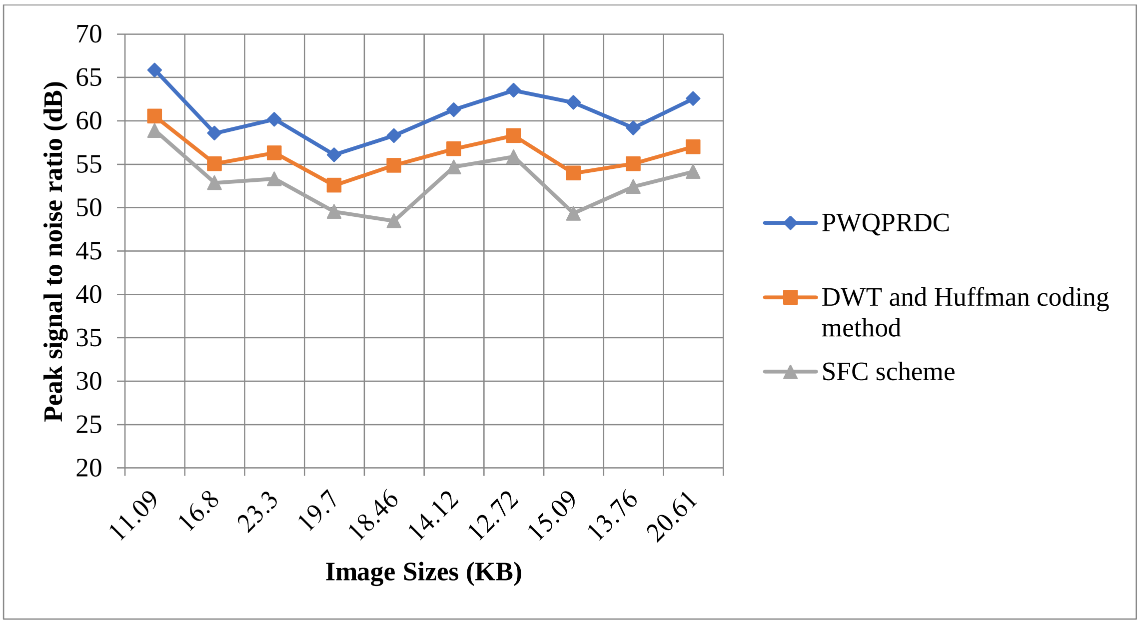


Figure 5 peak signals to noise ratio chart versus image size

Figure 5 illustrates the performance results of the peak signal-to-noise ratio concerning various sizes of input natural images, measured in kilobytes (KB) and collected from the dataset. Three methods have been applied for peak signal-to-noise ratio measurement such as the PWQPRDC technique, the existing DWT and Huffman coding method [1], SFC scheme [2]. Among these approaches, the PWQPRDC technique demonstrates higher performance to obtain the peak signal-to-noise ratio. This improvement in the PWQPRDC technique is achieved by the application of Camargo's indexive gamma map filtering technique. Initially, the natural image pixels are gathered within a filtering window. Camargo's index is then employed to compute the likelihood measure between the pixels and the central pixels, facilitating the removal of noise pixels and the enhancement of image quality. The average of ten results indicates that the performance of the PWQPRDC technique is enhanced by 8% compared to the existing [1] method and by 15% when compared to [2].

Compression ratio: This metric is determined by the ratio between the size of the original image and the size of the compressed image. It is mathematically calculated as follows,

$$CR = \left[\frac{NI_{UC}(size)(KB)}{NI_C(size)\,(KB)}\right] \quad (16)$$

Where, CR denotes a compression ratio, $NI_{UC}(size)$ indicates original or uncompressed natural images, $NI_C(size)$ denotes the compressed natural images. Higher the compression ratio, the method is said to be more efficient.

Table 3 Comparison of compression ratio

Natural images	Original Image Sizes (KB)	Compression ratio		
		PWQPRDC	DWT and Huffman coding method	SFC scheme
Image 1	11.09	1.405	1.339	1.301
Image 2	16.8	1.222	1.144	1.126
Image 3	23.3	1.239	1.174	1.131
Image 4	19.7	1.331	1.164	1.138
Image 5	18.46	1.375	1.216	1.174
Image 6	14.12	1.440	1.227	1.201
Image 7	12.72	1.394	1.216	1.195
Image 8	15.09	1.418	1.265	1.220
Image 9	13.76	1.426	1.317	1.300
Image 10	20.61	1.284	1.25	1.226

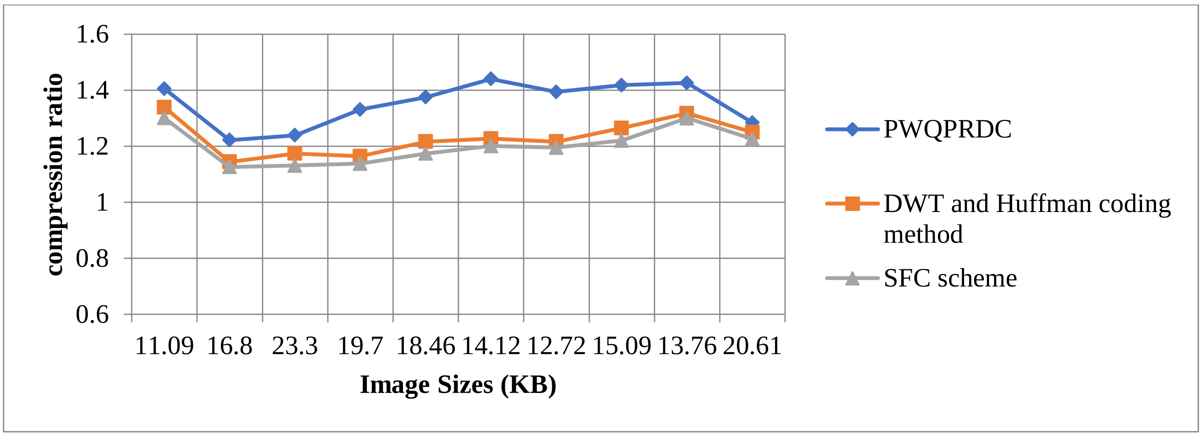


Figure 6 compression ratio chart versus image size

Figure 6 illustrates the performance results concerning compression ratios based on the sizes of natural images. The graph portrays the input image sizes on the 'x' axis and the corresponding compression ratio performance on the 'y' axis. These compression ratios were assessed using three distinct compression methods. Notably, the graph emphasizes that the employment of the PWQPRDC technique increased the compression ratios. This is because of effective execution of three different processes in generating compressed images: wavelet transformation, quantization, and coding techniques. The process begins with Poisson wavelet transformation, which decomposes the input image into various subbands containing both low and high-frequency components. Following this, Dead-Zone Quantization is applied to quantize the low-frequency components. Finally, the piecewise regression with encoding is utilized to encode the values, thereby reducing the image's size. As a result, the compression ratio increased by 10% and 13% when compared to the DWT and Huffman coding method [1], as well as the SFC scheme [2].

Compression time: It is measured by the time consumption of the image compression which includes three process namely wavelet transformation, quantization and coding. The formula for compression time is given below,

$$CT = time\ [T_W + Q_0 + Y_{cod}] \quad (17)$$

From (17),CT indicates a Compression time, ‘NI indicates natural images, T_W denotes wavelet transformation,Q_0 quantization, Y_{cod} denotes a coding. The Compression time is measured in milliseconds (ms).

Table 4 Comparison of compression time

Natural images	Original Image Sizes (KB)	compression time (ms)		
		PWQPRDC	**DWT and Huffman coding method**	**SFC scheme**
Image 1	11.09	13.65	15.85	18.36
Image 2	16.8	18.25	20.36	22.85
Image 3	23.3	25.41	28.85	30.45
Image 4	19.7	22.3	26.65	28.11
Image 5	18.46	20.69	25.96	27.96
Image 6	14.12	16.52	18.12	20.85
Image 7	12.72	14.33	16.85	18.60
Image 8	15.09	18.96	20.2	22.85
Image 9	13.76	16.85	17.85	19.68
Image 10	20.61	24.74	26.52	28.68

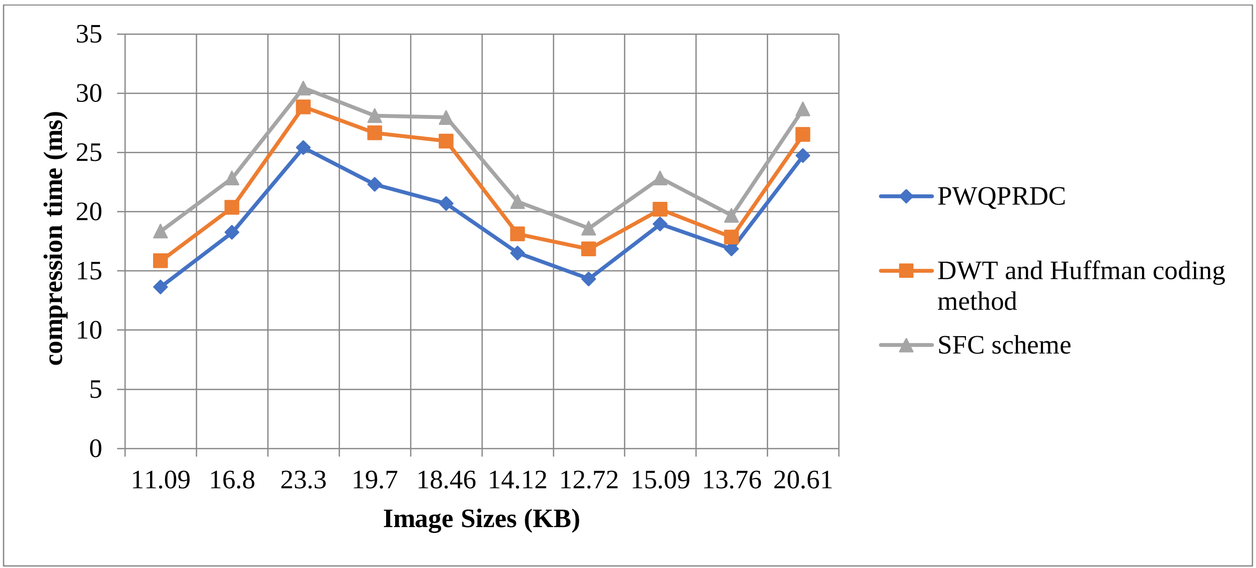


Figure 7 compression time chart versus image size

Figure 7 illustrates the relationship between compression time and the size of a natural image, measured in KB. A quantitative comparative analysis demonstrates a significant reduction in compression time when employing the PWQPRDC technique, in contrast to existing methods. Let's consider a statistical evaluation using an $11.09KB$ image size in the first iteration. Specifically, the PWQPRDC technique achieved a compression time of $13.65ms$, while the other two methods referenced as [1] and [2] showed compression times of $15.5ms$ and $18.36ms$, respectively. Similarly, various performance outcomes were obtained with different sizes of input image and results are compared. The results exhibit non-linearity due to variations in input image sizes. A comprehensive comparison of ten results reveals that the PWQPRDC technique reduces compression time by 12% and 20% when compared to [1] and [2], respectively. This reduction is achieved due to the application of image preprocessing using Camargo's indexive gamma map filtering technique before image compression. This preprocessing step helps minimize the time required for image compression. Furthermore, the proposed transformation and quantization, based on piecewise regression coding, further reduce the time needed to generate the compressed image.

Storage saved: Storage saved refers to the amount of storage space that has been conserved or freed up as a result of image compression. It quantifies the reduction in storage requirements achieved by compression, allowing for more efficient use of storage resources.

$$SS = NI_{UC}(size) - NI_{C}(size) \quad (18)$$

From (18),SS indicates a storage saved, '$NI_{UC}(size)$ indicates original or uncompressed natural images, $NI_{C}(size)$denotes the compressed natural images. It is measured in Kilobytes (KB).

Table 5 Comparison of storage saved

Natural images	Original Image Sizes	Storage saved (KB)		
		PWQPRDC	DWT and Huffman	SFC scheme

	(KB)		coding method	
Image 1	11.09	3.16	2.75	2.5
Image 2	16.8	3	2.06	1.82
Image 3	23.3	4.45	3.41	2.65
Image 4	19.7	4.8	2.7	2.3
Image 5	18.46	4.95	3.2	2.6
Image 6	14.12	4.25	2.55	2.29
Image 7	12.72	3.55	2.21	2.01
Image 8	15.09	4.39	3.06	2.57
Image 9	13.76	4.03	3.21	3.04
Image 10	20.61	4.52	4.05	3.71

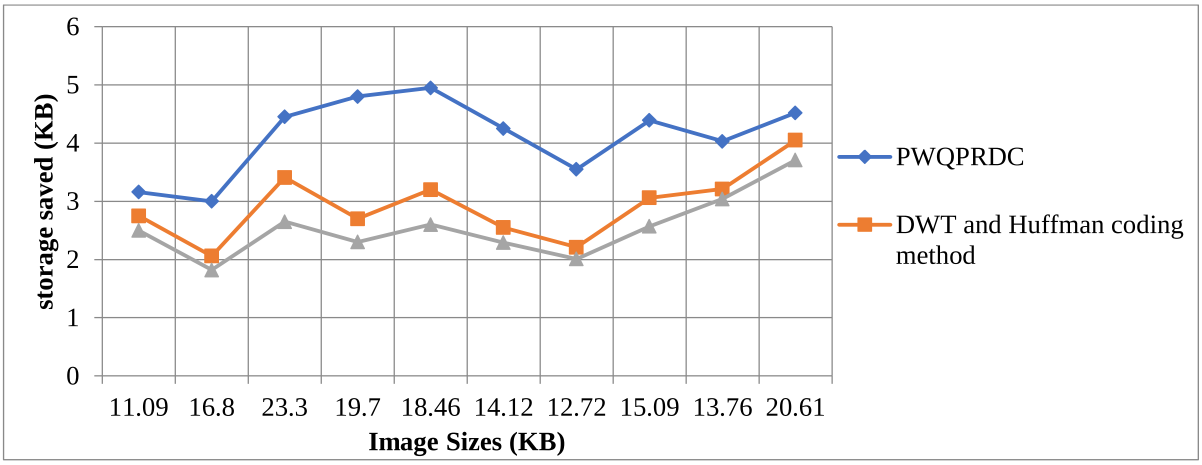


Figure 8 Storage saved chart versus image size

Figure 8 provides a comparison of storage savings achieved through three different methods: the PWQPRDC technique, the existing DWT and Huffman coding method [1], and the SFC scheme [2]. In this figure, the horizontal axis represents the sizes of natural images, while the vertical axis shows the storage savings performance measured in kilobytes (KB). As the image size varies, the storage savings results correspondingly increase or decrease. Among these three methods, the PWQPRDC technique demonstrates notably superior storage savings in comparison to the existing methods [1] and [2]. This enhancement is achieved through the implementation of the proposed Poisson wavelet quantized piecewise regressive distributed coding-based image compression and decompression technique. The overall performance data reveals that the PWQPRDC technique achieves a 43% improvement in storage savings compared to [1] and a 64% improvement compared to [2].

4. Conclusions

In this paper, a novel technique called PWQPRDC is developed to enhance the transmission and storage efficiency of natural images in a network. The process begins with efficient noise removal through filtering to enhance image quality before compression. Subsequently, the Poisson wavelet transform is applied to decompose the images, with a focus on quantizing the low-frequency components. Compressed images are generated using the piecewise regression-based coding technique. At the receiver side, the inverse compression process is carried out to recover the original image. Comprehensive experimental evaluations were conducted using a natural images dataset. The quantitative analysis results demonstrate that the PWQPRDC technique outperforms traditional compression methods, achieving higher peak signal-to-noise ratio, compression ratio, storage savings, and reduced time consumption.

References

[1] Shiju Thomas, Addapalli Krishna, Sabeen Govind, Aditya Kumar Sahu, "A Novel Image Compression method using Wavelet Coefficients and Huffman Coding", Journal of Engineering Research, Elsevier, 2023, Pages 1-20. https://doi.org/10.1016/j.jer.2023.08.015

[2] Ankita Vaish , Saumya Patel, "A sparse representation based compression of fused images using WDR coding", Journal of King Saud University - Computer and Information Sciences, Elsevier, Volume 34, Issue 8, 2022, Pages 6165-6178. https://doi.org/10.1016/j.jksuci.2022.02.002

[3] Tassnim Dardouri, Mounir Kaaniche, Amel Benazza-Benyahia, and Jean-Christophe Pesquet, "Dynamic Neural Network for Lossy-to-Lossless Image Coding", IEEE Transactions on Image Processing, Volume 31, 2022, Pages 569 – 584. **DOI:** 10.1109/TIP.2021.3132825

[4] Taizo Suzuki and Liping Huang, "Edge-Aware Extended Star-Tetrix Transforms for CFA-Sampled Raw Camera Image Compression", IEEE Transactions on Image Processing, Volume 31, September 2022, Pages 6072 – 6082. **DOI:** 10.1109/TIP.2022.3205470

[5] Juan Wang, Yiping Duan, Xiaoming Tao, Mai Xu, Jianhua Lu, "Semantic Perceptual Image Compression With a Laplacian Pyramid of Convolutional Networks", IEEE Transactions on Image Processing , Volume 30, 2021, Pages 4225 – 4237. **DOI:** 10.1109/TIP.2021.3065244

[6] Enas Abu Jrai, Shoroq Alsharari, Laiali Almazaydeh, Khaled Elleithy, And Osama Abu-Hamdan, "Improving LZW Compression of Unicode Arabic Text Using Multi-Level Encoding and a Variable-Length Phrase Code", IEEE Access , Volume 11, 2023, Pages 51915 – 51929. **DOI:** 10.1109/ACCESS.2023.3279021

[7] Tong Chen, Haojie Liu, Zhan Ma, Qiu Shen, Xun Cao, Yao Wang, "End-to-End Learnt Image Compression via Non-Local Attention Optimization and Improved Context Modeling", IEEE Transactions on Image Processing , Volume 30, 2021, Pages 3179 – 3191. **DOI:** 10.1109/TIP.2021.3058615

[8] Guo-Dong Su, Chin-Chen Chang, Chia-Chen Lin, "A High Capacity Reversible Data Hiding in Encrypted AMBTC-Compressed Images", IEEE Access, Volume 8, 2020, Pages 26984 – 27000. **DOI:** 10.1109/ACCESS.2020.2966234

[9] Haisheng Fu, Feng Liang, "Learned image compression with generalized octave convolution and cross-resolution parameter estimation", Signal Processing, Elsevier, Volume 202, 2023, Pages 1-13.https://doi.org/10.1016/j.sigpro.2022.108778

[10] Zhongqiang Li, Alexandra Ramos, Zheng Li, Michelle L. Osborn, Xin Li, Yanping Li, Shaomian Yao, Jian Xu, "An optimized JPEG-XT-based algorithm for the lossy and lossless compression of 16-bit depth medical image", Biomedical Signal Processing and Control, Elsevier, Volume 64, 2021, Pages 1-14.https://doi.org/10.1016/j.bspc.2020.102306

[11] André Nortje, Willie Brink, Herman A. Engelbrecht, Herman Kamper, "BINet: A binary inpainting network for deep patch-based image compression", Signal Processing: Image Communication, Elsevier, Volume 92, 2021, Pages 1-11.https://doi.org/10.1016/j.image.2020.116119

[12] Haisheng Fu, Feng Liang, Bo Lei, Qian Zhang, Jie Liang, Chengjie Tu, Guohe Zhang , "An extended context-based entropy hybrid modeling for image compression", Signal Processing: Image Communication, Elsevier, Volume 95, 2021, Pages 1-10.https://doi.org/10.1016/j.image.2021.116244

[13] Yaman Dua, Ravi Shankar Singh, Kshitij Parwani, Smit Lunagariya, Vinod Kumar, "Convolution Neural Network based lossy compression of hyperspectral images", Signal Processing: Image Communication, Elsevier, Volume 95, 2021, Pages 1-10.https://doi.org/10.1016/j.image.2021.116255

[14] Saad Merrouche, Boban Bondzˇulic´, Milenko Andric´, Dimitrije Bujakovic, "Accuracy analysis of lossless and lossy disparity map compression", Ain Shams Engineering Journal, Elsevier, Volume 13, Issue 3, 2022, Pages 1-9. https://doi.org/10.1016/j.asej.2021.09.007

[15] Mu Li, Kede Ma, Jane You, David Zhang, Wangmeng Zuo, "Efficient and Effective Context-Based Convolutional Entropy Modeling for Image Compression", IEEE Transactions on Image Processing , Volume 29, 2020, Pages 5900 – 5911. **DOI:** 10.1109/TIP.2020.2985225

[16] Mu Li, Wangmeng Zuo, Shuhang Gu, Jane You, David Zhang, "Learning Content-Weighted Deep Image Compression", IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 43, Issue 10, 2021, Pages 3446 – 3461. **DOI:** 10.1109/TPAMI.2020.2983926

[17] Yuvraj V. Parkale & Sanjay L. Nalbalwar, "Application of compressed sensing for image compression based on optimized Toeplitz sensing matrices", EURASIP Journal on Advances in Signal Processing, Springer, Volume 2021, 2021, Pages 1-30. https://doi.org/10.1186/s13634-021-00743-5

[18] Ranjeet Kumar, Utpreksh Patbhaje, A. Kumar, "An efficient technique for image compression and quality retrieval using matrix completion", Journal of King Saud University - Computer and Information Sciences, Elsevier, Volume 34, Issue 4, April 2022, Pages 1231-1239. https://doi.org/10.1016/j.jksuci.2019.08.002

[19] Jungan Chen , Jean Jiang , Xinnian Guo , and Lizhe Tan , "Bit-Error Aware Lossless Image Compression with 2D-LayerBlock Coding", Journal of Sensors , Hindawi, Volume 2021, November 2021, Pages 1-18. https://doi.org/10.1155/2021/7331459

[20] Mengtian Wu, Zaixing He, Xinyue Zhao, Shuyou Zhang, "General generative model-based image compression method using an optimisation encoder", IET Image Processing, Wiley, Volume 14, Issue 9, 2020, Pages 1750-1758. **https://doi.org/10.1049/iet-ipr.2019.0715**