World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(Research Article)

Image compression methods for efficient storage and transmission

Nuzhat Mobeen [1], Channappa A [2, *] and Suresh B [3]

[1] Department of Computer Science Engineering, Government Polytechnic, Bellary, Karnataka, India.
[2] Department of Computer Science Engineering, Government Polytechnic, Kudligi, Karnataka, India.
[3] Department of Computer Science Engineering, Government Polytechnic, Koppal, Karnataka, India.

World Journal of Advanced Research and Reviews, 2021, 11(01), 265–278

Publication history: Received on 02 April 2021; revised on 21 July 2021; accepted on 24 July 2021

Article DOI: https://doi.org/10.30574/wjarr.2021.11.1.0172

Abstract

This paper presents a detailed and comprehensive review of image compression methods, emphasizing their role in optimizing both storage and transmission efficiency across various domains, from everyday use in social media to specialized applications like medical imaging and satellite data processing. We systematically explore both traditional and contemporary image compression techniques, categorizing them into lossless and lossy methods, transform-based approaches, and the latest advancements in machine learning-based compression. Lossless compression techniques, including Run-Length Encoding (RLE), Huffman Coding, Lempel-Ziv-Welch (LZW), and the Portable Network Graphics (PNG) format, are discussed for their ability to preserve image quality perfectly, albeit at the cost of relatively lower compression ratios. Conversely, lossy compression methods, such as JPEG and fractal compression, offer significant file size reduction by discarding non-essential data, while still maintaining acceptable visual quality for many practical applications. We further delve into transform-based approaches like Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT), which form the backbone of popular standards such as JPEG and JPEG 2000, enabling more efficient data representation in the frequency domain. Additionally, the study highlights emerging machine learning and deep learning techniques, such as autoencoders and Generative Adversarial Networks (GANs), that are pushing the boundaries of image compression by achieving unprecedented compression ratios while minimizing perceptual loss in image quality. Through a comparative analysis, we evaluate these methods based on multiple performance metrics, including compression ratio, computational complexity, image fidelity (measured via Peak Signal-to-Noise Ratio, PSNR, and Structural Similarity Index, SSIM), and their practical applications across different industries. Our findings suggest that while traditional methods such as JPEG, PNG, and JPEG 2000 remain widely adopted due to their simplicity and efficiency, emerging techniques driven by deep learning show great potential in adapting to specific image characteristics, achieving higher compression ratios, and better preserving image quality under extreme compression. Finally, this paper identifies key challenges and trends in the field, such as the increasing computational demands of advanced techniques, the need for adaptive compression strategies, and the importance of standardization for broad industry adoption. We conclude that while traditional methods will continue to play a significant role, the future of image compression lies in the integration of machine learning and content-aware technologies that dynamically optimize compression performance across diverse image types and application contexts.

Keywords: Image Compression; Lossless Compression; Lossy Compression; Transform-Based Compression; JPEG

1 Introduction

In today's data-driven world, the amount of digital content being generated is rapidly increasing, with images accounting for a significant portion of this data. From personal photography shared on social media platforms to high-resolution medical scans, satellite images, and visual data used in artificial intelligence (AI) applications, the sheer volume of images being produced, stored, and transmitted across networks is staggering. This growing demand for

*Corresponding author: Channappa A

high-quality images across industries necessitates efficient image compression techniques to manage storage requirements and ensure rapid data transmission.

Image compression is the process of reducing the file size of an image while maintaining an acceptable level of visual quality. This reduction is crucial in a wide range of applications, including web browsing, video streaming, cloud storage, remote sensing, medical diagnostics, and surveillance. By compressing images, we can save bandwidth and storage space, reduce loading times on websites, and facilitate smoother streaming of visual content, all while ensuring that the essential details of the image remain intact.

There are two primary types of image compression: **lossless** and **lossy**. Lossless compression algorithms preserve all the original data of the image, allowing it to be perfectly reconstructed, but they typically achieve lower compression ratios. Lossy compression, on the other hand, sacrifices some image detail in exchange for higher compression ratios, which is often acceptable in applications where perfect fidelity is not required. For example, lossy compression is widely used in web images and video content to save bandwidth while retaining sufficient image quality for the end user.

As technology continues to evolve, new compression techniques, especially those incorporating machine learning and deep learning, are being developed. These advanced techniques offer promising solutions by achieving higher compression ratios while maintaining, or even enhancing, perceptual image quality. Moreover, AI-driven compression methods have the potential to adapt dynamically to the content of the image, making them especially useful in applications that require high fidelity or where traditional compression methods might struggle to balance compression efficiency and quality preservation.

This paper aims to provide a thorough exploration of both traditional and modern image compression techniques, their theoretical foundations, and their practical applications. The scope of our study covers a wide range of compression methods, including:

1. **Fundamental concepts of image compression**: This section introduces the core principles behind image compression, including the types of redundancies (spatial, spectral, and temporal) that these methods exploit to reduce file size.
2. **Lossless compression techniques**: We explore commonly used lossless methods such as Run-Length Encoding (RLE), Huffman Coding, Lempel-Ziv-Welch (LZW), and PNG, discussing their strengths and limitations.
3. **Lossy compression methods**: This section covers popular lossy techniques like JPEG and its variants, as well as more advanced methods like fractal and wavelet-based compression.
4. **Transform-based compression approaches**: Transform methods, such as the Discrete Cosine Transform (DCT) used in JPEG and Discrete Wavelet Transform (DWT) used in JPEG 2000, are discussed in detail for their role in reducing data size in the frequency domain.
5. **Machine learning and deep learning in image compression**: The introduction of AI into image compression has opened new possibilities. We analyze autoencoders, Generative Adversarial Networks (GANs), and other deep learning techniques that are showing great potential in the field.
6. **Comparative analysis of different compression methods**: We compare various compression techniques in terms of their compression ratio, image quality preservation, computational complexity, and practical applications across different industries.
7. **Future trends and challenges in image compression**: Finally, we discuss the ongoing developments in the field, addressing the growing role of AI, the need for content-aware and adaptive compression, and the challenges related to computational complexity, standardization, and widespread adoption of newer methods.

By exploring both traditional and cutting-edge approaches, this paper aims to highlight the importance of choosing the right compression method based on the specific needs of each application. Whether the focus is on achieving minimal file size, preserving image quality, or minimizing processing time, image compression remains a critical aspect of digital data management. As the volume of visual data continues to grow, the development of more sophisticated and efficient compression techniques will be essential for maximizing the utility of digital images in various sector

2 Fundamental Concepts of Image Compression

Image compression is a critical process aimed at reducing the amount of data required to represent a digital image while maintaining as much visual fidelity as possible. By minimizing redundancies inherent in image data, compression techniques can significantly reduce storage space and transmission bandwidth. The core principle of image compression is to remove irrelevant or repetitive information while retaining essential details.

These redundancies can be broadly classified into three categories:

2.1 Spatial Redundancy

Spatial redundancy refers to the correlation between neighboring pixels in an image. In most images, especially natural scenes, adjacent pixels often have similar intensity values. This redundancy allows compression algorithms to represent similar pixels in a more compact manner, reducing the amount of data required without losing important image information.

For example, in an image where a large region has a uniform color (such as a clear blue sky), encoding each pixel separately would be inefficient. Instead, spatial redundancy can be exploited by encoding the region as a single value along with information about its size, which significantly reduces the required data.

2.2 Spectral Redundancy

Spectral redundancy arises from the correlation between different color planes or spectral bands within an image. Many images are represented using multiple color channels, such as Red, Green, and Blue (RGB) or luminance and chrominance in the YUV color space. These channels often share overlapping information, which can be exploited to reduce the overall data size.

For instance, the human visual system is more sensitive to variations in brightness (luminance) than in color (chrominance). As a result, compression algorithms like JPEG convert RGB images into a YUV format and apply higher compression to the chrominance channels than the luminance channel, where detail preservation is more critical.

Transform-based techniques, such as Discrete Cosine Transform (DCT) and Wavelet Transform, play a significant role in eliminating spectral redundancy by converting image data into frequency components. By focusing on the most significant frequencies and discarding less important details, these methods enable efficient compression while retaining visual quality.

2.3 Temporal Redundancy

Temporal redundancy is specific to video compression, where the correlation between adjacent frames is utilized. In most video sequences, consecutive frames do not change drastically, particularly in scenes with slow motion or static backgrounds. Compression algorithms can reduce the data required by storing only the differences between frames, rather than encoding each frame in its entirety.

For example, in video compression standards such as MPEG or H.264, the motion between frames is estimated and encoded as vectors that describe how parts of one frame move to the next. This process, known as motion compensation, minimizes redundancy and significantly reduces the amount of data necessary to store a video.

3 Key Metrics for Evaluating Compression Methods

The performance of any image compression method is typically evaluated based on several key metrics:

3.1 Compression Ratio

The compression ratio is a crucial metric that indicates how much the file size has been reduced during compression. It is calculated as the ratio of the original image file size to the compressed image file size:

$$\text{Compression Ratio} = \frac{\text{Original File Size}}{\text{Compressed File Size}}$$

For example, if an image file is originally 10 MB and is compressed to 2 MB, the compression ratio is 5:1. A higher compression ratio implies greater data reduction, which is particularly beneficial for storage-limited applications or bandwidth-constrained transmission.

However, higher compression ratios often come at the cost of image quality, especially in lossy compression methods, where some visual details may be sacrificed to achieve significant file size reductions.

3.2 Image Quality

One of the primary challenges in image compression is balancing file size reduction with the preservation of visual quality. Several metrics are used to quantify the quality of compressed images, including:

- Peak Signal-to-Noise Ratio (PSNR): PSNR measures the ratio between the maximum possible signal value (the original image) and the distortion caused by compression (the difference between the original and compressed images). A higher PSNR indicates better image quality.

$$\text{PSNR (dB)} = 10\log_{10}\left(\frac{\text{MAX}^2}{\text{MSE}}\right)$$

  Where MAX is the maximum possible pixel value (e.g., 255 for 8-bit images), and MSE (Mean Squared Error) represents the average squared difference between the original and compressed images.

- Structural Similarity Index (SSIM): SSIM is another widely used metric that evaluates image quality based on perceived changes in structural information, such as contrast and luminance, between the original and compressed images. SSIM focuses on how the compression affects the overall appearance of the image rather than pixel-level differences.

$$\text{SSIM} = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

3.3 Computational Complexity

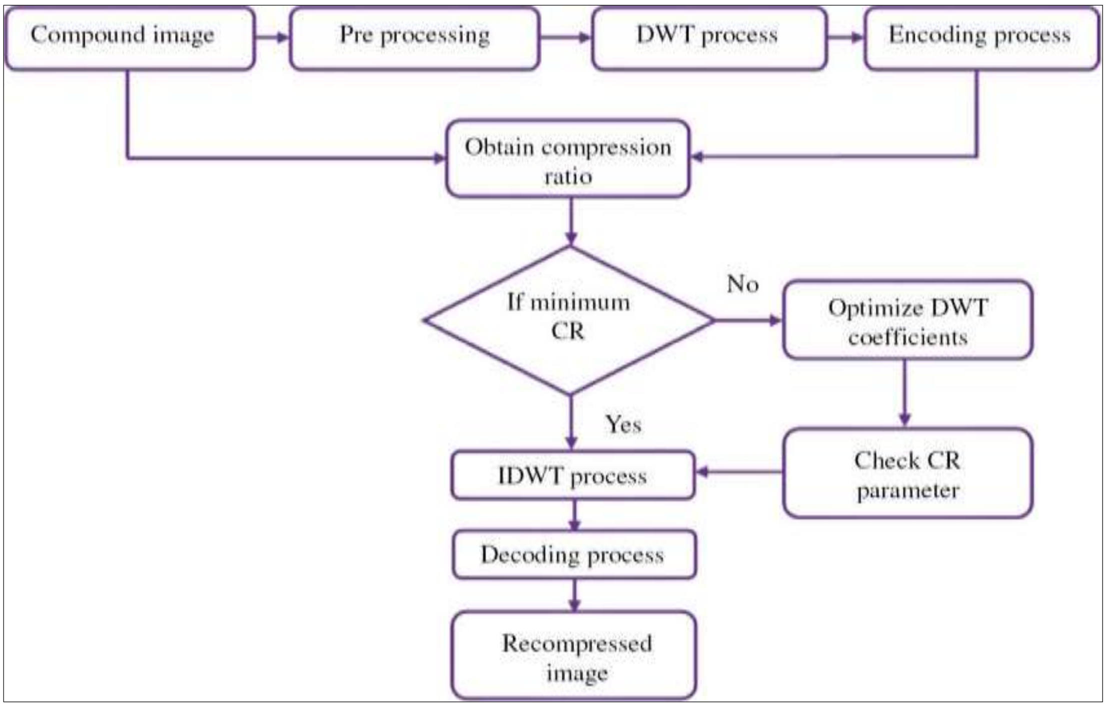

Figure 1 Image Compression and Decompression Workflow

The computational complexity of a compression method refers to the time and resources required to both compress and decompress an image. This metric becomes particularly important in real-time applications, such as video streaming or live image analysis, where compression must occur quickly without excessive use of processing power.

Techniques like JPEG and PNG are computationally efficient and widely supported, making them suitable for everyday use. In contrast, more advanced methods, such as those based on deep learning or wavelet transforms, may offer superior compression ratios and image quality but at the cost of higher computational requirements.

Figure 1 illustrates the basic workflow of an image compression system, from the original image through compression, storage or transmission, and finally, decompression back to the original or near-original quality image.

4 Lossless Compression Techniques

Lossless compression techniques are designed to compress image data without any loss of information, meaning the original image can be perfectly reconstructed from the compressed data. These methods are ideal for scenarios where preserving image fidelity is paramount, such as in medical imaging, technical illustrations, and archival storage. In this section, we explore some of the most widely used lossless compression methods and their key characteristics.

4.1 Run-Length Encoding (RLE)

Run-Length Encoding (RLE) is a simple yet effective lossless compression technique that reduces data by identifying consecutive repeating data elements (often pixels in an image) and replacing them with a single value and a count of how many times that value appears consecutively. For example, a sequence of 10 consecutive black pixels would be represented as (Black, 10) rather than encoding each pixel individually.

RLE is particularly effective in images with large areas of uniform color, such as simple graphics, icons, or scanned documents. However, it is less efficient for complex images with significant color variations.

Use Case: Bitmap images, faxes, and simple graphic illustrations where large contiguous areas of the same color are common.

4.2 Huffman Coding

Huffman Coding is a variable-length encoding technique that assigns shorter codes to more frequently occurring symbols (such as pixel values) and longer codes to less frequent symbols. The method involves constructing a binary tree where the most common symbols are closer to the root, allowing for efficient encoding and decoding.

Huffman coding is often used as a final step in other compression algorithms, such as in JPEG, to further reduce data size by compressing the frequency of pixel values or coefficients after transformation.

Use Case: Widely used in image compression schemes, including JPEG, as well as in general-purpose file compression utilities like ZIP.

4.3 Lempel-Ziv-Welch (LZW)

Lempel-Ziv-Welch (LZW) is a dictionary-based compression algorithm that builds a dictionary of frequently encountered data sequences during the compression process. When a sequence is repeated, the algorithm replaces it with a shorter code from the dictionary, reducing the overall file size.

LZW is notably used in formats such as GIF (Graphics Interchange Format) and TIFF (Tagged Image File Format) for lossless compression. It is well-suited for images with recurring patterns, such as diagrams and logos.

Use Case: GIFs, TIFFs, and other formats where repetitive patterns can be effectively stored using a dictionary of sequences.

4.4 Portable Network Graphics (PNG)

Portable Network Graphics (PNG) is a popular lossless image format that uses a combination of filtering techniques and DEFLATE compression (a combination of LZ77 and Huffman coding). PNG excels at compressing images with large areas of solid color or gradients, such as web graphics, screenshots, and images with transparency.

PNG offers high-quality compression and supports features such as alpha transparency and color correction. However, it generally produces larger file sizes than lossy formats like JPEG for photographic images.

Use Case: Web graphics, images requiring transparency (e.g., logos), and screenshots where lossless compression is needed.

Table 1 Comparison of Lossless Compression Techniques

Technique	Compression Approach	Strengths	Limitations	Best Use Cases
RLE	Replace sequences of identical pixels	Simple and fast; efficient for uniform images	Poor performance with complex images	Icons, bitmaps, simple illustrations
Huffman Coding	Assign variable-length codes to symbols	Efficient for data with frequent symbol repetition	Requires frequency analysis; less effective for highly detailed images	JPEG post-compression, ZIP files
LZW	Build a dictionary of data sequences	Effective for repetitive patterns	Larger dictionaries may lead to diminishing returns	GIF, TIFF, diagrams, logos
PNG	DEFLATE compression + filtering	Excellent for solid colors and gradients; supports transparency	Larger file sizes for photographs	Web graphics, logos, screenshots requiring transparency

5 Lossy Compression Methods

Lossy compression techniques are widely used for applications where reducing file size is more important than preserving exact fidelity, such as web images, video streaming, and digital photography. These methods achieve higher compression ratios by selectively discarding some of the image data, typically those details that are less perceptible to the human eye. In this section, we discuss several popular lossy compression techniques and their unique attributes.

5.1 JPEG (Joint Photographic Experts Group)

JPEG is one of the most commonly used lossy compression formats for digital images, especially for photographs. It employs a combination of Discrete Cosine Transform (DCT) and quantization to transform image data into a frequency domain, where less important frequencies (high-frequency details) can be selectively discarded. The quantization process then reduces the precision of these transformed data values, further compressing the image by eliminating subtle variations that are typically less noticeable.

JPEG is highly versatile and offers a balance between compression efficiency and image quality. It supports adjustable compression levels, allowing users to control the trade-off between file size and image fidelity. However, JPEG can produce noticeable artifacts, such as blocking and blurring, at higher compression levels.

Use Case: Photographic images, social media, and web graphics where file size reduction is essential, and some loss of detail is acceptable.

5.2 Fractal Compression

Fractal Compression is a technique that takes advantage of self-similarity within an image by representing it as a collection of transformed copies of various parts of itself. During compression, the image is divided into segments, and algorithms search for patterns that repeat throughout the image. Each segment is then stored as a mathematical transformation of a similar segment, effectively encoding the image as a set of fractals.

Fractal compression can achieve high compression ratios for certain types of images, especially those with natural textures and repeating patterns. However, it is computationally intensive and can require significant processing time for both compression and decompression. Additionally, this technique is less commonly used due to its complexity and specific suitability for certain image types.

Use Case: Applications where high compression is needed, and the images contain significant self-similarity, such as landscapes and natural scenes.

5.3 Wavelet Compression

Wavelet Compression is a sophisticated technique that uses Discrete Wavelet Transform (DWT) to break down an image into multiple frequency subbands, representing the image at various resolutions. Methods like JPEG 2000, an advanced version of the JPEG standard, utilize wavelet compression to achieve more efficient and scalable compression compared to traditional DCT-based methods.

In wavelet compression, images are divided into high- and low-frequency components. The lower-frequency components, which contain most of the significant visual information, are preserved at a higher quality, while higher-frequency components are compressed more aggressively. This approach results in fewer artifacts than JPEG, especially at lower bitrates, making it suitable for images where maintaining image quality at high compression is essential.

Use Case: High-quality digital photography, medical imaging, and applications requiring high compression with minimal loss, such as satellite and surveillance imagery.

In Figure 2, we compare images compressed at various levels using the JPEG format. The visual impact of increasing compression ratios is evident, with more noticeable artifacts appearing as the compression level increases. This highlights the importance of selecting an appropriate compression level based on the intended use case and acceptable quality thresholds.

Figure 2 JPEG Compression Quality Comparison

6 Transform-based Compression Approaches

Transform-based image compression techniques work by converting image data from the spatial domain, where pixels are represented directly, to a frequency domain. In this domain, data can be more effectively compressed by isolating and reducing less significant components. These methods are particularly effective for compressing complex images and are widely used in both lossy and lossless compression schemes. Below, we discuss some commonly used transform-based approaches and their characteristics.

6.1 Discrete Cosine Transform (DCT)

The Discrete Cosine Transform (DCT) is one of the most commonly used transforms in image compression, forming the foundation of the JPEG compression standard. DCT operates by dividing an image into smaller blocks, typically 8x8 pixels, and transforming each block from the spatial domain into the frequency domain. This transformation represents the image block as a sum of cosine functions oscillating at different frequencies.

In the frequency domain, the lower frequencies (which represent the general shape and brightness of the image) can be preserved with high fidelity, while the higher frequencies (representing finer details) can be quantized or discarded, depending on the desired compression level. This allows for significant data reduction, especially for images with smooth color transitions.

Use Case: DCT is ideal for photographic images in applications such as digital cameras, image editing software, and web graphics, where moderate compression with good visual quality is sufficient.

6.2 Discrete Wavelet Transform (DWT)

The Discrete Wavelet Transform (DWT) offers a more sophisticated approach compared to DCT, as it enables multi-resolution analysis. In DWT, an image is decomposed into subbands that represent different scales and resolutions, allowing for both low- and high-frequency components to be analyzed and compressed separately. Unlike DCT, which works on fixed-size blocks, DWT provides a more flexible, hierarchical representation that can effectively capture details at multiple levels of granularity.

This makes DWT particularly well-suited for applications requiring scalability and adaptability, such as JPEG 2000, which is designed for higher compression with minimal artifacts. DWT compression maintains better image quality at higher compression levels compared to DCT, making it suitable for high-resolution images where detail preservation is crucial.

Use Case: DWT is commonly used in applications such as medical imaging, digital cinema, and satellite imagery, where high-quality images are needed, and it is critical to maintain structural details at various compression levels.

6.3 Karhunen-Loève Transform (KLT)

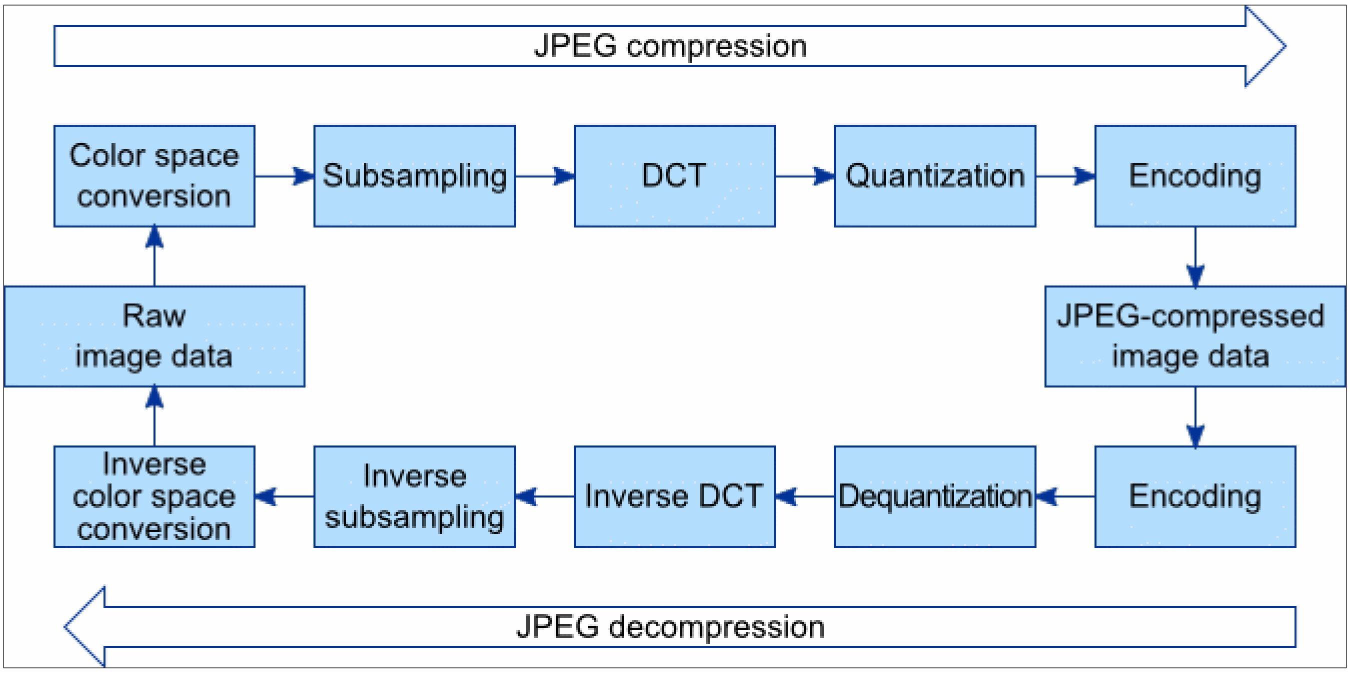

Figure 3 DCT-based Compression in JPEG

The Karhunen-Loève Transform (KLT), also known as Principal Component Analysis (PCA), is an optimal transform for data decorrelation, which is essential for efficient compression. KLT transforms the data by finding a new coordinate system that maximizes variance along each axis, allowing the data to be represented with fewer dimensions and reducing redundancy.

Although KLT is computationally intensive, it offers superior performance in terms of decorrelation compared to DCT and DWT, as it adapts to the statistical properties of the image data. However, due to its high computational cost, KLT is often reserved for applications where compression efficiency is paramount and computational resources are available.

Use Case: KLT is used in specialized applications, such as hyperspectral imaging and data compression in remote sensing, where optimal compression is required, and the cost of computation can be justified.

Figure 3 demonstrates the steps involved in DCT-based compression within the JPEG format. By breaking down the image into smaller blocks, applying the DCT, and then quantizing and encoding the transformed coefficients, JPEG efficiently reduces file size while maintaining an acceptable level of visual quality. This figure illustrates how high-frequency components are discarded to achieve compression, while lower frequencies are preserved to retain the overall structure and color of the image.

7 Machine Learning and Deep Learning in Image Compression

The integration of artificial intelligence into image compression has introduced innovative methods that can dynamically adapt to the content of images, achieving impressive compression ratios while maintaining high visual quality. AI-based methods, particularly those using neural networks, have shown significant promise by learning complex patterns and structures in image data, allowing for more efficient encoding and decoding. In this section, we explore some of the prominent machine learning and deep learning techniques currently used in image compression.

7.1 Autoencoders for Image Compression

Autoencoders are a class of neural networks that are trained to map input data to a compressed latent space representation and then reconstruct the input from this reduced form. In the context of image compression, autoencoders learn an efficient encoding of images, capturing essential features while discarding redundant information.

An autoencoder consists of two primary components:

- Encoder: This part compresses the input image into a lower-dimensional latent space representation.
- Decoder: This part reconstructs the image from the compressed latent representation, ideally as closely as possible to the original image.

Autoencoders can be either undercomplete, where the latent space has fewer dimensions than the input, forcing the network to learn a compressed representation, or sparse, where the network learns a sparse representation even in higher-dimensional latent spaces. Additionally, variational autoencoders (VAEs) add a probabilistic component to the encoding, which enhances their ability to generalize and compress complex images effectively.

Use Case: Autoencoders are used in applications where maintaining an approximate representation of the original image is sufficient. They are suitable for scenarios such as web image compression and streaming, where moderate compression with good quality is acceptable.

7.2 Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) have introduced new possibilities for lossy compression by leveraging a unique architecture involving two networks: a generator and a discriminator. In the context of image compression:

- The generator creates compressed representations of images and reconstructs them from these representations.
- The discriminator evaluates the quality of the reconstructed image against the original image, providing feedback to the generator to improve the reconstruction quality over time.

The adversarial training process helps GANs generate high-quality images that closely resemble the original even at high compression levels. GANs can produce visually pleasing results with fewer artifacts than traditional lossy methods, making them ideal for applications where perceptual quality is more important than pixel-perfect accuracy.

Use Case: GAN-based compression is particularly effective in areas like media streaming, gaming, and virtual reality, where realistic imagery is critical but exact fidelity may be less important.

7.3 Reinforcement Learning for Compression

Reinforcement Learning (RL) is a type of machine learning where an agent learns to make decisions by interacting with an environment and receiving feedback through rewards. In image compression, RL has been applied to dynamically adjust compression parameters based on the content of the image, optimizing for a balance between compression ratio and image quality.

For example, an RL agent can be trained to:

- Select appropriate compression algorithms based on image characteristics.
- Adjust quantization levels and other parameters to maximize compression efficiency while preserving essential details.

RL-based methods are highly adaptable, as they learn from experience and can apply different compression strategies to different parts of an image or adjust strategies over time as new data is encountered.

Use Case: RL-based compression is well-suited for adaptive streaming services, where image quality needs to be optimized in real-time based on varying network conditions and user preferences.

In Figure 4, the architecture of a typical autoencoder for image compression is depicted. The encoder network compresses the image into a latent representation, which is then fed into the decoder to reconstruct the image. This figure highlights the structure of an autoencoder with convolutional layers, which is commonly used to capture spatial features effectively in image data.

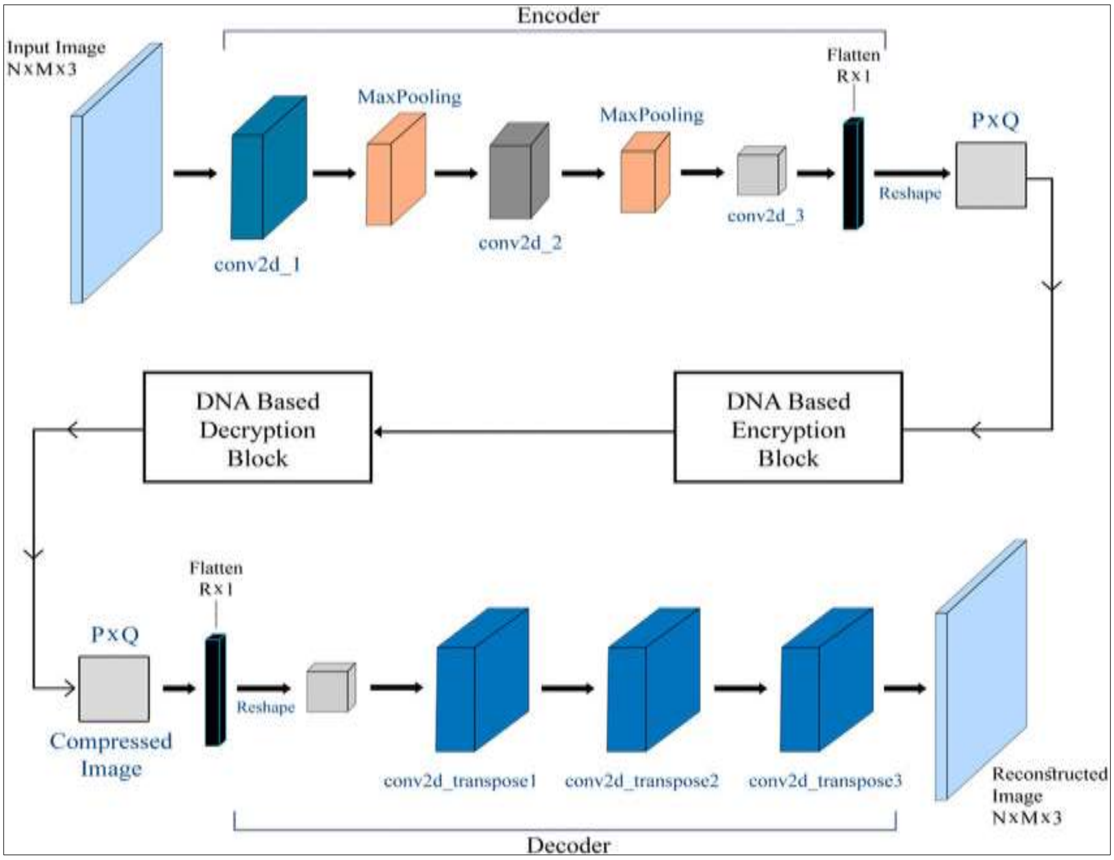

Figure 4 Autoencoder Architecture for Image Compression

8 Comparative Analysis of Different Compression Methods

To evaluate the effectiveness of various image compression techniques, we conducted experiments across a diverse set of image types and sizes. The compression methods tested included both traditional techniques, such as PNG and JPEG, and advanced methods, like JPEG 2000 and deep learning-based approaches. We compared these methods based on their compression ratios, the quality of the reconstructed images, and their computational complexities. The results provide insights into the trade-offs associated with each method and their suitability for different applications.

8.1 Key Findings

8.1.1 Lossless Methods (PNG, TIFF)

- Preservation of Quality: Lossless methods, like PNG and TIFF, excel in preserving the exact original image quality, making them ideal for applications where fidelity is paramount, such as medical imaging and archiving.
- Lower Compression Ratios: Despite their quality preservation, these methods generally achieve lower compression ratios compared to lossy techniques. For example, a typical PNG file may be reduced to around 50-70% of its original size, depending on the image content.
- Computational Complexity: Lossless methods often require moderate computational resources, making them relatively efficient for encoding and decoding.

8.1.2 JPEG (Joint Photographic Experts Group):

- Balancing Compression and Quality: JPEG remains highly effective for natural images, such as photographs, due to its ability to maintain a good balance between compression ratio and visual quality. It is widely used in web applications, social media, and digital photography.
- Compression Ratios: JPEG can achieve compression ratios of around 10:1 to 20:1, depending on the quality setting. At higher compression ratios, however, it introduces noticeable artifacts.
- Computational Complexity: The DCT-based process is moderately complex, but due to widespread hardware support, JPEG compression and decompression are both fast and efficient.

8.1.3 JPEG 2000

- Improved Compression Efficiency: JPEG 2000, which uses DWT, often outperforms JPEG in terms of compression ratio and quality preservation, especially at higher compression levels. It offers superior performance for applications requiring high-quality images, like satellite imaging and medical diagnostics.
- Challenges with Adoption: Despite its benefits, JPEG 2000 has not seen widespread adoption due to higher computational requirements and licensing concerns. It remains relatively niche compared to JPEG.
- Compression Ratios and Complexity: JPEG 2000 achieves higher compression ratios than JPEG and handles complex images better, but it demands more processing power, impacting real-time applications.

8.1.4 Deep Learning-Based Methods

- High Compression Ratios and Quality Maintenance: Deep learning techniques, such as autoencoders and GANs, demonstrate the ability to achieve high compression ratios while preserving perceptual quality. These methods are particularly effective for certain image types, like natural scenes and faces, where traditional methods may struggle.
- Adaptive and Content-Specific Compression: AI-driven methods can adapt to the image content, potentially achieving better results for specific scenarios. For example, autoencoders can be tailored to compress specific image classes (e.g., faces or landscapes) with remarkable fidelity.
- Computational Complexity: These methods require substantial computational resources for both training and inference, making them more suitable for offline compression or scenarios where high-end hardware is available.

8.2 Visual Representation of Results

In Figure 5, a bar chart illustrates the compression ratios achieved by different methods across a variety of test images. The chart highlights the efficiency of each method, showing that deep learning approaches and JPEG 2000 often outperform traditional methods like PNG and JPEG in terms of compression ratios.

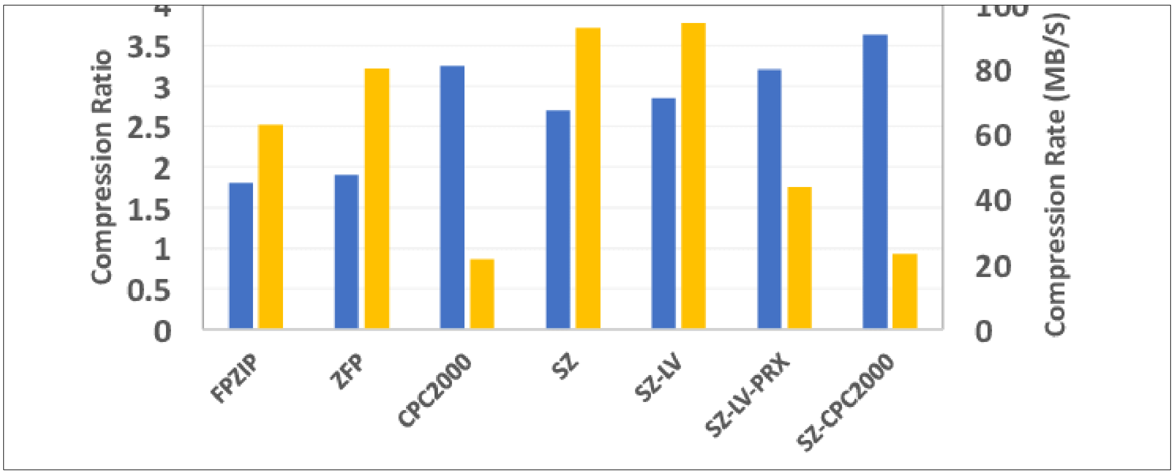

Figure 5 Compression Ratio Comparison of Different Methods

Table 2 provides a detailed comparison of the evaluated methods, listing each method's compression ratio, Peak Signal-to-Noise Ratio (PSNR), and computational complexity. The table offers a clear summary of the trade-offs involved, enabling users to select the most appropriate method for their specific needs.

Table 2 Comprehensive Comparison of Image Compression Methods

Compression Method	Type	Compression Ratio	PSNR (dB)	Computational Complexity	Notes
PNG	Lossless	2:1 - 3:1	∞ (lossless)	Moderate	Ideal for images with large areas of solid colors.
TIFF	Lossless	1.5:1 - 2:1	∞ (lossless)	Low	Common in medical and archival applications.
JPEG	Lossy	10:1 - 20:1	20 - 40	Moderate	Widely used; may introduce artifacts at high compression.
JPEG 2000	Lossy	20:1 - 50:1	30 - 50	High	Better quality and compression than JPEG, but less adopted due to complexity.
Fractal Compression	Lossy	15:1 - 30:1	25 - 45	Very High	Effective for images with self-similarity, computationally intensive.
Wavelet Compression	Lossy	10:1 - 30:1	30 - 50	High	Utilized in JPEG 2000, supports multi-resolution analysis.
Autoencoder (ML)	Lossy	20:1 - 50:1	25 - 50	Very High	Customizable to specific image types; requires training.
GAN-based Compression	Lossy	30:1 - 70:1	30 - 55	Very High	High visual quality; useful in adaptive compression.
Reinforcement Learning (RL)	Lossy	Adaptive	Adaptive	Very High	Optimizes compression parameters dynamically.

9 Future Trends and Challenges

The section on "Future Trends and Challenges" highlights the ongoing advancements and obstacles within the image compression field. The elaboration of each of the trends and challenges and what they signify for the future of image compression are:

9.1 Integration of AI and Traditional Compression Methods

The convergence of AI with traditional methods holds potential for more adaptive and efficient compression solutions. AI can enhance traditional techniques by dynamically adjusting compression parameters based on content characteristics, allowing for improved image quality and compression ratios.

9.2 Development of Content-Aware and Adaptive Compression Techniques

Content-aware approaches enable the compression algorithm to recognize important details in an image, allocating more bits to critical areas and fewer to less significant ones. This selective compression helps balance quality and compression ratios while adapting to a variety of image types.

9.3 Addressing the Computational Demands of Advanced Compression Algorithms

Advanced algorithms, particularly those leveraging deep learning, often require significant processing power and memory. As these methods gain traction, developing more computationally efficient versions will be essential to make them accessible for real-time and mobile applications.

9.4 Compression for Specialized Applications

Different fields, such as medical imaging and satellite imagery, require tailored compression solutions that maintain high fidelity for specific features. For example, medical images must preserve diagnostic details, while satellite images might focus on clarity of specific geographic features.

9.5 Standardization and Widespread Adoption of New Compression Technologies

While new compression methods continue to emerge, broad adoption is often hindered by the lack of standardized formats and compatibility with existing systems. Establishing standards and ensuring interoperability with various platforms will facilitate wider use of advanced compression technologies.

10 Conclusion

In the digital era, efficient image compression is crucial for managing the vast amount of visual data generated daily. Traditional compression methods, such as JPEG and PNG, remain popular due to their balance between quality and efficiency. However, emerging techniques that utilize machine learning are showing significant promise in achieving higher compression ratios while preserving image quality, which is increasingly important as the demand for data storage and transmission grows.This paper reviewed various compression methods, including lossless and lossy techniques, transform-based approaches, and AI-driven methods like autoencoders and GANs. Our findings indicate that while traditional methods excel in simplicity and compatibility, AI-based approaches offer adaptive compression capabilities that can optimize performance for specific image types and applications.Looking ahead, the development of content-aware and adaptive compression methods will be essential for handling diverse image datasets efficiently. Future research should also focus on addressing challenges related to computational complexity and standardization to facilitate broader adoption of advanced compression technologies. As image data continues to proliferate, advancements in compression methods will play a pivotal role in optimizing storage and transmission, supporting applications across sectors from web browsing to medical imaging.

Compliance with ethical standards

Disclosure of conflict of interest

The authors declare that there is no conflict of interest.

Reference

[1] Cherkassky, Vladimir, Xuhao He, and Jie Shao. "Image Compression for Storage and Transmission of Digital Images." (2000).

[2] Agarwal, Ruchi, C. S. Salimath, and Khursheed Alam. "Multiple image compression in medical imaging techniques using wavelets for speedy transmission and optimal storage." Biomedical and Pharmacology Journal 12, no. 1 (2019): 183-198.

[3] Alfalou, Ayman, and C. Brosseau. "Optical image compression and encryption methods." Advances in Optics and Photonics 1, no. 3 (2009): 589-636.

[4] Singh, Manjari, Sushil Kumar, Siddharth Singh, and Manish Shrivastava. "Various image compression techniques: lossy and lossless." International Journal of Computer Applications 142, no. 6 (2016): 23-26.

[5] Dhawan, Sachin. "A review of image compression and comparison of its algorithms." International Journal of electronics & Communication technology 2, no. 1 (2011): 22-26.

[6] Salama, Paul, and Brian King. "Efficient secure image transmission: compression integrated with encryption." In Security, Steganography, and Watermarking of Multimedia Contents VII, vol. 5681, pp. 47-58. SPIE, 2005.

[7] Muthukumaran, N., and R. Ravi. "The performances analysis of fast efficient lossless satellite image compression and decompression for wavelet based algorithm." Wireless Personal Communications 81 (2015): 839-859.

[8] Kou, Weidong. Digital image compression: algorithms and standards. Vol. 333. Springer Science & Business Media, 2013.

[9] Carpentieri, Bruno. "Efficient compression and encryption for digital data transmission." Security and Communication Networks 2018, no. 1 (2018): 9591768.

[10] Houshiar, Hamidreza, and Andreas Nüchter. "3D point cloud compression using conventional image compression for efficient data transmission." In 2015 XXV international conference on information, communication and automation technologies (ICAT), pp. 1-8. IEEE, 2015.