

A Comparative Analysis of Image Partitioning and Compression Mechanisms

Sampada P. Chaudhari¹ and Nileshsingh V. Thakur²

¹Dept. of Computer Science and Engineering, Prof Ram Meghe College of Engineering & Management, Badnera, India
Email: sampada.chaudhari7@gmail.com

²Dept. of Computer Science and Engineering, Yeshwantrao Chavan College of Engineering, Nagpur, India
Email: thakurnisvis@rediffmail.com

Abstract—This paper presents a comprehensive comparative analysis of image partitioning and compression mechanisms, two fundamental techniques in image processing and data compression. Image partitioning involves dividing an image into smaller components, while compression aims to reduce the size of digital images while preserving visual quality. The analysis encompasses various aspects of both techniques, including methodologies, algorithms, and performance metrics. For image partitioning, segmentation methods such as block-based, region-based, and hierarchical partitioning are examined, along with their respective advantages and limitations. Additionally, the paper explores compression mechanisms such as transform coding, quantization, entropy coding, and predictive coding, highlighting their roles in achieving efficient compression. Furthermore, the comparative analysis evaluates the trade-offs between different partitioning and compression techniques in terms of compression ratio, image quality, computational complexity, and applicability across different types of images and applications. Various compression standards such as JPEG and PNG are also reviewed to provide insights into real-world implementations. Finally, the paper discusses future research directions and emerging trends in image partitioning and compression, including the integration of machine learning techniques for improved segmentation and compression performance. The findings presented in this comparative analysis can serve as a valuable resource for researchers, practitioners, and developers working in the fields of image processing, data compression, and multimedia systems.

Index Terms— Image partitioning, Block-based partitioning, Region-based partitioning, Hierarchical partitioning and Image compression.

I. INTRODUCTION

In the realm of digital image processing and data compression, image partitioning and compression mechanisms stand as indispensable pillars, facilitating the efficient storage, transmission, and manipulation of visual information. Image partitioning involves the segmentation of images into smaller, more manageable components, while compression aims to reduce the redundancy and size of digital image data without compromising perceptual quality. Both techniques play pivotal roles in various applications spanning from multimedia communication to medical imaging and computer vision.

Image partitioning techniques encompass a spectrum of methodologies, each tailored to address specific challenges and requirements. From traditional block-based partitioning, which divides images into fixed-size

blocks, to more sophisticated region-based and hierarchical approaches, which exploit local and global image characteristics, the field offers a diverse array of solutions. Similarly, compression mechanisms employ a multitude of algorithms, including transform coding, quantization, entropy coding, and predictive coding, each contributing to the reduction of data size through different strategies.

While these techniques serve a common goal of optimizing image data representation, they exhibit distinct trade-offs in terms of compression ratio, computational complexity, and preservation of image quality. Understanding these trade-offs is crucial for selecting the most suitable approach for a given application scenario. Furthermore, the landscape of image processing and compression is constantly evolving, with emerging technologies such as machine learning increasingly influencing the design and performance of partitioning and compression algorithms.

In this paper, we embark on a comparative analysis of image partitioning and compression mechanisms, aiming to provide a comprehensive understanding of their strengths, weaknesses, and applicability across various domains. Through a systematic exploration of methodologies, algorithms, and performance metrics, we seek to elucidate the factors driving the choice between different partitioning and compression techniques. Additionally, we examine the landscape of compression standards such as JPEG and PNG, shedding light on their underlying principles and practical implications.

By synthesizing existing literature, experimental findings, and emerging trends, this comparative analysis aims to serve as a valuable resource for researchers, practitioners, and developers navigating the complex terrain of image processing and compression. Ultimately, our endeavor is to foster deeper insights, informed decision-making, and innovative advancements in this dynamic and indispensable field.

II. RELATED WORK

Image partitioning, the process of dividing an image into meaningful regions or segments, is a crucial step in various image processing tasks such as object detection, image segmentation, and content-based image retrieval. Over the years, numerous techniques have been proposed to address the challenges of image partitioning, each offering unique advantages and limitations. In this literature review, we present an overview of some prominent image partitioning methods. Thresholding methods partition images based on pixel intensity values. Techniques such as global thresholding, Otsu's method, and adaptive thresholding have been widely used for image segmentation tasks [1, 2]. Region growing algorithms iteratively merge neighboring pixels or regions based on predefined criteria such as similarity in color, texture, or intensity. Region merging techniques, on the other hand, start with a fine partitioning of the image and iteratively merge regions to achieve larger segments [3, 4]. Graph-based approaches model images as graphs, where pixels or regions are represented as nodes, and edges encode relationships between them. Techniques such as minimum spanning tree-based segmentation, normalized cuts, and graph cuts have demonstrated effectiveness in partitioning images into coherent regions [5, 6].

Clustering algorithms such as k-means clustering and fuzzy c-means clustering partition images by grouping similar pixels or feature vectors into clusters. These methods are particularly useful for unsupervised image segmentation tasks [7, 8]. The watershed transform treats the gradient magnitude of an image as a topographic surface and simulates the flooding of this surface to partition the image into catchment basins. Watershed-based methods have shown promise in segmenting images with complex or irregular boundaries [9, 10]. With the advent of deep learning, convolutional neural networks (CNNs) have emerged as powerful tools for image partitioning. Fully convolutional networks (FCNs), U-Net, and Mask R-CNN are among the popular architectures used for semantic segmentation tasks, achieving state-of-the-art results on various datasets [11, 12]. Hybrid and multimodal approaches combine multiple segmentation techniques or modalities (e.g., color, texture, depth) to enhance the accuracy and robustness of image partitioning. These methods leverage complementary information from different sources to improve segmentation results [13, 14]. In conclusion, image partitioning remains an active area of research, with a diverse range of methods catering to different application scenarios and requirements. While traditional techniques provide foundational principles, recent advancements in deep learning offer promising avenues for achieving more accurate and efficient segmentation results.

Image compression is essential for reducing the storage and transmission requirements of digital images while maintaining acceptable visual quality. Over the years, a multitude of techniques and algorithms have been proposed to address this challenge. In this literature review, we provide an overview of some prominent image compression methods. Transform coding techniques [15, 16], such as the Discrete Cosine Transform (DCT) used in JPEG compression, transform the spatial domain image data into a frequency domain representation. This enables the concentration of image energy into fewer coefficients, facilitating efficient compression. Wavelet-based compression methods [17, 18] utilize wavelet transforms to decompose images into different frequency

bands. This multi-resolution representation allows for more localized analysis of image features, resulting in efficient compression with better preservation of fine details. Quantization [19, 20] is a crucial step in image compression, where the transformed coefficients are mapped to a finite set of discrete values. By reducing the precision of coefficients, quantization introduces loss in image quality but significantly reduces the amount of data to be encoded.

Entropy coding techniques [21], such as Huffman coding and Arithmetic coding, are applied to further compress the quantized coefficients by assigning shorter codes to more frequently occurring symbols. These lossless coding methods exploit statistical properties of the image data to achieve additional compression. Predictive coding schemes [22] leverage the correlation between neighboring pixels to predict the value of a pixel from its neighbors. Only the prediction errors, i.e., the differences between the predicted and actual pixel values, are encoded and transmitted, leading to efficient compression. Hybrid compression methods [23, 24] combine multiple compression techniques to exploit their complementary strengths. For example, the JPEG 2000 standard combines wavelet transform with arithmetic coding to achieve superior compression performance. With the advent of deep learning, neural network architectures have been proposed for image compression tasks [25, 26]. These architectures learn to efficiently represent images using compact codes, often surpassing traditional methods in terms of compression efficiency and visual quality. Image compression techniques are categorized into lossy and lossless compression based on whether they preserve all image information or sacrifice some details to achieve higher compression ratios [27, 28]. Fractal coding is explored with spiral architecture for lossy compression [29, 30]. The choice between lossy and lossless compression depends on the specific requirements of the application. Image compression remains an active area of research, with continuous advancements in algorithms and methodologies. While traditional techniques provide foundational principles, emerging approaches such as deep learning-based compression offer promising avenues for achieving higher compression efficiency and better visual quality. Image partitioning and compression mechanisms are vital components of modern image processing and data compression systems. While traditional techniques provide foundational principles, ongoing research efforts focus on advancing algorithms, enhancing compression efficiency, and addressing emerging challenges in image analysis and transmission.

III. EXPLORATION OF MATHEMATICAL MODEL FOR IMAGE PARTITIONING

Creating a mathematical model for image partitioning involves representing the process of dividing an image into meaningful regions or segments using mathematical equations or algorithms. Let I be the input image with dimensions $M \times N$, where M is the height and N is the width of the image. We can represent the image I as a matrix of pixel intensities. Let $S = \{S_1, S_2, \dots, S_k\}$ be the set of segments obtained after partitioning the image I , where k is the total number of segments. Define criteria C for segmenting the image, which could be based on color, texture, intensity, or any other feature. Let $f(I_{ij})$ be the value of the criterion for pixel I_{ij} . The partitioning algorithm determines the segments S_i based on the segmentation criteria C . This can be achieved using various techniques such as thresholding, region growing, graph cuts, or clustering. Each segment S_i can be represented by a set of pixels P_i , where $P_i = \{(x_1, y_1), (x_2, y_2), \dots, (x_{n_i}, y_{n_i})\}$, and n_i is the number of pixels in segment S_i . The mathematical model for image partitioning can be summarized as: $S = \text{Partition}(I, C)$.

Let's consider a simple thresholding-based image partitioning algorithm. Given an input image I and a threshold value T , we can partition the image into two segments based on pixel intensity:

1. Define the criterion C as the pixel intensity I_{ij} .
2. Partition the image I based on the threshold T :

$$S_1 = \{(x, y) | I_{x,y} \leq T\}$$

$$S_2 = \{(x, y) | I_{x,y} > T\}$$
3. Return the set of segments $S = \{S_1, S_2\}$.

This example demonstrates a basic mathematical model for image partitioning using thresholding. More sophisticated algorithms can be developed based on different segmentation criteria and partitioning techniques.

To create a mathematical model for image partitioning, we can represent the image as a graph and formulate partitioning as a graph partitioning problem. Represent the image as an undirected graph $G = (V, E)$, where: V is the set of vertices representing pixels in the image. E is the set of edges connecting neighboring pixels. Assign weights to edges based on the similarity between pixel values. This can be done using various metrics such as color similarity, intensity difference, or texture similarity. Let w_{ij} denote the weight of the edge connecting vertices i and j . Partition the graph G into k disjoint subsets $S_1, S_2, \dots, S_k$, where k is the desired number of segments. This partitioning can be formulated as an optimization problem aiming to minimize the cut cost between partitions while maximizing the similarity within each partition.

Define an objective function that captures the trade-off between intra-segment similarity and inter-segment dissimilarity: minimize $\sum_{i=1}^k \sum_{j=1}^k w(S_i, S_j)$ where $w(S_i, S_j)$ is the cut cost between partitions S_i and S_j . Ensure that each pixel belongs to exactly one segment: $\forall v \in V: \sum_{i=1}^k x_{vi}$ where x_{vi} is a binary variable indicating whether pixel v belongs to segment i . Solve the optimization problem to obtain the optimal partitioning of the image into k segments. Consider a simple case where we aim to partition a grayscale image into two segments based on intensity similarity. We can represent the image as a graph, where vertices correspond to pixels, and edges connect neighboring pixels. Weights of edges can be assigned based on the absolute difference in intensity values between pixels. By formulating the objective function and constraints, we can solve the optimization problem using techniques such as spectral clustering or graph cut algorithms to obtain the desired partitioning. This mathematical model provides a framework for image partitioning based on graph theory, allowing for flexibility in defining similarity metrics and partitioning constraints. Table I provides a concise comparison of various image partitioning mechanisms based on their descriptions, advantages, limitations, and typical applications. Table II provides an overview of the performance of different image partitioning mechanisms based on various criteria, including accuracy, computational complexity, robustness, and scalability.

TABLE I. COMPARATIVE ANALYSIS OF DIFFERENT IMAGE PARTITIONING MECHANISMS

Mechanism	Description	Advantages	Limitations	Applications
Thresholding	Segments image based on pixel intensity	Simple, computationally efficient	Sensitive to noise, may require tuning	Object detection, document analysis
Region Growing/Merging	Iteratively merges neighboring pixels	Handles local variations well	Sensitivity to initial seeds	Medical imaging, object recognition
Graph-Based Segmentation	Models image as a graph, cuts connections	Handles complex structures	Computationally intensive	Medical imaging, remote sensing
Clustering Algorithms	Groups pixels based on similarity	Unsupervised, adaptable	Depends on initial parameters	Image retrieval, pattern recognition
Watershed Transform	Simulates flooding on gradient map	Preserves object boundaries	Over-segmentation in noisy images	Medical imaging, microscopy
Deep Learning-Based	Utilizes neural networks for segmentation	High accuracy, learns complex features	Requires large labeled datasets	Semantic segmentation, autonomous driving

TABLE II. PERFORMANCE ASSESSMENT OF IMAGE PARTITIONING MECHANISMS

Mechanism	Accuracy	Computational Complexity	Robustness	Scalability
Thresholding	Medium to High	Low	Low	High
Region Growing/Merging	High	Medium	Medium to High	Medium to High
Graph-Based Segmentation	High	High	High	Low to Medium
Clustering Algorithms	Medium to High	Medium to High	Medium	High
Watershed Transform	Medium to High	Medium to High	Medium to High	Low to Medium
Deep Learning-Based	High	High	High	High

IV. EXPLORATION OF MATHEMATICAL MODEL FOR IMAGE COMPRESSION

A mathematical model for image compression involves representing the process of reducing the size of digital images while preserving their visual quality using mathematical equations or algorithms. Let I be the input image with dimensions $M \times N$, where M is the height and N is the width of the image. The image can be represented as a matrix of pixel intensities. Apply a transformation T to the image I to convert it into a more compressible representation. This could be a Discrete Cosine Transform (DCT) or a Wavelet Transform. Let $I' = T(I)$ represent the transformed image. Quantize the transformed image I' to reduce the precision of its coefficients. This step introduces loss in image quality but significantly reduces the amount of data to be encoded. Let $Q(I')$ represent the quantized image.

Apply entropy coding techniques such as Huffman coding or Arithmetic coding to further compress the quantized coefficients. These techniques assign shorter codes to more frequently occurring symbols, exploiting statistical properties of the data for compression. Define the compression ratio (CR) as the ratio of the size of the original image (S_o) to the size of the compressed image (S_c): $CR = S_o/S_c$. Evaluate the quality of compression using objective metrics such as Peak Signal-to-Noise Ratio (PSNR), Mean Squared Error (MSE), or Structural

Similarity Index (SSI). Higher values of PSNR and SSI indicate better compression quality, while lower values of MSE indicate less distortion. Optimize the compression process to achieve a balance between compression ratio and compression quality. This involves tuning parameters such as quantization step size, entropy coding scheme, and transformation method.

Consider a simple example of image compression using the JPEG standard:

1. Apply the Discrete Cosine Transform (DCT) to the input image I to obtain the transformed image I' .
2. Quantize the transformed image I' using a quantization matrix.
3. Apply Huffman coding to encode the quantized coefficients.
4. Calculate the compression ratio and compression quality metrics such as PSNR and MSE to evaluate the performance of compression.

This example illustrates the basic steps involved in image compression and provides a foundation for more advanced compression techniques. Table III provides a simplified comparison of various image compression mechanisms based on their descriptions, advantages, limitations, and typical applications. Table IV provides an overview of the performance of different image compression mechanisms based on various criteria, including compression ratio, compression speed, image quality, complexity, and format flexibility.

TABLE III. SIMPLIFIED COMPARATIVE ANALYSIS OF DIFFERENT IMAGE COMPRESSION MECHANISMS

Mechanism	Description	Advantages	Limitations	Typical Applications
JPEG	Lossy compression using DCT	High compression ratios	Loss of image quality	Web images, digital photography
JPEG 2000	Wavelet-based compression	Efficient compression of large images	Higher computational complexity	Medical imaging, satellite imagery
PNG	Lossless compression using DEFLATE	Preserves image quality	Larger file sizes	Lossless image storage
GIF	Lossless compression using LZW	Supports animation	Limited color palette	Web graphics, animations
HEVC (H.265)	Advanced video compression standard	Higher compression efficiency	Higher encoding complexity	Video streaming, broadcasting
WebP	Lossy and lossless compression format	Supports transparency and animation	Limited browser support	Web images, digital animation

TABLE IV. PERFORMANCE ASSESSMENT OF IMAGE COMPRESSION MECHANISMS

Mechanism	Compression Ratio	Compression Speed	Image Quality	Complexity	Format Flexibility	Typical Applications
JPEG	High	Fast	Good	Low	Lossy	Web images, digital photography
JPEG 2000	Very High	Medium to Slow	Excellent	Medium to High	Lossy and Lossless	Medical imaging, satellite imagery
PNG	Low to Moderate	Medium	Excellent	Low	Lossless	Lossless image storage
GIF	Moderate	Fast	Good	Low	Lossless	Web graphics, animations
HEVC (H.265)	Very High	Slow	Excellent	High	Lossy	Video streaming, broadcasting
WebP	High	Medium	Good to Excellent	Medium	Lossy and Lossless	Web images, digital animation

V. CONCLUSION

Image partitioning and compression mechanisms play pivotal roles in various aspects of image processing, storage, and transmission. Image partitioning techniques are essential for segmenting images into meaningful regions or segments, facilitating various image analysis tasks such as object detection, recognition, and segmentation. From thresholding-based methods to sophisticated graph-based segmentation algorithms, there exists a wide range of techniques catering to different types of images and applications. While each method has its strengths and weaknesses, the choice of technique depends on factors such as image characteristics, computational resources, and specific requirements of the application. Image compression is crucial for reducing the size of digital images while maintaining acceptable visual quality. From traditional techniques like JPEG to modern standards such as JPEG 2000 and HEVC (H.265), image compression algorithms have evolved to achieve higher compression ratios, better image quality, and increased flexibility. Lossy compression methods

sacrifice some image information to achieve higher compression ratios, while lossless compression methods preserve all image details but typically result in larger file sizes. The choice between lossy and lossless compression depends on factors such as the application requirements, available bandwidth, and storage constraints. Image partitioning and compression mechanisms are indispensable components of image processing workflows, enabling efficient analysis, storage, and transmission of digital images. While image partitioning facilitates the extraction of meaningful information from images, image compression optimizes storage and bandwidth utilization without significant loss of visual quality. Advances in these areas continue to drive innovation in fields such as computer vision, multimedia communication, medical imaging, and remote sensing. As technology evolves, it is essential to explore novel techniques and methodologies to address emerging challenges and enhance the efficiency and effectiveness of image partitioning and compression mechanisms.

REFERENCES

- [1] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," in *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62-66, Jan. 1979, doi: 10.1109/TSMC.1979.4310076.
- [2] R. C. Gonzalez and R. E. Woods, "Digital Image Processing," 4th ed. Pearson, 2018.
- [3] V. Dey, Y. Zhang, and M. Zhong (2010). A review on image segmentation techniques with remote sensing perspective. In: Wagner W., Székely, B. (eds.) ISPRS TC VII Symposium – 100 Years ISPRS, Vienna, Austria, July 5–7, 2010, IAPRS, Vol. XXXVIII, Part 7A.
- [4] Jianbo Shi and J. Malik, "Normalized cuts and image segmentation," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888-905, Aug. 2000, doi: 10.1109/34.868688.
- [5] V. R. Parihar and N. V. Thakur, "Graph Theory Based Approach for Image Segmentation Using Wavelet Transform", *International Journal of Image Processing (IJIP)*, vol. 8, no. 5, pp. 255-277, (2014).
- [6] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124-1137, Sept. 2004, doi: 10.1109/TPAMI.2004.60.
- [7] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, no. 14, pp. 281–297, 1967.
- [8] J. C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*, Springer, pp. 95-99, 1981.
- [9] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 583-598, June 1991, doi: 10.1109/34.87344.
- [10] R. Adams and L. Bischof, "Seeded region growing," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 6, pp. 641-647, June 1994, doi: 10.1109/34.295913.
- [11] J. Long, E. Shelhamer and T. Darrell, "Fully convolutional networks for semantic segmentation," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 2015, pp. 3431-3440, doi: 10.1109/CVPR.2015.7298965.
- [12] Ronneberger, O., Fischer, P., Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, W., Frangi, A. (eds) *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. MICCAI 2015. Lecture Notes in Computer Science(), vol 9351. Springer, Cham. https://doi.org/10.1007/978-3-319-24574-4_28
- [13] P. Arbeláez, M. Maire, C. Fowlkes and J. Malik, "Contour Detection and Hierarchical Image Segmentation," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 898-916, May 2011, doi: 10.1109/TPAMI.2010.161.
- [14] S. Xie and Z. Tu, "Holistically-Nested Edge Detection," *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, 2015, pp. 1395-1403, doi: 10.1109/ICCV.2015.164.
- [15] A. B. Watson, "Image Compression Using the Discrete Cosine Transform," *The Mathematica Journal*, vol. 4, no. 1, pp. 81-88, 1994.
- [16] W. B. Pennebaker and J. L. Mitchell, "JPEG: Still Image Data Compression Standard," Kluwer Academic Publishers, 2004.
- [17] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," in *IEEE Transactions on Information Theory*, vol. 36, no. 5, pp. 961-1005, Sept. 1990, doi: 10.1109/18.57199.
- [18] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674-693, July 1989, doi: 10.1109/34.192463.
- [19] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 243-250, June 1996, doi: 10.1109/76.499834.
- [20] Rajandeep Kaur, Pooja Choudhary, "A Review of Image Compression Techniques," *International Journal of Computer Applications*, vol. 142, no. 1 (May 2016), 8-11, doi: 10.5120/ijca2016909658.
- [21] D. A. Huffman, "A Method for the Construction of Minimum-Redundancy Codes," in *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098-1101, Sept. 1952, doi: 10.1109/JRPROC.1952.273898.

- [22] A. K. Jain, "Fundamentals of Digital Image Processing," Prentice-Hall, 1989.
- [23] A. Skodras, C. Christopoulos and T. Ebrahimi, "The JPEG 2000 still image compression standard," in *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 36-58, Sept. 2001, doi: 10.1109/79.952804.
- [24] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," in *IEEE Signal Processing Magazine*, vol. 15, no. 6, pp. 74-90, Nov. 1998, doi: 10.1109/79.733497.
- [25] J. Ballé, V. Laparra, and E. P. Simoncelli, "End-to-end Optimized Image Compression," Proceedings of the International Conference on Learning Representations (ICLR), 2017.
- [26] G. Toderici *et al.*, "Full Resolution Image Compression with Recurrent Neural Networks," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 5435-5443, doi: 10.1109/CVPR.2017.577.
- [27] Y. Bai, X. Liu, K. Wang, X. Ji, X. Wu and W. Gao, "Deep Lossy Plus Residual Coding for Lossless and Near-Lossless Image Compression," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 5, pp. 3577-3594, May 2024, doi: 10.1109/TPAMI.2023.3348486.
- [28] A. Said and W. A. Pearlman, "Image compression using the spatial-orientation tree," *1993 IEEE International Symposium on Circuits and Systems*, Chicago, IL, USA, 1993, pp. 279-282 vol.1, doi: 10.1109/ISCAS.1993.393712.
- [29] V. T. Nileshsingh and O. G. Kakde, "Fractal Color Image Compression on a Pseudo Spiral Architecture," *2006 IEEE Conference on Cybernetics and Intelligent Systems*, Bangkok, Thailand, 2006, pp. 1-6, doi: 10.1109/ICCIS.2006.252295.
- [30] N. V. Thakur and O. G. Kakde, "A Novel Compression Technique for Color Image Database," *2006 International Conference on Advanced Computing and Communications*, Mangalore, India, 2006, pp. 240-243, doi: 10.1109/ADCOM.2006.4289891.