International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Enhancing Image Retrieval Systems: A Comprehensive Review of Machine Learning Integration In CBIR

Maher Alrahhal[1], Vinod Kumar Shukla[2]

Submitted: 14/03/2024 **Revised :** 29/04/2024 **Accepted:** 06/05/2024

Abstract: This paper presents a comprehensive review of advancements and methodologies in Content-Based Image Retrieval (CBIR) systems, with a focus on integrating machine learning algorithms to enhance performance. It examines the CBIR framework, covering key stages from feature extraction to similarity measurement and image retrieval, and underscores the importance of both global (color, texture, shape) and local (keypoints, patterns) features in image representation. The paper explores various extraction methods and their effects on retrieval accuracy, categorizing features into global and local, and discussing their roles and limitations. The application of machine learning in CBIR is divided into unsupervised learning (clustering), supervised learning (classification), and deep learning. It evaluates algorithms like K-means, SVM, ANN, and CNN in the context of CBIR, analyzing recent literature to assess their functionality and challenges. Deep learning, especially CNNs, is highlighted as a promising approach due to its strengths in translation, scale, rotation invariance, and direct learning from data. The paper identifies research gaps, including issues related to effective feature fusion, the development of scalable methods for large databases, and the integration of machine learning for better semantic understanding. It concludes by emphasizing the importance of addressing these gaps to improve CBIR systems in terms of retrieval performance, scalability, and efficiency. This review provides valuable insights for researchers and practitioners, offering a detailed overview of current trends and future directions in CBIR.

Keywords: Content-Based Image Retrieval (CBIR), Machine Learning in CBIR, Image Feature Extraction, Global Image Features, Local Image Features, Semantic Gap in Image Retrieval, Visual Vocabulary, SIFT, HOG.

1. Introduction

The CBIR framework is divided into mandatory and optional stages, as illustrated in Figure 1. CBIR begins when a user submits a query image, and the processes applied to the query image are performed in the same sequence for all images in the database. These processes can be executed online for the user's submitted image or applied offline to dataset images before query submission. The framework may include an optional preprocessing stage, involving operations like resizing, segmentation, denoising, and rescaling [1].

After preprocessing, the next crucial step is features extraction, where visual concepts are converted into numerical representations. Extracted features can be low-level (e.g., color, shape, texture) or local descriptors. This stage is essential, as it transforms the image's visual content into a form that machines can process and compare.

[1]*Postdoctoral Fellow, School of Engineering Architecture Interior Design, Amity University Dubai, United Arab Emirate*
Corresponding Author Email: malrahhal@amitydubai.ae
ORCID ID: 0000-0001-9688-343

[2]*Associate Professor, School of Engineering Architecture Interior Design, Amity University Dubai, United Arab Emirate*
vshukla@amityuniversity.ae
ORCID: 0000-0002-1143-0992

The final stage in CBIR is similarity measurement, where the features extracted from the query image are compared with those of the images in the database. The goal is to calculate the similarity or distance between the query image and each image in the dataset. This similarity measure helps in identifying and retrieving images that are most similar to the query image. After similarity computations, the images are sorted and ranked based on their similarity scores. The images most similar to the query image are ranked highest, while less similar images are ranked lower. Ranking can be done using distance values or similarity scores, with common metrics such as Euclidean distance, Cosine similarity, or others, depending on the feature extraction technique used [2-5].

Once the images are ranked, the system retrieves and displays the top-k images as the most relevant results. The value of k can be predefined or adjusted according to user preferences or system requirements.

The feature extraction step is the most important in CBIR, as it translates human perceptions into machine-readable numerical data. Features are generally classified as either global or local [6]:

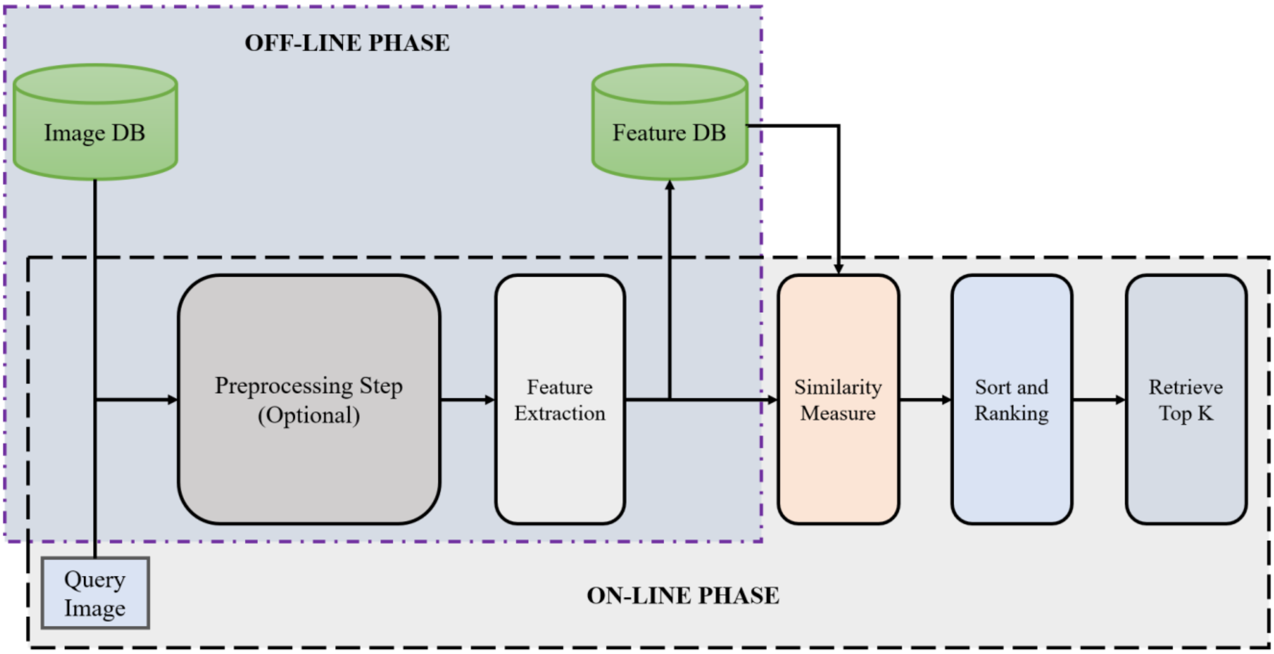

Fig 1. General framework of the CBIR system.

- **Global features** (e.g., texture, color, shape) provide an overall representation of the image. These algorithms are faster in calculating similarities but lack the ability to distinguish between objects and backgrounds.
- **Local features**, by contrast, focus on specific key points or regions of the image, such as corners, edges, or blobs. These features are more robust, allowing the system to handle changes in scale, rotation, and background.

Integrating these features into machine learning algorithms can significantly enhance CBIR performance. Recent developments in deep learning have further improved retrieval accuracy, although they often come with the trade-off of increased computational time [7].

Additionally, CBIR systems can generate high-dimensional features when converting visual content into numerical data, which can lead to performance degradation—commonly referred to as the "curse of dimensionality." Dimensionality reduction techniques can help mitigate this issue by simplifying the data while retaining essential information.

The effectiveness of a CBIR system also depends on the similarity measure used to compare feature vectors. An inappropriate similarity measure can lead to fewer relevant images being retrieved, reducing system accuracy. On the other hand, selecting the correct similarity measure can significantly improve accuracy.

CBIR performance is typically evaluated using metrics like precision and recall, which assess the system's effectiveness in retrieving relevant images.

The paper makes several key contributions to the field of CBIR:

- **Comprehensive Review of CBIR Frameworks**: It offers an in-depth review of advancements and methodologies in CBIR systems, with a particular focus on the integration of machine learning algorithms to enhance performance.

- **Detailed Analysis of Feature Extraction Methods**: The paper provides a thorough analysis of the CBIR framework, highlighting the significance of both global and local features in image representation. It explores various feature extraction methods and their influence on retrieval accuracy.

- **Exploration of Machine Learning in CBIR**: The research investigates the application of machine learning in CBIR, categorizing the discussion into unsupervised learning (clustering), supervised learning (classification), and deep learning, particularly focusing on CNNs. It evaluates various machine learning algorithms and their roles in CBIR systems.

- **Identification of Research Gaps**: The paper critically identifies research gaps in CBIR, including the need for efficient feature combination and fusion, the development of scalable methods for large-scale databases, and the integration of machine learning for improved semantic understanding.

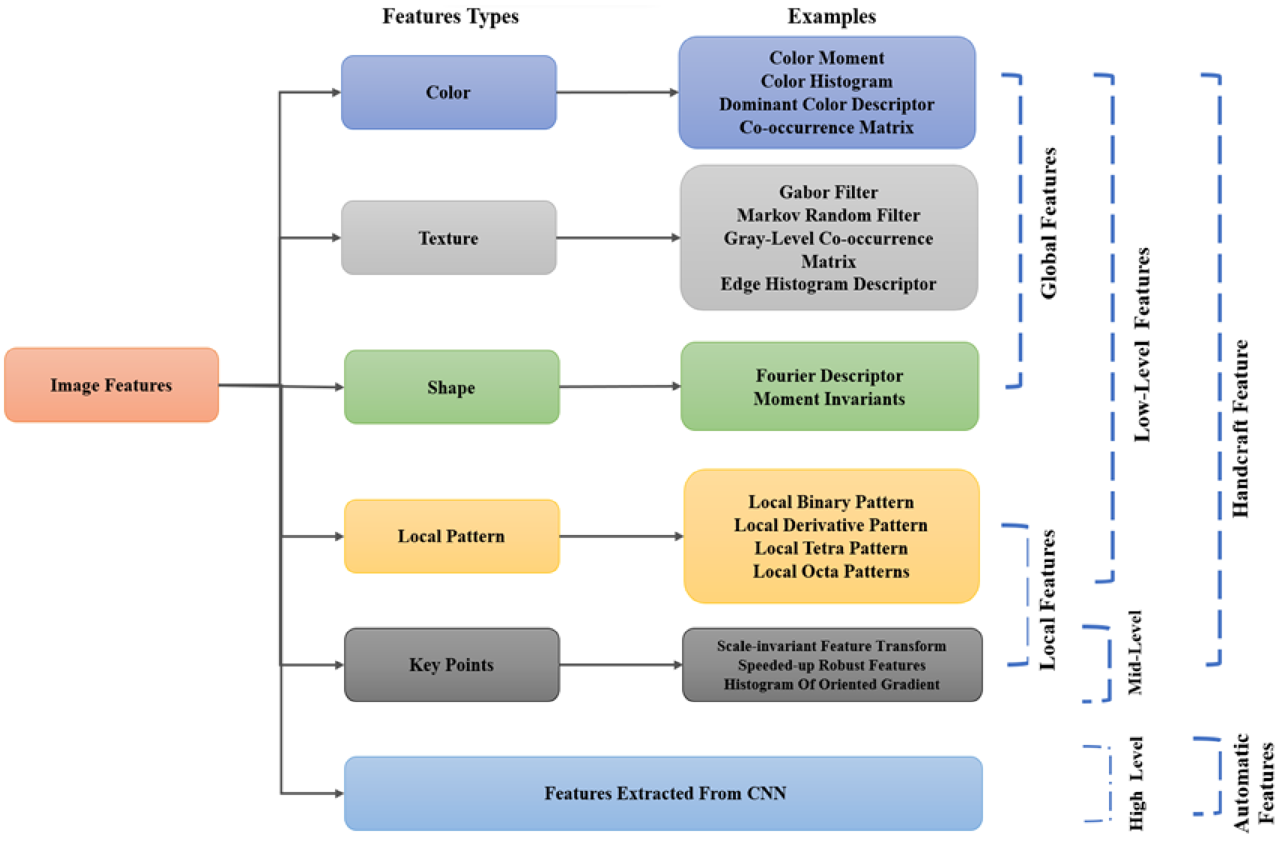

Fig. 2. Features category extracted from the Image.

The remainder of this paper is structured as follows: The second section examines the features extracted from images, focusing on the different types of features used in CBIR, particularly global and local features. This section discusses various extraction methods and their role in enhancing image retrieval accuracy. The third section explores machine learning-based CBIR, addressing unsupervised learning (clustering), supervised learning (classification), and the transformative role of deep learning, with a particular focus on Convolutional Neural Networks (CNNs). The fourth section presents a critical analysis of the current research gaps in the field. Finally, the fifth section concludes the paper by summarizing the key points and findings.

2. Features Extracted from The Images

CBIR focuses on the extraction and selection of features that represent the semantic content of images. **Global features** (such as color, texture, and shape) describe the entire image, providing an overall representation. In contrast, **local features** are obtained by segmenting the image or calculating key points, such as corners and edges. Local features are invariant to scale, rotation, and translation, making them more robust in diverse conditions. Figure 2 illustrates the categories of these features.

2.1. Global Features

Global features play a crucial role in CBIR systems as they capture the high-level visual characteristics of images. The most commonly used global features are color, texture, and shape, each of which contributes to image representation and description in distinct ways [8]. The main global features are:

A. Color Features

Color is a highly researched area in CBIR, as it is expected that three-dimensional color images will produce better results than one-dimensional grayscale images. Different color spaces can be used to represent colors in a similar manner, with common spaces including RGB, HSV, and CIE. Several descriptors are employed to characterize these color spaces, such as color moments, color correlations, color histograms, dominant color descriptors, and color co-occurrence matrices.

Color features are considered robust because they are invariant to translation, rotation, and scale changes. However, they do have spatial limitations, which can be addressed by using additional descriptors to compensate for this constraint.

B. Texture Features

Texture is an important property used in CBIR, as it provides a measure of surface qualities like smoothness, coarseness, and regularity by analyzing the variation in surface intensity. However, texture-based image retrieval is often hampered by computational complexity and sensitivity to noise. Despite these challenges, various algorithms are employed to analyze texture, including the Gabor filter, Markov random fields, Wavelet transform, Gray Level Co-occurrence Matrix (GLCM), and Edge Histogram Descriptor (EHD).

C. Shape Features

Shape is a low-level feature in CBIR that identifies objects within an image. Shape extraction can be based on either the boundary or the region of an object. In the region-based method, features are extracted from the entire region of the object, while the boundary-based method focuses on features along the object's boundary. Various methods are used for shape feature extraction, including Fourier descriptors and moment invariants. Shape descriptors differ in their ability to handle translation and scaling, which is why they are often combined with other descriptors to improve accuracy.

In early CBIR studies, images were retrieved using a single feature type, but the results were unsatisfactory, as images typically contain multiple pictorial elements. To enhance retrieval accuracy and performance, researchers began to fuse two or more features—such as color, texture, and shape—into a single system. This process is known as feature fusion.

D. Global Feature-Based Existing Studies

Shrivastava and Tyagi [9] proposed a feed-forward architecture for a CBIR system designed to retrieve relevant images through a three-stage process. In the first stage, N images are retrieved from a dataset of M images using color features, calculated via color histograms. The second stage refines the results by selecting P-relevant images from the subset of N images using texture features, which are extracted through Gabor filters. Finally, in the third stage, shape features are obtained using Fourier descriptors to retrieve K-relevant images from the set of P images. The system supports relevance feedback by adjusting the values of N, K, and P to improve precision. The system was evaluated on the Corel and CIFAR datasets, achieving precision estimates of 0.769 and 0.859, respectively. However, the system lacks a block for spatial information-based image classification.

Younus et al. [10] introduced a new CBIR system aimed at extracting both texture and color features. The authors employed four feature extraction methods, including color moments (color histograms), wavelet moments, and co-occurrence matrices. To enhance the clustering process, the authors combined the k-means algorithm with particle swarm optimization (PSO). The system was evaluated on the WANG dataset, which consists of 1,000 images classified into 10 categories. The results demonstrated improved accuracy in most categories, except for architecture and buses. However, it is important to note that the method did not incorporate shape features in its similarity distance calculation.

Ponomarev et al. [11] presented a CBIR system that integrates color, texture, and shape features. The system used the Gabor, and wavelet transforms to extract these features, with Manhattan distance employed to measure similarity between the query images and the dataset. The system was tested on three datasets: Corel, Li, and Caltech101. The system achieved average precision values of 0.83, 0.88, and 0.7, respectively. Despite its promising results, the system's complexity is increased due to the integration of multiple feature types.

Srivastava and Khare [12] introduced a novel CBIR algorithm that utilizes multi-resolution analysis to enhance image retrieval. The algorithm analyzes images at multiple levels, allowing it to capture details that may be overlooked when using a single level of analysis. Texture information is extracted using the Local Binary Pattern (LBP), while shape features are obtained using Legendre moments. By combining multiple local and global features derived from the LBP descriptor, a robust feature vector is formed. The proposed technique was evaluated on five datasets and demonstrated superior performance in terms of both accuracy and sensitivity compared to standard methods. However, it is worth noting that the computational cost of the algorithm increases due to the multi-resolution analysis.

Sajjad et al. [13] developed an invariant CBIR system designed to handle texture rotation and color changes. The system integrates texture and color attributes into a 360-dimensional feature vector. In the color feature extraction step, images are converted to the HSV color space and quantized using a color histogram. Only the Hue and Saturation channels are considered to ensure illumination invariance. To extract texture features, the system employs Rotated Local Binary Patterns (RLBP), which capture rotationally invariant texture information. The CBIR system was tested on the Corel 1K and Corel 10K datasets, and its performance was evaluated based on its ability to retrieve relevant images using texture-based features, color-based attributes, and its robustness to texture rotation and color variation.

Zheng et al. [14] proposed a CBIR approach based on block processing with overlapping. The approach involves several steps. First, images are transformed into the HSI color space to facilitate color feature extraction. Next, the images are divided into blocks, with the central block selected for further analysis. Color features are extracted using histogram projection, which captures the color distribution within the selected block. Texture features are extracted using the Roberts Edge detection method, designed to identify edges and intensity differences. To calculate image similarity, the authors used a weighted Euclidean distance metric, with the weights determined through trial and error. However, the proposed CBIR system demonstrated lower performance compared to other state-of-the-art methods, such as those outlined by Tadi Bani and Fekri-Ershad [20].

Zhao et al. [15] presented a novel approach in the field of CBIR by combining color, shape, and texture features. The proposed approach utilizes the following feature extraction methods:

- Color Features: Color Distribution Entropy (CDE) is employed to retrieve color characteristics, depicting the scattering of colors within images.
- Shape Features: Hue Moments are used for shape feature extraction, providing insights into the shapes of objects within the images.
- Texture Features: The Color Level Co-occurrence Matrix (CLCM) is utilized to retrieve texture characteristics, capturing the spatial relationships between various color levels.

To measure the similarity between the query image and the dataset images, the authors applied a weighted normalized similarity measure, where the weights are user-defined based on the user's experience or domain knowledge. While the proposed CBIR system achieved high precision, it struggled with images containing complex objects. This limitation arises from the use of Hue Moments for shape attributes, as it may fail to accurately identify images containing multiple objects or may incorrectly interpret separate edges as a single edge.

Phadikar et al. [16] introduced a novel CBIR system that operates in the compressed domain, specifically within the Discrete Cosine Transform (DCT) domain. In this system, color moments, color histograms, and edge histograms are extracted in the compressed domain. To enhance retrieval performance, a Genetic Algorithm (GA) is employed to assign varying levels of relevance to the extracted features based on their dissimilarity.

The application of GA positively impacted the precision of the CBIR system, but also increased the computation time required for retrieval. However, the authors found that features extracted in the compressed domain helped balance the total time spent on image retrieval. Overall, the combination of compressed domain feature extraction with GA feature weighting improved the system's accuracy in retrieving relevant images, striking a balance between retrieval performance and computational efficiency.

Pavithra and Sharmila [17] introduced a multi-step CBIR method aimed at enhancing image retrieval performance by reducing search and computational costs.

- In the first step, color features were computed using color moment measurements, including mean and standard deviation for each channel in the RGB color space. This color feature extraction helped reduce the search space and, consequently, the computational requirements.
- The second stage involved extracting texture and shape features from a sub-dataset created in the first stage. LBP were used to extract texture information, while the Canny edge detector was employed to capture edge information. These added features provided additional discriminative information, improving precision in retrieval.
- The search process utilized Manhattan distance as the similarity metric between the query image and database images.

The multi-stage CBIR system demonstrated higher precision and lower processing time compared to other approaches. However, the authors noted that the system's runtime is influenced by the size of the dataset. They also suggested that integrating artificial intelligence algorithms could further improve the system's performance, and diversifying the type and size of the datasets could enhance the retrieval outcomes. Overall, the proposed multi-stage CBIR technique successfully combined color, texture, and shape features, resulting in improved retrieval accuracy and computational efficiency.

In another study, Pavithra and Sharmila [18] proposed an innovative method for seed point selection in dominant color-based image retrieval techniques. Their approach was designed to enhance the retrieval capabilities of dominant color descriptors used for image retrieval. The authors tested their dominant color descriptor on four image databases, and the results demonstrated higher retrieval accuracy compared to existing methods. However, it is important to note that the proposed strategy focused solely on dominant color information, without considering shape, texture, or spatial information. This limitation suggests a semantic gap in the retrieval process, as images with the same dominant color may belong to different semantic classes.

To close this semantic gap and obtain more relevant retrieval results, the authors suggested combining their approach with other feature extraction methods, such as shape, texture, and spatial characteristics. By incorporating multiple features, the system could capture a broader range of image properties and improve retrieval accuracy,

TABLE 1. Summary of the Literature on Global Feature-Based Methods

REF	Features	Feature Extraction Method	Methods	Limitations
[9]	Color, Texture, Shape	Color Histogram, Gabor Filter, Fourier Descriptor	Three-stage image retrieval process.	Color histogram lacks spatial information. Fourier descriptor entails high computational cost.
[10]	Color, Texture	Color Moment, Color Histogram, Wavelet Moment, Co-occurrence Matrix	Combined k-mean clustering with particle swarm optimization (PSO).	Prone to irrelevant image retrieval due to low clustering algorithm accuracy.
[11]	Color, Texture, Shape	Color Auto Correlogram, Gabor Transform, Wavelet Transform	Incorporates color, texture, and shape features for accurate similarity measurement.	High computational cost due to using multiple features.
[12]	Texture, Shape	Wavelet Transform/LBP, Legendre Moments	Combines local and global features, analyzing images at multiple levels.	Increased computational cost.
[13]	Color, Texture	Quantized Color Histogram, RLBP	Extracts color features in HSV color space, ensuring illumination invariance. Utilizes RLBP for rotation-invariant texture features.	Lack of clear information about Corel dataset results.
[14]	Color, Texture	Color Histogram, DWT, EDH	Combines local and global features. Uses EHD to incorporate local edge distribution.	Does not involve any machine learning algorithms.
[15]	Color, Texture, Shape	CDE, CLCM, Hue Moments	Considers color, texture, and shape features for similarity measurement. High accuracy.	Accuracy is contingent on the nature of the query image.

Table 1. Summary of the literature on global feature-based methods

REF	Features	Feature Extraction Method	Methods	Limitations
[16]	Color, Texture	Color Histogram, Color Moment, MPEG-7 Edge Descriptor	Incorporates color and texture features in similarity measurement. Utilizing GA enhances system accuracy with excellent results.	Genetic Algorithm (GA) introduces an impact on computational cost.
[17]	Color, Texture, Shape	Color Moment, LBP, Canny Edge Detector	Averts linear dataset search.	Running time varies based on the number of images in the new sub-dataset.
[18]	Color	Dominant Color Descriptor	Introduces seed point selection to mitigate the drawback of DCD.	Falls short of bridging the semantic gap.
[19]	Color, Shape	Color Histogram, Canny Edge Histogram	Implements an ANN to discern semantic class information.	Lacks spatial information and provides no details about running time.
[20]	Color, Texture	Quantized Color Histogram, Gabor Filter, GLCM	Extracts features in spatial and frequency domains, demonstrating invariance to rotation and low sensitivity to noise.	Exhibits high run time.
[21]	Color, Texture, Shape	Color Moment, Ranklet Transformation, Invariant Moment	Integrates non-parametric and parametric features.	High computational cost arises due to the high feature vector dimension.
[22]	Color, Texture, Shape	Color Moment, GLCM, Geometric Shape Feature	Considers color, texture, and shape features in similarity measurement.	Potential for reducing retrieval time with the application of a suitable optimization algorithm.
[23]	Color, Texture	Color Moment, DWT/Gabor Filter/CEDD	Achieves high accuracy values.	High computational cost attributed to the high feature vector dimension.

ensuring that images with similar semantic content are accurately retrieved.

Ashraf et al. [19] introduced a novel CBIR system aimed at bridging the gap in image retrieval efficiency. The proposed method combines color and edge features to form a comprehensive feature descriptor. For color feature extraction, a color histogram analysis was employed, while the Canny edge detector was used for edge extraction in the YCbCr color space.

To further enhance feature representation, discrete wavelet transformation was applied, with the Haar wavelet selected due to its computational efficiency compared to other wavelet functions. To classify images into semantic categories, an artificial neural network (ANN) was trained using the extracted features. It is important to note that the training and testing of the ANN require significant computational resources. The system used the Manhattan distance metric for similarity measurement. The results showed good precision and recall performance, demonstrating the system's effectiveness. However, the lack of spatial information limits the system's ability to capture fine details and spatial relationships in images. Additionally, no information was provided regarding the system's computational cost, preventing an assessment of its scalability for large datasets.

Tadi Bani and Fekri-Ershad [20] proposed a novel CBIR system designed to extract global and local textures in both frequency and spatial domains. Color features were obtained in the spatial domain, with Gaussian filtering applied as preprocessing to reduce noise. Global texture features were extracted using the Gray Level Co-occurrence Matrix (GLCM) in the spatial domain, while color features were extracted using quantized color histograms within the RGB color space. Local texture features were derived using the Gabor filter to improve retrieval performance. The system was tested on the Simplicity dataset, achieving high precision values compared to other approaches. It was also found to be rotation-invariant and able to tolerate moderate levels of noise. However, the system's runtime was longer due to the use of multiple feature types.

Rana et al. [21] introduced a new CBIR approach that combines both parametric (color, shape) and nonparametric (texture) features. Parametric features were analyzed using color moments and moment invariants, while nonparametric features were analyzed using the ranklet transform. The resulting feature vector had a length of 247, which slowed down the algorithm, a notable drawback. The method was tested on five datasets. Additionally, Bella and Vasuki [22] proposed the FIF-IRS method, which fused color moments in the HSV color space with GLCM in eight directions. The evaluation metrics for FIF-IRS included precision, retrieval speed, and error rate, indicating it as a promising approach. Retrieval time could be further reduced through the use of optimization algorithms.

Ashraf et al. [23] proposed a CBIR system that combines low-level features, such as texture and color, to improve image retrieval performance. Color features were extracted using color moments within the HSV color space, while texture features were obtained using Discrete Wavelet Transform (DWT) and Gabor wavelet techniques. The system integrated color and edge descriptors into a feature vector, resulting in more accurate retrieval results. However, the large size of the feature vector also increased the time required for search and comparison. The proposed system demonstrated high precision and recall in experiments conducted on the Corel 1000 and Corel 15,000 datasets. It is important to note that, like many other methods in the literature, this approach lacks spatial and comprehensive texture information.

Alsmadi et al. [24] introduced a new CBIR system that integrates shape, color, and texture features. The system extracted color features using the Canny edge histogram in the YCbCr color space, while texture features were derived using the Gray Level Co-occurrence Matrix (GLCM). Shape features were extracted using the Canny edge method in the RGB color space. The system utilized simulated annealing (SA) and a genetic algorithm (GA) to enhance the solution quality. The CBIR system outperformed other state-of-the-art approaches, achieving an average precision of 0.901 and a recall average of 0.1803. However, it is important to consider that the cooling process in the SA method and the need for numerous iterations may lead to extended computation times.

Table 1 provides a comprehensive overview of literature focused on methods based on features such as color, texture, and shape.

2.2. Local Features

The use of local image features in CBIR is becoming increasingly popular due to their advantages over global features. Unlike global features, local features are invariant to scale and rotation, making them more robust under various conditions. They provide consistent matching by capturing detailed information from specific regions of an image. This makes CBIR systems more accurate and efficient in retrieving relevant images from large datasets. Consequently, the adoption of local visual features highlights their effectiveness in addressing scale and rotation invariance issues, ultimately enhancing the overall performance and quality of CBIR systems.

A. Scale-Invariant Feature Transform

SIFT (Scale-Invariant Feature Transform) is one of the most renowned local descriptors in image analysis, introduced by David Lowe [25]. It consists of a keypoint detector and a descriptor extractor. One of SIFT's key advantages is its invariance to image rotation and scaling, making it useful in several applications. However, SIFT has limitations in high-dimensional matching tasks and requires a fixed-size vector encoding mode for image similarity comparisons. These factors contribute to its shortcomings in image retrieval, particularly due to its high memory usage and computational demands in certain cases, rendering it less effective in some scenarios.

Montazer and Giveki [26] introduced a CBIR system utilizing SIFT and Local Derivative Pattern (LDP) to create feature descriptors. To mitigate the high memory consumption and computational costs associated with SIFT, the authors proposed two dimensionality reduction techniques. The CBIR system was evaluated on four databases and demonstrated excellent retrieval performance for object-based images. However, additional refinements are necessary for the system to perform effectively with natural images.

Sharif et al. [27] proposed a CBIR system that unifies visual words produced by SIFT and Binary Robust Invariant Scalable Key Points (BRISK) methods. BRISK, a key component of the system, addresses SIFT's limitations in handling low-light conditions and poorly localized key points. To reduce computational costs, the system allows for selecting different percentages of image features. However, this proposed method has yet to be tested on large, unlabeled datasets.

B. Speeded-Up Robust Features

Bay et al. [28] introduced the SURF algorithm (Speeded Up Robust Features), a robust local descriptor that addresses SIFT's limitation of high dimensionality. While dimensionality reduction techniques can be applied to mitigate this drawback, they may impact system performance during feature computation. Inspired by the SIFT algorithm, the authors developed SURF to offer faster speeds and greater robustness. SURF uses an indexing scheme based on the Laplacian symbol, enabling quicker feature calculation and image matching. However, SURF has limitations when handling image rotations.

In a related study, Jabeen et al. [29] proposed a new CBIR system that integrates FREAK (Fast Retina Keypoint) descriptors with SURF features. FREAK demonstrates superior classification performance, while SURF excels in managing changes in illumination and scale. This fusion of descriptors generates visual words using the Bag of Visual Words (BoVW) model, which helps reduce the semantic gap in image retrieval. The visual words are clustered using K-means, and histograms of visual words are created for each image. These histograms are then used to train an SVM classifier to recognize the semantic content in images. The proposed system was evaluated on three image collections, including Corel 1000 and Corel 150, and showed high efficiency in terms of average precision, retrieval accuracy, and computational complexity. It is worth noting that descriptors like FREAK and SURF do not inherently capture color information.

C. Histogram of Oriented Gradients (HOG)

Dalal and Triggs [30] proposed the Histogram of Oriented Gradients (HOG) as an improved descriptor that outperforms existing descriptors, such as wavelets. HOG estimates the shape and appearance of local objects based on the direction of edges or the distribution of local intensity gradients, without needing the precise location of the gradient or edge. The image is divided into smaller spatial regions, called cells, where the orientations of edges within each cell are summed to produce local 1D gradient direction histograms. These histograms are then combined to represent the image. HOG also calculates energy by comparing the local histograms of larger blocks and normalizes the cells within the blocks, enhancing its resistance to shadowing and illumination effects. Over the past decade, HOG has been widely used in various applications, particularly in object recognition.

Mehmood et al. [31] proposed a CBIR system that combines the strengths of HOG and SURF descriptors. SURF is used to extract local features and performs well on noisy images, low-illumination settings, and images with clear backgrounds. HOG, on the other hand, extracts global features, providing more spatial information and improving retrieval performance. The two descriptors, each with its own visual vocabulary, are combined to form a larger vocabulary. While a larger vocabulary can enhance retrieval results, it also increases computational costs. To mitigate this issue, the authors selected a subset of the extracted features for further processing.

The fused feature matrix was clustered using K-means++, and histograms for each image were computed. An SVM classifier was then used for image classification. The system was tested on four well-known datasets (Caltech 256, Corel 1K, Corel 1.5K, and Corel 5K). Although HOG is efficient, it cannot directly construct feature vectors from multispectral images, leading to a loss of spatial and spectral information.

Table 2. Summary of the Literature for Key Pints and Local Pattern-Based Methods.

REF	Features	Feature Extraction Method	Methods	Limitations
[26]	KeyPoint, Local Pattern	SIFT, LDP	Demonstrates high performance for images containing objects.	The length of the feature vector is 3000, and there's a need for improvement when applied to nature images.
[27]	KeyPoint	SIFT, BRISK	Aims to reduce the semantic gap between high-level and low-level features.	Not tested against large-scale unlabeled datasets.
[29]	KeyPoints	SURF, FREAK	Targets the reduction of the semantic gap between high-level and low-level features.	Does not provide any color information.
[31]	HOG, KeyPoint	HOG, SURF	Provides additional spatial information and performs well in noisy and low-illumination situations.	HOG cannot be directly used for multispectral images.
[34]	KeyPoint, Local Pattern	LBPV, LIOP	Aims to reduce the semantic gap between high-level and low-level features, utilizing PCA for dimensionality reduction.	Not tested against large-scale datasets, and not directly applicable to multispectral images.
[35]	KeyPoints	SFIT, LIOP	Strives to reduce the semantic gap between high-level and low-level features. Exhibits invariance to rotation, changing scale, illumination, and performs well in low-contrast cases.	Presents a high-dimensional descriptor.

C. Local Pattern

LBP (Local Binary Patterns) was introduced by Ojala et al. [32] as a qualitative technique for local pattern analysis. It compares the central pixel in a neighborhood to its surrounding neighbors using a threshold. LBP is robust, as it remains invariant under monotonic grayscale transformations, and is computationally efficient. However, LBP has some limitations.

Guo et al. [33] introduced LBP Variance (LBPV) as an extension of LBP to address these limitations. LBPV includes a global rotation-invariance step after applying the local variant of LBP. The authors also proposed a feature reduction technique using similarity measurements to speed up the matching process.

Sarwar et al. [34] presented a CBIR system that uses LIOP (Local Intensity Order Pattern) and LBPV to improve performance by reducing semantic gaps. These two feature descriptors are used to create smaller visual vocabularies, which are then combined into a larger visual dictionary. To reduce the size of the visual dictionary, principal component analysis is employed. Histograms are computed, and an SVM is trained using both LBPV and LIOP.

The system was evaluated on three datasets: Holidays (WANG-1K), WANG-2K, and WANG-2K. The system was efficient in terms of precision, recall, and computational cost. However, it has not been tested on large datasets like ImageNet or ImageCLEF, and it cannot construct feature vectors from multispectral images, leading to a loss of spectral and spatial information. A summary of Keypoints, Local Patterns, and Local Patterns from the literature is provided in Table 2.

3. Machine Learning Based Content Based Image Retrieval

In recent years, there has been a significant trend in CBIR systems towards the integration of machine learning algorithms to create models capable of handling new input data and providing accurate predictions. This integration has led to notable improvements in image search performance. This section is divided into three subsections: unsupervised learning (clustering), supervised learning (classification), and deep learning.

3.1. Unsupervised Learning (Clustering)

After the feature extraction process and feature vector construction are completed in a CBIR system, the next step is clustering. Clustering involves grouping image descriptors into separate clusters based on similarities, while ensuring that these clusters also exhibit semantic differences from each other. It is considered an unsupervised learning algorithm since it does not require predefined labels for the image data. Although K-means and K-means++ are the most common clustering algorithms used in CBIR, particularly with local feature extraction methods, other clustering techniques are rarely applied. These methods use local feature extraction from images and then apply the clustering process. Clustering algorithms process the feature vectors to find the optimal grouping of images based on similarity, which makes the management and retrieval of images in CBIR systems highly efficient.

As mentioned earlier, Yousuf et al. [35] used the K-means clustering algorithm in combination with a visual vocabulary created from SIFT and LIOP descriptors. This fusion increased the size of the visual vocabulary, which improved the image retrieval process. However, K-means clustering has some limitations. One issue is that it requires the number of clusters to be specified in advance, which can be challenging without prior knowledge or experimentation. Additionally, the choice of initial centroids influences the quality of the clustering. Incorrect selection of centroids may cause the algorithm to converge to local optima, thereby affecting the final clustering quality. Moreover, while using a larger number of clusters might reduce error, it can also lead to overfitting, where the model becomes too tailored to the training data and may perform poorly on unseen data.

Another drawback of K-means is its sensitivity to outliers and noise. Outliers can distort centroid calculations and cluster assignments, resulting in suboptimal clustering. Despite these limitations, K-means remains a popular clustering algorithm in CBIR systems, and researchers continue to develop techniques to overcome these challenges and improve its efficiency.

Mehmood et al. [31] applied the K-means++ algorithm to a combined visual dictionary created from HOG and SURF descriptors. K-means++ is an improvement over the traditional K-means algorithm, addressing some of its limitations. One of the key improvements is in the selection of initial centroids. K-means++ uses a weight-based method to assign probabilities to potential centroids, ensuring that the chosen centroids better represent the data distribution. This results in more accurate clustering from the outset.

Although the initial selection process for centroids in K-means++ is more complex and time-consuming compared to the standard K-means algorithm, it offers several advantages. By assigning weights to centroids, K-means++ avoids convergence to local optima, thereby improving the overall clustering accuracy.

TABLE 3. Classification Of the Main Characteristics, Limitations, And Examples for the Main Machine Learning Categories

METHODS	Main Characteristics	Limitations	Examples
SUPERVISED LEARNING	Requires the presence of labels. Predict and classify data to one of predefined classes.	Classification accuracy is directly impacted by the size of the training set.	SVM, ANN
UNSUPERVISED LEARNING	No need for labels. Learns the similarities and semantics between input data and generalizes a model to handle unseen inputs.	Overfitting, Scalability, and clustering algorithm performance are affected by the number of clusters.	K-means, K-means++
DEEP LEARNING	Can be Supervised or Unsupervised. Generates learning models. Predict and classify data to one of predefined classes. Generate learning model.	High computational cost and complex structure. Learns the similarities and semantics between input data and generalizes a model to handle unseen inputs.	CNN, Deep Neural Network, Deep Belief Network, Boltzmann Machine

Furthermore, the use of weighted centroids leads to faster convergence, requiring fewer iterations to reach the optimal clustering solution. This reduction in the number of iterations decreases the computational cost and enhances the efficiency of the clustering process. Overall, by incorporating K-means++ in their CBIR system, Mehmood et al. achieved better performance in terms of clustering accuracy and reduced computational overhead compared to the traditional K-means algorithm.

3.2. Supervised Learning (Classification)

In CBIR, supervised learning algorithms differ from unsupervised ones in that they have prior knowledge of image groups and their corresponding labels, making the task one of classification. These algorithms are provided with a training dataset consisting of labeled images, which helps them detect patterns and relationships between features and class labels. When a new image is presented, the supervised learning algorithm predicts the most appropriate predefined group or label for the image based on the knowledge it has learned from the training data.

A. Support Vector Machine

SVM (Support Vector Machine) is one of the most widely used supervised classifiers in pattern recognition and image classification applications. It classifies new data by associating it with predefined classes based on patterns learned from the training data. SVM can be categorized into two types: linear SVM and non-linear SVM [36, 37].

- **Linear SVM**: In a linear SVM, the feature space is linearly separable, meaning that classes can be separated by a straight line or hyperplane. The goal of a linear SVM is to find the optimal hyperplane that maximizes the margin between the classes, allowing for better generalization to new data. Linear SVM is particularly effective when the classes can be separated in the original feature space.

- **Non-linear SVM**: Often, data is not linearly separable in its original feature space. To address this, non-linear SVM uses kernel functions to map the data into a higher dimensional feature space where linear separation becomes possible. The "kernel trick" allows the computation of dot products in the transformed space without explicitly calculating the transformation. Common kernel functions, such as the radial basis function (RBF) or polynomial kernels, enable non-linear SVM to efficiently handle complex non-linear decision boundaries.

The selection of the kernel function is a crucial factor in determining the performance of SVM. Different kernels capture different types of non-linear relationships in the data. The choice of the appropriate kernel depends on the characteristics of the data and the specific problem at hand. To achieve optimal classification performance, it is important to test different kernel functions and fine-tune their parameters.

Overall, SVM is a powerful tool for supervised classification in CBIR and pattern recognition tasks. It is effective for both linearly separable and non-linearly separable datasets, offering accurate classification based on patterns learned from the training data.

B. Artificial Neural Networks

ANNs are widely used in practical applications, including image retrieval. The architecture of ANNs is designed to mimic the functioning of the human neuronal system, making them highly effective in data processing. ANNs are adaptable to solving problems across various domains and are characterized by features such as robustness, high parallelism, fault tolerance, noise resilience, and nonlinearity. ANNs operate by simulating interconnected neurons, also known as nodes or artificial neurons, which receive, process, and transmit information. These neurons are organized into layers, with each layer containing multiple nodes. The connections between nodes are weighted, where the weights represent the strength and influence of signals transmitted between them [38].

The weights of these connections are updated through optimization algorithms, such as backpropagation, which aims to minimize the difference between the network's output and the desired outcome during training. This learning process enables the network to identify and generalize patterns from input data, making it well-suited for image retrieval tasks. Due to ANNs' ability to process complex relationships and learn from large datasets, they are a powerful tool for image retrieval [39].

When trained on a labeled dataset of images, an ANN model can extract meaningful features and recognize patterns unique to certain image classes or attributes. This learned knowledge can then be applied to sort or retrieve images based on their similarity to the training data. The parallel processing capabilities of ANNs make computational tasks faster, especially with modern hardware designs, thus enhancing the efficiency of image retrieval. Additionally, ANNs are robust to errors and noise, enabling the system to handle imperfect or noisy input data, which is common in real-world scenarios where images may have defects or variations [40].

TABLE 4. Summary of the Performance of Machine Learning Algorithms-Based Approaches For CBIR

REF.	Machine Learning Algorithm	Method	Limitations
[31]	K-means++/SVM	Offers more spatial information. Performs better in noise and low illumination situations.	HOG cannot be directly applied to multispectral images.
[34]	SVM	Reduces the semantic gap between high-level and low-level features. Invariant to image rotation and monotonic intensity change. Using PCA to reduce dimensionality.	Not tested on large-scale datasets. Cannot be directly applied to multispectral images.
[35]	K-means/SVM	Reduces the semantic gap between high-level and low-level features	The resulting descriptor becomes high-dimensional.
[41]	ANN	Retrieval based on the image core (main) object.	Segmentation slows down the system.
[46]	CNN	No need for annotations or labels. The length of the feature vector is 16. Reduces required memory and run time.	Accuracy decreases with larger datasets.
[47]	CNN/Unsupervised CNN/Supervised	Reduces the dimension of the feature descriptor. Retains spatial information.	Increased retrieval time.
[48]	CNN	Uses VGG-16. Utilizes a similarity score.	Requires enhancement in terms of speed during the training and testing stages. More time to construct the gravitational field database.
[49]	CNN	Reduces computational cost.	Increased retrieval time if not using sparse representation.

Overall, ANNs' ability to emulate the human neuronal system, combined with their powerful information processing capabilities, makes them a valuable tool in the field of image retrieval. Their capacity to extract meaningful information from data and identify patterns leads to intelligent solutions in image analysis and retrieval tasks.

Ashraf et al. [41] proposed a CBIR system designed to retrieve images based on their main subjects. The system employs the Bandelet transform as a feature extraction technique, specifically applied to the primary object within images. For texture classification, an ANN based on backpropagation was used, with four categories: deficiency of contour blocks, vertical, horizontal, and left/right diagonal. The ANN architecture included a hidden layer with 20 neurons and an output layer with 4 neurons. Texture features were extracted from the ANN's output using Gabor filters.

In addition, color-based feature extraction was performed in both the YCbCr and RGB color spaces to enhance system performance. This was achieved through the use of color wavelets and color histograms. Another ANN was employed to classify the query image into its corresponding class and then compare it with images within the same class, thereby improving retrieval accuracy.

The approach introduced by Ashraf et al. was segmentation-based, which contributed to higher precision in image retrieval. However, this method can result in slower processing speeds compared to other techniques. In conclusion, the CBIR system proposed by Ashraf et al. integrates the Bandelet transform, ANN-based texture classification, and color feature extraction techniques to retrieve images based on their central objects. While segmentation led to improved precision, it came at the cost of slower processing speeds.

3.3. Deep Learning

Over the past few decades, deep learning has emerged as a popular machine learning technique for solving real-world problems. Deep learning architectures, modeled after the structure of the human brain, process data through stages of transformation and representation. These architectures have been highly successful in various domains, including object recognition, and this success can be extended to CBIR to help overcome the semantic gap. Deep learning algorithms, such as CNNs, Deep Neural Networks (DNNs), Deep Belief Networks, and Boltzmann Machines, are particularly powerful in computer vision tasks. Among these, CNNs have gained significant popularity in CBIR [42].

CNNs consist of three types of layers: convolutional, pooling, and fully connected layers. In the convolutional layers, filters are applied to input images to learn important features. Pooling layers, or subsampling layers, reduce the spatial dimensions of the input, which helps decrease the amount of information and computation required. Finally, the fully connected layer predicts the class or label of the input image. The key difference between CNNs and ANNs is that CNNs use convolutional and pooling layers before the fully connected layer, while in ANNs, all neurons are interconnected [43].

CNNs have several advantages over ANNs. They are robust to translation, scaling, and rotation, making them ideal for computer vision tasks. Additionally, CNNs do not require hand-crafted feature extraction, as they are capable of learning relevant features directly from the data. However, CNNs do rely on labeled datasets, which can be a limitation in certain cases [44].

Table 3 provides an overview of machine learning categories, their main characteristics, and examples. Deep learning, with CNNs at its core, has proven to be a successful approach in CBIR due to its feature extraction capabilities and its efficiency in computer vision tasks.

Wan [45] explored the behavior of CNNs in various case studies within the CBIR field, aiming to improve image retrieval by using CNNs to represent image features and measure similarity. The results demonstrated that CNNs effectively extract features, leading to enhanced retrieval performance. However, it is important to acknowledge that combining a large visual dictionary with CNNs can introduce several challenges. Specifically, the increased size of the visual dictionary can impact memory storage and training time, potentially degrading retrieval capacity. These limitations highlight the need to consider practical and computational requirements when employing CNN-based approaches in CBIR systems.

Alzu'bi et al. [46] introduced a novel CBIR system utilizing bilinear CNNs. This innovative approach used CNNs for unsupervised feature extraction from image content, without the need for bounding boxes, annotations, or class labels. To optimize memory and computational costs, the extracted features underwent dimensionality reduction via pooling during the feature extraction process. The system was tested on large-scale image retrieval tasks, showing promising results. By leveraging the bilinear CNN's ability to capture and represent visual features, the system provided accurate retrieval results. The use of unsupervised feature extraction and dimensionality reduction demonstrated the system's capability to handle large-scale image datasets efficiently.

Tzelepi and Tefas [47] proposed a method to enhance CBIR performance by employing a CNN to represent features. They modified the classical CNN architecture by incorporating maximum pooling after convolutional layers, instead of using fully connected layers. This modification was intended to preserve spatial information, which is often lost with fully connected layers that connect to all input neurons. By reducing the dimensionality of the feature descriptor while retaining essential spatial information, their approach achieved high retrieval efficiency with minimal memory and processing time requirements. The method presented three different schemes based on the available information: fully unsupervised retraining, utilization of relevance information, and feedback-based retraining.

In the completely unsupervised retraining approach, there is only a dataset available, allowing the CNN to be trained without any class labels. The relevance information scheme is activated when a labeled dataset is available, enabling the use of class labels to enhance training. Finally, the relevance feedback-based retraining approach involves user interaction, where users provide feedback on the retrieved results, followed by iterative retraining of the CNN based on this feedback. These approaches cover various scenarios, providing flexibility to the proposed CBIR method, accommodating different levels of data availability and user involvement. Overall, this approach enhances retrieval performance by balancing computational efficiency, memory usage, and the use of available information sources.

Zheng et al. [48] proposed a VGGNet-based end-to-end CBIR system. Instead of using traditional class labels for CNN training, the authors used a gravitational field dataset with similarity score labels. The system was evaluated on three benchmark datasets—Oxford, Paris, Holidays, and Caltech 101—achieving high accuracies of 0.9620, 0.9410, and 0.8850, respectively. These results demonstrate the system's promising potential for accurate image retrieval.

However, constructing the gravitational field database used to train the CNN was found to be time-consuming. The authors also acknowledged that the system's speed should be improved during both the training and testing stages. Despite these limitations, the proposed end-to-end CBIR system shows strong potential for providing accurate image retrieval results. Future improvements will focus on reducing the time required to build the gravitational field database and enhancing the system's training and testing efficiency.

Sezavar et al. [49] proposed a CBIR framework using a convolutional neural network (CNN) for high-level feature extraction. They employed the last layer of the AlexNet architecture, originally introduced by Krizhevsky et al. [50]. The last layer was chosen because it produces the smallest feature vector, thereby reducing computational cost. To further optimize the process, the authors applied sparse representation, an effective compression technique that enhances retrieval performance while maintaining an acceptable level of accuracy. The proposed approach was evaluated on three datasets: ALOI, Corel, and MPEG-7. The results demonstrated good retrieval speed and accuracy, highlighting the efficiency of the method.

However, it is worth noting that while sparse representation enables faster bulk access, it comes at the expense of slightly reduced accuracy compared to other methods. Their paper also includes Table 4, which provides a detailed overview of the performance of machine learning algorithm-based approaches in CBIR, offering valuable insights into the comparative speed and accuracy of different methods.

In conclusion, the integration of machine learning algorithms into various stages of CBIR can significantly improve retrieval accuracy. However, the training and testing phases of these algorithms often require substantial processing time. While machine learning enhances accuracy, it also introduces computational challenges that must be addressed for CBIR systems to be efficient. Further research into optimization techniques and hardware advancements is necessary to reduce processing time and make machine learning-based CBIR more practical and scalable.

4. RESEARCH GAP

From the reviewed studies, several research gaps can be identified in the field of CBIR:

- **Feature Combination and Fusion**: Using a single type of feature for image representation often results in suboptimal retrieval performance. Therefore, research is needed on efficient methods to merge and combine different types of features, such as color, texture, and shape, to produce more comprehensive and discriminative representations. The challenge lies in finding appropriate fusion methods that can leverage interactions among different features, even when they vary in dimensionality and properties.
- **Database-Specific Fusion Approaches**: While feature fusion often improves results, most fusion approaches are typically tailored to specific image databases or domains. There is a research gap in developing generic fusion approaches that can be adapted to different types of image databases, encompassing various visual content, image sizes, and domain-specific characteristics. Moreover, these approaches must maintain retrieval performance when applied to diverse datasets

without being hindered by the challenges posed by different image collections.

- **Dimensionality Reduction for High-Dimensional Features**: Techniques such as local patterns, keypoints, and deep learning methods (e.g., CNNs) have shown promise in improving retrieval precision. However, they often produce high-dimensional feature vectors, leading to increased computational complexity and storage requirements. Developing dimensionality reduction techniques specifically designed for high-dimensional features is critical to enhancing the scalability of CBIR systems.

- **Integration of Machine Learning and CBIR**: Although machine learning algorithms have been utilized in CBIR tasks such as feature selection, visual vocabulary construction, and image classification, there is a gap in developing efficient approaches that seamlessly integrate machine learning into the CBIR pipeline. This includes exploring new ways to co-optimize tasks like feature extraction and classification and harnessing the full potential of machine learning to improve retrieval performance and address the semantic gap.

- **Scalability for Large-Scale Databases**: Many current CBIR approaches perform well on small databases but degrade in performance when applied to large-scale databases due to increased computational and storage demands. Research is needed to develop scalable CBIR approaches that maintain high retrieval precision and efficiency without significantly increasing execution time, even with large image databases.

Addressing these research gaps will contribute to advancements in CBIR, improving retrieval performance, scalability, and efficiency, ultimately leading to more practical and effective image retrieval systems.

5. CONCLUSION

This comprehensive review of CBIR systems, with a focus on the use of machine learning techniques, represents a significant step forward in advancing the field. The examination of various feature extraction methods, both global and local, highlights their crucial role in enhancing the precision and effectiveness of CBIR systems. The research shows that while global features provide a quick overview of image content, local features offer more detailed and accurate representations, which are essential for complex retrieval tasks. The integration of machine learning algorithms—encompassing supervised, unsupervised, and deep learning techniques like CNNs—has been transformative for CBIR. These algorithms have significantly improved image retrieval accuracy, addressed the semantic gap, and facilitated the handling of high-dimensional data. Deep learning, in particular, has revolutionized the field by providing unparalleled feature learning and adaptability across diverse image datasets.

However, the review also identifies critical research gaps and challenges. These include the need for more efficient feature fusion methods, scalable solutions for large image databases, and the application of machine learning to enhance semantic interpretation in CBIR. Additional challenges arise from the high computational demands and the need for large, labeled datasets, particularly in deep learning.

In conclusion, while significant progress has been made in integrating machine learning into CBIR systems, there is still considerable potential for further advancements. Addressing these research gaps will not only improve existing systems but also lay the foundation for more sophisticated, effective, and accurate image retrieval solutions. The future of CBIR is closely tied to the continuous evolution of machine learning techniques and their successful integration with image retrieval processes, promising new breakthroughs in digital image and data analysis.

REFERENCES

[1] Alrahhal, M., & Supreethi, K. P. (2024). Integrating machine learning algorithms for robust content-based image retrieval. International Journal of Information Technology. https://doi.org/10.1007/s41870-024-02169-2

[2] Al-Qazzaz, N., Alrahhal, M., Jaafer, S., Ali, S., & Ahmad, S. (2023). Automatic diagnosis of epileptic seizures using entropy-based features and multimodal deep learning approaches. Medical Engineering and Physics. https://doi.org/10.1016/j.medengphy.2024.104206

[3] Alrahhal, M., & Supreethi, K. P. (2024). Enhancing image retrieval accuracy through multi-resolution HSV-LNP feature fusion and modified K-NN relevance feedback. International Journal of Information Technology. https://doi.org/10.1007/s41870-024-02000-y

[4] Alrahhal, M., & Supreethi, K. P. (2021). Covid-19 diagnostic system using medical image classification and retrieval: A novel method for image analysis. The Computer Journal, 65(8), 2146–2163. https://doi.org/10.1093/comjnl/bxab051

[5] Alrahhal, M., & Rahhal, M. (2021). Detection of COVID-19 disease using artificial intelligence techniques and radiographic medical images. Journal of King Abdulaziz University Computing and Information Technology Sciences, 10(1), 73–96. https://doi.org/10.4197/comp.10-1.4

[6] Alrahhal, M., & Supreethi, K. P. (2021). Full Direction Local Neighbors Pattern (FDLNP): A novel method for multimedia image retrieval system using ensemble classifiers on distributed framework. International Journal of Advanced Computer Science and Applications, 12(1). https://doi.org/10.14569/ijacsa.2021.0120116

[7] Alrahhal, M., & Supreethi, K. P. (2020). Multimedia image retrieval system by combining CNN with handcraft features in three different similarity measures. International Journal of Computer Vision and Image Processing, 10(1), 1–23. https://doi.org/10.4018/ijcvip.2020010101

[8] Alrahhal, M., & Supreethi, K. P. (2019). Content-based image retrieval using local patterns and supervised machine learning techniques. In 2019 Amity International Conference on Artificial Intelligence (AICAI). https://doi.org/10.1109/aicai.2019.8701255

[9] Shrivastava, N., & Tyagi, V. (2015). An efficient technique for retrieval of color images in large databases. *Computers & Electrical Engineering, 46*, 314–327. https://doi.org/10.1016/j.compeleceng.2014.11.009

[10] Younus, Z. S., Mohamad, D., Saba, T., Alkawaz, H. M., Rehman, A., Al-Rodhaan, M., & Al-Dhelaan, A. (2015). Content-based image retrieval using PSO and k-means clustering algorithm. *Arabian Journal of Geosciences, 8*(8), 6211–6224. https://doi.org/10.1007/s12517-014-1584-7

[11] Ponomarev, A., Nalamwar, H. S., Babakov, I., Parkhi, C. S., & Buddhawar, G. (2016). Content-based image retrieval using color, texture, and shape features. *Key Engineering Materials, 685*, 872–876. https://doi.org/10.4028/www.scientific.net/KEM.685.872

[12] Srivastava, P., & Khare, A. (2017). Integration of wavelet transform, local binary patterns, and moments for content-based image retrieval. *Journal of Visual Communication and Image Representation, 42*, 78–103. https://doi.org/10.1016/j.jvcir.2016.11.008

[13] Sajjad, M., Ullah, A., Ahmad, J., Abbas, N., Rho, S., & Baik, S. W. (2018). Integrating salient colors with rotational invariant texture features for image representation in retrieval systems. *Multimedia Tools and Applications, 77*(4), 4769–4789. https://doi.org/10.1007/s11042-017-5010-5

[14] Zheng, X., Tang, B., Gao, Z., Liu, E., & Luo, W. (2016). Study on image retrieval based on image texture and color statistical projection. *Neurocomputing, 215*, 217–224. https://doi.org/10.1016/j.neucom.2015.07.157

[15] Zhao, Z., Tian, Q., Sun, H., Jin, X., & Guo, J. (2016). Content-based image retrieval scheme using color, texture, and shape features. *International Journal of Signal Processing, Image Processing and Pattern Recognition, 9*(1), 203–212. https://doi.org/10.14257/ijsip.2016.9.1.19

[16] Phadikar, B. S., Phadikar, A., & Maity, G. K. (2018). Content-based image retrieval in DCT compressed domain with MPEG-7 edge descriptor and genetic algorithm. *Pattern Analysis and Applications, 21*(2), 469–489. https://doi.org/10.1007/s10044-016-0589-0

[17] Pavithra, L. K., & Sharmila, T. S. (2018). An efficient framework for image retrieval using color, texture, and edge features. *Computers & Electrical Engineering, 70*, 580–593. https://doi.org/10.1016/j.compeleceng.2017.08.030

[18] Pavithra, L. K., & Sree Sharmila, T. (2019). An efficient seed points selection approach in dominant color descriptors (DCD). *Cluster Computing, 22*(4), 1225–1240. https://doi.org/10.1007/s10586-019-02907-3

[19] Ashraf, R., Ahmed, M., Jabbar, S., Khalid, S., Ahmad, A., Din, S., & Jeon, G. (2018). Content-based image retrieval by using color descriptor and discrete wavelet transform. *Journal of Medical Systems, 42*(3), 44. https://doi.org/10.1007/s10916-017-0880-7

[20] Tadi Bani, N., & Fekri-Ershad, S. (2019). Content-based image retrieval based on combination of texture and color information extracted in spatial and frequency domains. *The Electronic Library, 37*(4), 650–666. https://doi.org/10.1108/EL-03-2019-0067

[21] Rana, S. P., Dey, M., & Siarry, P. (2019). Boosting content-based image retrieval performance through integration of parametric and nonparametric approaches. *Journal of Visual Communication and Image Representation, 58*(3), 205–219. https://doi.org/10.1016/j.jvcir.2018.11.015

[22] Thusnavis Bella, M. I., & Vasuki, A. (2019). An efficient image retrieval framework using fused information features. *Computers & Electrical Engineering, 75*, 46–60. https://doi.org/10.1016/j.compeleceng.2019.01.022

[23] Ashraf, R., Ahmed, M., Ahmad, U., Habib, M. A., Jabbar, S., & Naseer, K. (2020). MDCBIR-MF: Multimedia data for content-based image retrieval by using multiple features. *Multimedia Tools and Applications, 79*(13–14), 8553–8579. https://doi.org/10.1007/s11042-018-5961-1

[24] Alsmadi, M. K. (2020). Content-based image retrieval using color, shape, and texture descriptors and features. *Arabian Journal for Science and Engineering, 45*(4), 3317–3330. https://doi.org/10.1007/s13369-020-04384-y

[25] Low, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision, 60*(2), 91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94

[26] Montazer, G. A., & Giveki, D. (2015). Content-based image retrieval system using clustered scale-invariant feature transforms. *Optik, 126*(18), 1695–1699. https://doi.org/10.1016/j.ijleo.2015.05.002

[27] Sharif, U., Mehmood, Z., Mahmood, T., Javid, M. A., Rehman, A., & Saba, T. (2019). Scene analysis and search using local features and support vector machine for effective content-based image retrieval. *Artificial Intelligence Review, 52*(2), 901–925. https://doi.org/10.1007/s10462-018-9636-0

[28] Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). Speeded-up robust features (SURF). *Computer Vision and Image Understanding, 110*(3), 346–359. https://doi.org/10.1016/j.cviu.2007.09.014

[29] Jabeen, S., Mehmood, Z., Mahmood, T., Saba, T., Rehman, A., Mahmood, M. T., & Rubin, D. L. (2018). An effective content-based image retrieval technique for image visuals representation based on the bag-of-visual-words model. *PLoS One, 13*(4), e0194526. https://doi.org/10.1371/journal.pone.0194526

[30] Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (Vol. 1, pp. 886–893). San Diego, CA. https://doi.org/10.1109/CVPR.2005.177

[31] Mehmood, Z., Abbas, F., Mahmood, T., Javid, M. A., Rehman, A., & Nawaz, T. (2018). Content-based image retrieval based on visual words fusion versus features fusion of local and global features. *Arabian Journal for Science and Engineering, 43*(12), 7265–7284. https://doi.org/10.1007/s13369-018-3062-0

[32] Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(7), 971–987. https://doi.org/10.1109/TPAMI.2002.1017623

[33] Guo, Z., Zhang, L., & Zhang, D. (2010). Rotation invariant texture classification using LBP variance (LBPV) with global matching. *Pattern Recognition, 43*(3), 706–719. https://doi.org/10.1016/j.patcog.2009.08.017

[34] Sarwar, A., Mehmood, Z., Saba, T., Qazi, K. A., Adnan, A., & Jamal, H. (2019). A novel method for content-based image retrieval to improve the effectiveness of the bag-of-words model using a support vector machine. *Journal of Information Science, 45*(1), 117–135. https://doi.org/10.1177/0165551518782825

[35] Yousuf, M., Mehmood, Z., Habib, H. A., Mahmood, T., Saba, T., Rehman, A., & Rashid, M. (2018). A novel technique based on visual words fusion analysis of sparse features for effective content-based image retrieval. *Mathematical Problems in Engineering, 2018*, 1–13. https://doi.org/10.1155/2018/2134395

[36] Ashraf, R., Bashir, K., Irtaza, A., & Mahmood, M. (2015). Content-based image retrieval using embedded neural networks with bandletized regions. *Entropy, 17*(6), 3552–3580. https://doi.org/10.3390/e17063552

[37] Wan, J. (2014). Deep learning for content-based image retrieval. In *Proceedings of the ACM International Conference on Multimedia - MM '14* (pp. 157–166). New York, USA.

[38] Alzu'bi, A., Amira, A., & Ramzan, N. (2017). Content-based image retrieval with compact deep convolutional features. *Neurocomputing, 249*, 95–105. https://doi.org/10.1016/j.neucom.2017.03.072

[39] Tzelepi, M., & Tefas, A. (2018). Deep convolutional learning for content-based image retrieval. *Neurocomputing, 275*, 2467–2478. https://doi.org/10.1016/j.neucom.2017.11.022

[40] Zheng, Q., Tian, X., Yang, M., & Wang, H. (2019). Differential learning: A powerful tool for interactive content-based image retrieval. *Engineering Letters, 27*(1), 202–215. http://www.engineeringletters.com/issues_v27/issue_1/EL_27_1_23.pdf

[41] Sezavar, A., Farsi, H., & Mohamadzadeh, S. (2019). Content-based image retrieval by combining convolutional neural networks and sparse representation. *Multimedia Tools and Applications, 78*(15), 20895–20912. https://doi.org/10.1007/s11042-019-7321-1

[42] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM, 60*(6), 84–90. https://doi.org/10.1145/3065386