

Deep Feature-based Image Retrieval via CNN and Autoencoder with Grey Wolf Optimized K-means Clustering

Ali Mohsin Ahmed Al-Sabaawi ^{1,*}, Balqees Talal Hasan ², Zaid J. Al-Araji ³,
and Mohsin Hasan Hussein ^{4,5}

¹ Software Department, College of Information Technology, Ninevah University, Mosul, Iraq

² Artificial Intelligence Department, College of Information Technology, Ninevah University, Mosul, Iraq

³ Department of Computer Networks and Internet, College of Information Technology, Ninevah University, Mosul, Iraq

⁴ Department of Computer Science, College of Computer Science and Information Technology, University of Kerbala, Kerbala, Iraq

⁵ College of Science, University of Warith Al-Anbiyaa, Kerbala, Iraq

Email: ali.mohsin@uoninevah.edu.iq (A.M.A.A.S.); balqees.hasan@uoninevah.edu.iq (B.T.H.);

zaid.jasim@uoninevah.edu.iq (Z.J.A.A.); mohsin.h@uokerbala.edu.iq (M.H.H.)

*Corresponding author

Abstract—An essential criterion in content-based image retrieval is its accuracy. High accuracy depends heavily on the quality of feature extraction and dimension reduction methods. To achieve this aim, this paper integrates Visual Geometry Group (VGG) 16-based feature extraction with Autoencoder-based dimensionality reduction, along with Grey Wolf Optimization (GWO), to enhance the K-means clustering algorithm. Initially, 4096 deep features were extracted from images using VGG16. These features were then reduced to 10, 15, and 30 latent features using an Autoencoder (AE) that captures complex data structures more effectively than linear approaches. Moreover, the clustering stage was further improved by applying GWO to promote the initialization of K-means centroids. The proposed method was assessed on two datasets: Corel-1k (10 classes) and UC Merced Land Use (UCM) (21 classes). Further, it was compared with the Principal Component Analysis (PCA)-based method and other recent deep learning approaches. The experimental results revealed that the proposed method outperformed other studies in both image retrieval accuracy by mean average precision, and clustering quality, assessed by Purity, Adjusted Rand Index (ARI), and Normalized Mutual Information (NMI). These results confirm that feature extraction using nonlinear dimension reduction by AE yields more discriminative features than PCA and other approaches in all metrics.

Keywords—Content-Based Image Retrieval (CBIR), deep feature extraction, Visual Geometry Group (VGG) 16, autoencoder-based dimensionality reduction, grey wolf optimizer, K-means clustering, deep learning

I. INTRODUCTION

With the advent of the Smartphone era, the internet and rapid digitization, visual content has been growing in

volume at an exponential pace in recent years. This has brought high profile to the problem of image retrieval. With large-scale repositories, it becomes increasingly difficult to identify relevant images, especially when the query requires more than a simple keyword to describe the visual semantics. Traditional hand-annotation and text-based indexing are still popular, but they are labor-intensive, subjective, and hard to scale. Furthermore, textual metadata is not necessarily accurate when it comes to grasping fine-grained visual information, causing mismatches in retrieval. These constraints prompt strategies that exploit the content of the images for more accurate and efficient retrieval. To overcome these drawbacks, Content-Based Image Retrieval (CBIR) was proposed to organize and index images based on their inherent visual properties. The paradigm became more solid in the mid-to-late 1990s, with the initial wave of research on content-based image retrieval [1]. Conventional text-based pipelines (e.g., bag-of-words representations, keyword-centric engines such as Lycos' multimedia search and Google Images) are constrained by the semantic gap between the terms the user provides and the image's visual content [2]. To address this mismatch, the proposed system builds an image search framework based on features directly extracted from images, which allows for the retrieval of images according to their visual similarity rather than using textual proxies.

Most recent applications include, but are not limited to, AI-based text mining [3], speech recognition [4], recommendation systems [5], medical image classification [6, 7], and target detection [8]. Artificial Intelligence (AI) in general and Deep Learning (DL) in particular have gained wide adoption in various fields, including Content-based Image Retrieval (CBIR) [1].

Deep Learning algorithms enable the extraction of important features process during training. These features can be captured more effectively by Deep Learning algorithms than by traditional methods [9]. Among the most important and efficient techniques in the area of computer vision for feature extraction from digital images is Convolutional Neural Networks (CNNs) [10].

Within the domain of CBIR, Deep Learning techniques have been widely adopted for feature extraction. Although many CBIR techniques rely on linear dimensionality reduction methods such as Principal Component Analysis (PCA) or classical clustering methods. PCA may fail to capture complex nonlinear relationships within the feature space. While, classical clustering methods are inherently limited by the choice of initial centroids. These limitations become evident when dealing with complex datasets containing high intra-class variability and semantic diversity, making the clustering process very difficult. The underlying nonlinear structures in most real-world datasets are beyond the scope of linear projection techniques, and many of the clustering methods are rather ineffective, therefore yielding clusters that are inconsistent, and extremely poor. Hence, there is a need to explore and expand the use of non-linear feature reduction methods and more advanced clustering methods, and to evaluate their performance on datasets with a high degree of complexity.

Despite the progress achieved in image retrieval over the past few years, legacy systems still suffer from several evident limitations, leading to poor overall system performance that could be further improved. This study proposes a hybrid multi-step framework to address several gaps in current CBIR systems. The framework incorporates the use of deep learning for feature extraction, the use of nonlinear data compression to provide the data with the most critical features, and Grey Wolf Optimization (GWO)-Optimized K-means clustering technique to enhance clustering robustness and retrieval performance. It addresses gaps in linear dimensionality reduction and in standard approaches to clustering. More specifically, the focus of this technique is on improving the clustering process, aiming to create denser, more robust, and stable clusters with semantic cohesion, and to improve the process's ability to reflect the nonlinear structure of the data. With this explanation in mind, the following research questions were formulated.

RQ1: Is it possible for Autoencoders to achieve nonlinear dimensionality reduction, and thus eliminate some of the issues associated with the use of PCA, which may result in less accurate retrieval of images, as a consequence of their inability to address problems associated with complex data effectively?

RQ2: Which feature dimensions are more capable of accurately representing images and facilitating the clustering process to achieve higher retrieval accuracy?

RQ3: Can the K-means clustering algorithm be improved when integrated with the GWO?

RQ4: Will the proposed method effective when evaluated on datasets with varying levels of complexity?

RQ5: How does the proposed method compare with the results of other studies that employ approaches similar to the proposed framework?

Based on the limitations identified in previous studies and the research questions formulated in this work, the proposed method employs nonlinear dimensionality reduction and an optimized clustering approach to improve the retrieval of images with varying levels of complexity. The main contributions of this work are summarized as follows:

- The proposed approach does not just simply combine the current approaches, but is a holistic approach that addresses the various drawbacks of the CBIR systems, such as the high-dimensional deep features, non-linear feature representation, and the sensitivity to the initialization of the centroids.
- The study investigates the effect of varying latent feature dimensions (10, 15, and 30) on retrieval accuracy and clustering performance. The experimental results demonstrate that as the visual complexity of the data increases, nonlinear Autoencoder embeddings become more effective by providing greater representational capacity.
- The proposed framework is extensively evaluated on two benchmark datasets with different levels of complexity, namely Corel-1K and UC-Merced. The results demonstrate its robustness and scalability across datasets containing both simple and complex image distributions.
- A comprehensive comparative evaluation is conducted against PCA-based dimensionality reduction approaches as well as several recent deep learning-based CBIR methods employing conventional and fuzzy clustering techniques. The proposed framework achieves competitive and superior retrieval performance in terms of mean Average Precision (mAP).

II. LITERATURE REVIEW

Several studies have been conducted to improve the performance of CBIR systems using deep learning, dimensionality reduction, and clustering techniques. This section provides a brief review of most related approaches, somewhat focusing on feature extraction approaches, nonlinear representation learning, and optimized clustering techniques.

Ugale and Railkar [1] proposed a CBIR approach via combining texture and shape features to produce the Multiscale Adaptive Local Ternary Pattern (MALTP) with the Residual Network with 50 layers (ResNet50) hybrid model. Retrieval of images from a database using Convolutional Neural Networks (CNNs) has gained popularity due to their multi-level embedding features. CNN has been shown to be highly effective in Computer-Based Image Retrieval (CBIR) applications in image recognition. This study obtained accurate results on two of the datasets used. Merely based on comparison of the results, Babenko *et al.* [11] attempted to classify three variants of CNN networks (D-Leaf, a fine-tuned AlexNet, and a pre-trained AlexNet) in the scope of feature

extraction and five techniques of machine learning classification.

Among the methods, the fine-tuned AlexNet achieved the best retrieval accuracy of 95.54%. Krizhevsky *et al.* [12] showed that activations from a fully connected layer were used as global feature descriptors for image retrieval and achieved excellent results on the ImageNet dataset. Following that, Lin *et al.* [13] proposed a learning framework that utilizes digit-like binary coding to streamline and accelerate image searches, highlighting the usefulness of CNNs for image feature extraction. Desai *et al.* [14] introduced a rapid image retrieval process that integrates a CNN and a Support Vector Machine (SVM) in a unified flow. In their process, VGG16 is employed as the feature extractor, architected with 12 convolutional layers followed by 4 fully connected layers and an end softmax layer. The SVM, on the other hand, performs classification to accelerate the return of results. On the Corel dataset, the method achieved an average precision of 0.8361 for the top-10 retrieved images. Ranjith *et al.* [15] proposed Deep Learning-based Inception v3 Model (DLIM), an inception-based deep learning framework for content-based image retrieval that learns discriminative image representations and computes similarity between query and database samples within the learned feature space. In parallel, Sudhish *et al.* [16] developed a content-based medical image retrieval pipeline that couples a CNN feature extractor with index clustering to organize feature maps. Their system demonstrated high precision in MRI brain image retrieval, thereby supporting radiological assessment and diagnostic decision-making for brain tumors. Ghaleb *et al.* [17] integrated CNN and the Gate Recurrent Unit (GRU) for feature extraction to retrieve images more accurately. The study failed to remove noise and redundant information, which, as a result, negatively affected. Hassan *et al.* [18] utilized a CNN for feature extraction and to bridge the semantic gap between low-level pixel data and human perception. The study focused on selecting the appropriate architecture of the CNN and the best hyperparameters.

It can be seen that utilizing CNN-based methods significantly improve semantic feature extraction. However, many existing approaches rely solely on deep features without addressing feature redundancy, dimensionality complexity, or clustering stability.

For Content-Based Medical Image Retrieval (CBMIR), effective image indexing is important for evaluating accuracy, retrieval speed, scalability, and memory footprint. This is particularly true for high-dimensional feature spaces that are extracted by CNNs. It is widely known that feature vectors are often sparse and are greatly impacted by the ‘curse of dimensionality’ [19, 20]. Therefore, this research implements dimensionality-reduction strategies to compress the representation space. This helps to alleviate the effects of sparsity, and improves the efficiency of indexing and retrieval performance. In this area, two of the most common methods for linear dimensionality reduction are PCA and classical Multidimensional Scaling (MDS). Both of these approaches rely on the eigen-decomposition of either the

covariance or the Gram matrix and formulate the problem in matrix form to identify low-dimensional embeddings. Of these methods, PCA is one of the most widely used methods for dimensionality reduction [21]. While PCA is quite popular, there are some drawbacks to conventional PCA in that it is performed only on vectorized data, and is only capable of reducing the dimensionality of a single data feature vector [22]. To meet this requirement, high-order feature-map tensors must be flattened to a high degree, which can result in a loss of the encoded spatial and channel relationships. This can negatively impact the features’ performance in terms of fidelity and discrimination. In contrast, an AE based on convolution operates by learning, in a nonlinear manner, a low-dimensional embedding directly from feature maps by preserving locality and channel interrelationships. The encoder may be subjected to regularization-denoising, sparsity, or contraction, while also aligning to retrieval via metric objectives to produce codes that are more discriminative and tolerant of noise [23].

From reviewing the literature, it can be seen that Autoencoders (AEs) reduce dimensionality by minimizing reconstruction error. This means they optimize for feature fidelity rather than for the substantial similarity necessary for image retrieval. Thus, AEs often have latent representations that preserve nuisance factors while down-weighting the discriminative cues [24]. They confer weak invariances without using labels or metric/contrastive terms. Thus, they usually underperform supervised or contrastive representations on retrieval benchmarks [25]. Wangi and Makandar [26] introduced the use of AE-based deep learning models for image feature extraction. Their work involved three variants, namely, a simple AE, a deep AE, and a convolutional AE. In contrast to the present study, their approach employed the AE directly as the feature extractor rather than for dimensionality reduction. Moreover, applying similarity across all extracted features can increase the computational complexity. Even though Autoencoders are quite effective for nonlinear dimensionality reduction, many existing methods prioritize reconstruction fidelity over discriminative feature learning for retrieval tasks. This can actually make retrieval weaker in complex datasets.

Initially, high-level semantically rich descriptors are derived using a VGG16 pre-trained on ImageNet, as these CNN features are known to transfer well to retrieval and recognition tasks with limited resources [27]. Then, a convolutional AE is applied to compress these descriptors nonlinearly while preserving the spatial/channel structure in the latent code. This way, the semantics defined by VGG16 is used, and the AE is considered a structure-preserving, task-aligned compressor. This reduces space and query time without compromise accuracy [23]. In other words, the AE method will force the learning of low-level features, often being more sensitive to reconstruction criteria than to semantic separation. The fact that the representation is anchored in VGG16 and the AE is used only for dimension reduction prevents reconstruction bias and yields compact codes that are still informative at the semantic level for retrieval.

After dimensionality reduction, the similarity measure becomes central to ranking database images against the query. However, a naïve exhaustive search over the entire collection is time- and memory-intensive and can dilute precision by comparing the query with many weakly related items. A more effective strategy is to partition the feature space so that visually similar images are co-located and dissimilar ones are separated. Retrieval can then be restricted to the most relevant group(s), thereby reducing search costs and often improving accuracy through clustering [27].

Clustering condenses a dataset into multiple groups (clusters) comprised of instances with similar characteristics and segregates instances with differing characteristics [28]. Clustering techniques have been adopted in several studies in various fields [29–33]. Among these techniques, K-means clustering is particularly popular, as it is simple and efficient. However, K-means is highly dependent on the initialization of centroids, and many clustering-based CBIR techniques still use random initialization. As a result, this has led to unstable clustering outcomes. K-means can also achieve poor accuracy and converge more slowly. The cluster ceases to move towards the global optimum, and falls into local optima. Therefore, some optimization methods are well-suited to improving the quality of the clustering solution and its stability. Within these boundaries, some metaheuristic methods, such as Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO), are used to improve the initialization process of K-means clustering [34].

Kumar *et al.* [35] introduced a Vision Transformer Hashing with Supervised Contrastive learning (VTHSC) framework to facilitate medical image retrieval. With this approach, the retrieval system is built on a pre-trained Vision Transformer backbone and includes supervised contrastive learning and deep hashing. Zhu *et al.* [36] proposed a Cross-modal Contrastive Learning framework (CCLS2T) for content-based remote sensing image retrieval. In this approach, retrieval accuracy is improved and operational efficiency is achieved through

spatiotemporal visual information and hashing-based indexing within asymmetric dual-encoder frameworks. In reference [37], the authors proposed a lightweight model that combined a CNN with a Transformer-based architecture to capture image features at both local and global levels.

Recent CBIR studies have examined Contrastive Learning and Vision Transformers create new architectures. They demonstrate the ability to capture global information and features at a larger scale through robust representation learning. However, these approaches usually require large, complex datasets and long training times. From the reviewed studies, a number of these limitations can be recognized in CBIR systems with respect to preserving nonlinear feature representations, the robustness of the clusters, and the limited validation of proposed methods on datasets with diverse characteristics and complexity levels. Therefore, there is a need for more effective frameworks with advanced robust feature representation and clustering optimization.

III. MATERIALS AND METHODS

In this research, the dataset is randomly split into 90% training and 10% testing. During the training phase, the VGG16 model, pre-trained on the ImageNet dataset, is used to extract deep features. Because of the large size of the features, a convolutional (AE) is used to extract a smaller latent representation of the deep features while maintaining the local structures and nonlinear relationships. The K-means clustering algorithm is used to cluster the latent representations. The initialization and optimization of the K-means centroids are enhanced using the GWO. During the testing phase, query images are taken from the testing dataset and undergo the same feature extraction and dimensionality reduction steps. Organized latent spaces facilitate the image retrieval process. Finally, a similarity measure is computed between the query feature vector and the cluster centers to identify the most relevant cluster, from which the top-n most similar images are retrieved. Fig. 1 shows the proposed method.

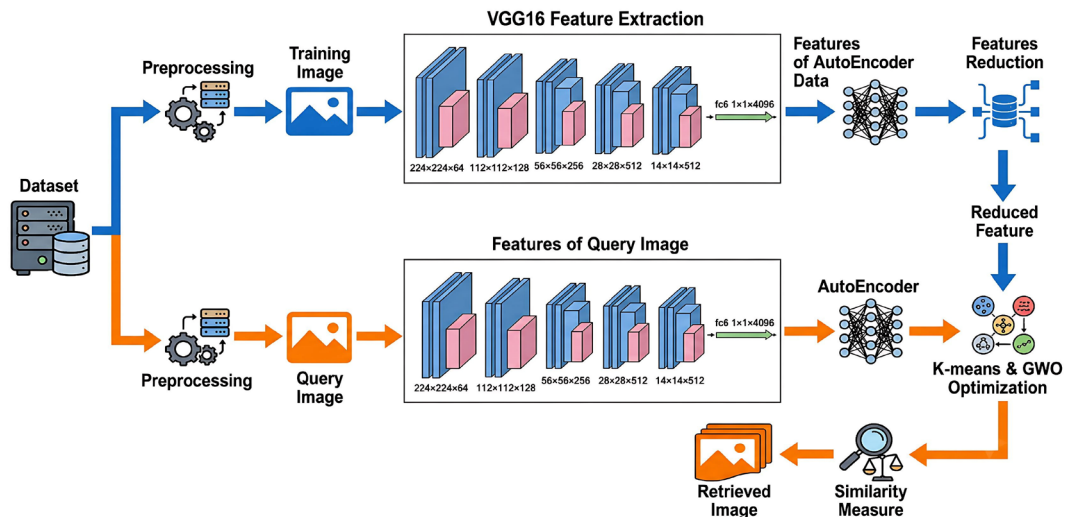


Fig. 1. The proposed method.

A. Feature Extraction

Feature extraction is performed using the VGG16 model proposed in Ref. [27]. VGG16 has gained broad acceptance owing to its simple architecture, depth, and strong transfer learning capabilities. These properties have made it a facto backbone in computer vision tasks like classification, detection, or feature extraction. It has sixteen weight layers (thirteen convolutional and three fully connected) organized in an extremely simple and uniform manner. All convolutional layers use small 3×3 kernels with a stride of 1 and proper padding to preserve spatial resolution, interleaved with max-pooling layers of size 2×2 for down-sampling purposes. The later layers of the network demonstrate a gradual increase in the depth of feature maps, i.e., $64 \rightarrow 128 \rightarrow 256 \rightarrow 512$, while stepwise decreasing in spatial resolution. Finally, the predictions are made using fully connected layers at the top, each with 4096 features.

B. Dimension Reduction

Several methods have been used to reduce the feature dimensionality. One method is Contrastive learning. Contrastive learning methods have recently achieved strong results in representation learning by enhancing matching between different augmented versions of the same instance. Moreover, according to current research, Vision Transformers and other transformer-based models have achieved peak performance on image understanding tasks. The methods require both extensive datasets and substantial processing power to function effectively [38, 39]. In contrast, the AE in this study delivers an efficient dimensionality reduction method that operates without supervision to support clustering-based retrieval. The AE demonstrates its value as a practical solution for the tested datasets because of its ability to balance performance requirements with computational needs and is still an appropriate choice for dimension reduction use [40, 41]. In this study, the AE is used to reduce dimensionality. The input from the feature extraction is 4096. It feeds the AE, producing three output features (10, 15, and 30). AEs are beneficial in that their latent spaces can model non-linear manifolds that capture the curved geometry of complex data distributions. In contrast to the flattening-based approaches, convolutional AEs are aware of the spatial structure since they operate directly on the feature maps (height $\times$ width $\times$ channels), leading to more faithful tensor-aware compression. Furthermore, an AE allows denoising, sparsity, or contractive penalties as regularization constraints to encourage the learning of robust and compact codes. Fig. 2 shows the details of the AE architecture used to reduce dimensionality.

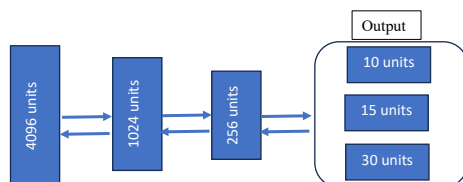


Fig. 2. Autoencoder architecture.

From the previous image, it can be seen that during the training phase, the input is 4096 features. Then, it is compressed into the first layer and later reduced to 256 units before reaching the last output (10, 15 and 30). In reverse, the decoder starts from 10, 15 and 30 features to reconstruct the original features (4096). The algorithm of the AE architecture is presented below.

Algorithm 1. AE-Based Dimensionality Reduction

Input:

Dataset $X \in \mathbb{R}^{(N \times 4096)}$
Latent dimension $m \in \{10, 15, 30\}$

Output:

Reduced feature representation Z

Begin

1. Normalize the dataset X using feature scaling.
2. Split X into training and validation subsets.
3. Construct the AE architecture:

Encoder:

- a. Dense layer (1024 neurons, Rectified Linear Unit (ReLU))
- b. Dropout layer (rate = 0.1)
- c. Dense layer (256 neurons, ReLU)
- d. Latent layer (m neurons, Linear)

Decoder:

- a. Dense layer (256 neurons, ReLU)
- b. Dropout layer (rate = 0.1)
- c. Dense layer (1024 neurons, ReLU)
- d. Output layer (4096 neurons, Linear)
4. Configure the training process:
 - Loss function: Mean Squared Error (MSE)
 - Optimizer: Adam (learning rate = 0.001)
 - Batch size: 128
 - Maximum epochs: 200
 - Early stopping patience: 15
5. Train the AE using:
 - Training data: (X_{train} , Y_{train})
 - Validation data: (X_{val} , Y_{val})
6. After training, retain only the Encoder component.
7. Transform the original dataset:
 - $Z = \text{Encoder}(X)$

End

C. Clustering K-means with GWO

K-Means is among the most widely used unsupervised learning algorithms for addressing clustering tasks. However, its performance is highly dependent on the initial selection of cluster centroids. Random initialization often results in reduced accuracy and a tendency to converge to local optima [32]. To overcome this limitation, various optimization techniques have recently been employed to improve the robustness and effectiveness of K-Means clustering. In this study, the GWO algorithm is employed to initialize the centroids of K-means.

GWO is a nature-inspired meta-heuristic algorithm based on the leadership hierarchy and hunting strategy of grey wolves. The leading wolves (α, β, δ) guide the search towards the global optimum, while follower wolves (ω) enhance local exploitation. This balance between exploration and exploitation helps avoid premature convergence to local optima [42]. The following steps indicate the implementation of K-means with GWO to find the optimal centroids and cluster the data. Algorithm 2

summarizes the integration of the Grey Wolf Optimizer (GWO) with K-means clustering for optimized centroid initialization.

Algorithm 2. The Integration of K-Means with GWO

Input:

Reduced feature dataset Z
 Number of clusters K
 Maximum iterations T
 Population size M

Output:

Optimized cluster centroids C^*
 Final clustering assignments

Begin

- 1- load and preprocess the reduced feature dataset Z .
- 2- Define the K-means objective function using Eq. (1).

$$J(C) = \sum_{i=1}^k \sum_{x_i \in S_j} \|x_i - c_j\|^2 \quad (1)$$

where S_j is the set of data points assigned to cluster j , c_j is the centroid of cluster j , $\|x_i - c_j\|^2$ is the distance between the data point and the centroid. k represents the number of clusters.

- 3- Initialize a population of wolves using Eq. (2).

$$x_m = [c_{m,1,1}, \dots, c_{m,1,D}, \dots, c_{m,k,D}] \quad (2)$$

where x_m represents the position of the m -th wolf, $c_{m,1,1}$ is the d -th feature of the j -th centroid for the m -th wolf.

- 4- Evaluate the fitness of each wolf using the K-means objective function.
- 5- Identify the best three wolves:
 - α (best solution)
 - β (second-best solution)
 - δ (third-best solution)
- 6- For each iteration $t = 1$ to T :

A- Compute the distance vectors using Eqs. (3)–(5):

$$D_\alpha = |C_1 \times x_\alpha - X| \quad (3)$$

$$D_\beta = |C_2 \times x_\beta - X| \quad (4)$$

$$D_\delta = |C_3 \times x_\delta - X| \quad (5)$$

B- Update wolf positions using Eqs. (6)–(8):

$$x1 = X_\alpha - A1 \times D_\alpha \quad (6)$$

$$x2 = X_\beta - A2 \times D_\beta \quad (7)$$

$$x3 = X_\delta - A3 \times D_\delta \quad (8)$$

C- Update coefficient vectors using Eqs. (9)–(10):

$$A_i = 2a \times r1 - a \quad (9)$$

$$C_i = 2 \times r \quad (10)$$

where X_i refers to the position vector of the wolves alpha, Beta and Delta, D_i indicates the distance of the wolves to the prey, A_i and C_i are the coefficient vectors, a linearly decreases from 2 to 0 over iterations, $r1$ and $r2$ random vectors in $[0,1]$.

D- Update wolf position

$$X(t+1) = \frac{(x1+x2+x3)}{3} \quad (11)$$

E- Recalculate fitness values

F- Update α , β , and δ wolves.

- 7- Use the optimized centroid positions obtained by GWO as the initial centroids for K-means clustering.
- 8- Perform final K-means clustering using the optimized centroids.

End

D. Similarity and Retrieved Images

In this work, the similarity between feature vectors is measured using the Pearson Correlation Coefficient. This metric captures the linear relationship between feature vectors, making it particularly suitable for deep feature representations where the relative patterns of activations are more informative than their absolute values. Moreover, Pearson correlation is invariant to scale and offset, which enhances its robustness when comparing high-dimensional and complex feature spaces.

The query image is initially processed by the pre-trained VGG16 model to extract deep feature representations. Subsequently, the extracted features are compressed using the AE. The resulting query feature vector is then compared with the cluster centroid feature vectors generated by the GWO-optimized K-means algorithm using the Pearson Correlation Coefficient as shown in Eq. (12). Based on the highest similarity score, the query image is assigned to the most relevant cluster. Thereafter, the retrieval process is restricted to the images within the selected cluster, and the similarity is computed again between the query feature vector and the feature vectors of the relevant cluster. Finally, the retrieved images are ranked in descending order according to their similarity scores, and the top- N most similar images are returned as the retrieval results.

$$\text{sim}(img, q) = \frac{\sum_{i=1}^n (img_i - \overline{Av_img_i})(q_i - \overline{Av_q})}{\sqrt{\sum_{i=1}^n (img_i - \overline{Av_img_i})^2} \sqrt{\sum_{h=1}^n (q_i - \overline{Av_q})^2}} \quad (12)$$

where, img_i is the feature of the image, Av_img_i represents the mean of the image's features. $\overline{Av_q}$ refers to the average of the query's features. n is the number of features. In this method, the similarity value between img and q ranges from $[-1, 1]$. The higher value means that the two users are more similar.

IV. RESULT AND DISCUSSION

A. Datasets Description

In this study, two benchmark datasets are utilized: Corel-1K [43] and UC Merced Land Use (UCM) [44]. The Corel-1K dataset consists of 1000 images organized into 10 categories, each containing 100 images either 256×384 or 384×256 resolution. The UCM dataset includes 2100 aerial images divided into 21 classes, with 100 images per class. Each image has a fixed resolution of 256×256 pixels. Table I displays the summary of the datasets.

TABLE I. THE SUMMARY OF THE DATASETS

Dataset	Number of Classes	Images per Class	Image Resolution
Corel-1K	10	100	256×384 or 384×256
UCM	21	100	256×256

B. Evaluation Metrics

The effectiveness of the CBIR system is measured using Precision, Average Precision (AP), and mAP. Precision

quantifies the proportion of relevant images among the retrieved results; AP represents the average precision for a single query; and mAP corresponds to the mean AP across all queries or clusters. The following equations represent the calculation of these metrics.

$$p = \frac{\text{number of relevant retrieved images}}{\text{total number of the retrieved images}} \quad (13)$$

$$Ap = \frac{1}{n} \sum_{i=1}^n p_i \quad (14)$$

$$mAp = \frac{1}{m} \sum_{k=1}^m Ap_k \quad (15)$$

where Ap_k refers to the AP of class k , n is the number of classes, m is the number of implementations.

In addition, the hyperparameters adopted in this study are demonstrated in Tables II and III.

TABLE II. THE HYPERPARAMETERS OF DEEP LEARNING METHODS

Methods	Batch Size	Epochs	Learning Rate	Optimization
VGG16	256	74	0.01	Stochastic Gradient Descent
AE	128	200	0.001	Adam

TABLE III. THE HYPERPARAMETERS OF GWO

No. of Wolves	Max Iteration
50	70

V. RESULTS AND DISCUSSION

As described earlier, VGG16 was employed to extract image features, followed by an AE to reduce dimensionality. The dataset was split 80% for training and 20% for testing for model development and evaluation. Three feature dimensions were considered in the study: 10, 15, and 30. Finally, K-Means combined with GWO was applied to cluster the images. Table IV presents the average precision results for these three dimensions, with precision reported for each class at 10, 15, and 30 dimensions. The last row in the table shows the mean average precision of the retrieval performance across the dataset, demonstrating the effectiveness of the proposed feature representation in capturing image details.

TABLE IV. THE MAP OF THE PROPOSED METHOD USING THE COREL-1K

Classes	Dim = 10	Dim = 15	Dim = 30
Africa	0.95	0.95	0.99
Beaches	0.92	0.97	0.97
Building	0.96	1	1
Bus	1	1	0.99
Dinosaur	1	1	1
Elephant	1	1	1
Flowers	1	0.96	0.96
Horses	1	1	1
Mountains	0.97	0.99	1
Food	0.93	0.97	1
mAP	0.973	0.984	0.99

As shown in Table V, the mAP results are highest when the feature dimension is set to 30. Although the performance of the second and fourteenth classes is slightly lower than that obtained with 15 dimensions, this does not imply that 15 dimensions are superior to 30. The difference is attributed to the clustering process for these specific classes, where some points may have been misclustered. Nevertheless, the results for the remaining classes clearly confirm the significant superiority of the 30-dimensional representation.

TABLE V. THE MAP OF THE PROPOSED METHOD USING THE UC-MERCED

Classes	Dim = 10	Dim = 15	Dim = 30
Agriculture	0.98	0.96	1
Airplane	0.72	1	0.96
Baseball	0.71	0.98	0.97
Beach	0.95	0.92	1
Building	0.94	0.81	0.89
Chaparral	1	0.95	0.99
Denseresidential	0.76	1	1
Forest	0.93	1	1
Freeway	0.76	1	0.99
Golfcourse	0.72	1	1
Harbor	0.87	1	1
Intersection	1	1	1
Mediumresidential	1	1	1
Mobilehomepark	0.99	1	0.91
Overpass	0.97	1	1
Parkinglot	0.79	0.79	1
River	0.93	1	1
Runway	0.92	0.9	1
Sparseresidential	0.98	0.99	0.98
Storgetanks	0.85	0.73	0.94
Tenniscourt	0.98	0.94	0.92
mAP	0.894	0.95	0.98

A. Ablation Study

The performance of the proposed approach is evaluated across multiple system configurations, including PCA and AE for dimensionality reduction, as well as standard K-means and GWO-optimized K-means for clustering, to assess their impact on overall system performance. The purpose of this section is to validate the design of the proposed method, which employs an AE as a nonlinear transformation that captures complex data structures, whereas PCA, as a linear technique, may fail to preserve such relationships. Furthermore, integrating GWO to optimize the clustering process is expected to enhance clustering quality and improve retrieval performance. Table VI and Table VII show the results of the proposed method (AE with optimized clustering) and Principal Component Analysis (PCA) with the optimized clustering method.

Although several classes achieved near-perfect precision values, such as Beaches, Buildings, Bus, Elephant, and Dinosaur, this behavior can be partially attributed to the characteristics of benchmark datasets such as Corel-1K, where some classes exhibit highly distinctive visual characteristics and low intra-class variation. Consequently, the deep feature embeddings extracted from the VGG16 network can effectively distinguish these categories, yielding very high retrieval accuracy.

TABLE VI. COMPARISON OF THE PERFORMANCE OF PCA WITH GWO CLUSTERING, AE WITH K-MEANS VS AE WITH GWO CLUSTERING OF THE UCM DATASET

Classes	PCA with GWO Clustering	AE with K-means	AE with GWO Clustering
	Dim = 30	Dim = 30	Dim = 30
Agriculture	0.96	1	1
Airplane	1	0.98	0.96
Baseball	0.97	0.99	0.97
Beach	0.99	1	1
Building	0.96	0.94	0.93
Chaparral	0.955	0.99	0.99
Denserresidential	0.96	0.95	1
Forest	1	1	1
Freeway	0.95	0.97	0.99
Golfcourse	0.96	0.92	1
Harbor	0.98	0.96	1
Intersection	0.95	0.94	1
Mediumresidential	0.95	0.97	1
Mobilehomepark	0.95	0.94	0.91
Overpass	0.97	0.97	1
Parkinglot	1	0.98	1
River	0.95	0.90	1
Runway	0.94	0.99	1
Sparseresidential	0.97	0.99	0.98
Storgetanks	0.99	0.91	0.94
Tenniscourt	0.97	0.97	0.92
<i>mAP</i>	0.96	0.964	0.98

TABLE VII. COMPARISON OF THE PERFORMANCE OF PCA WITH GWO CLUSTERING, AE WITH K-MEANS VS AE WITH GWO CLUSTERING OF THE COREL-1K DATASET

Classes	PCA with Optimized Clustering	AE with K-Means	AE with Optimized Clustering
	Dim = 30	Dim = 30	Dim = 30
Africa	0.94	0.88	0.99
Beaches	0.95	0.95	0.97
Building	0.98	0.99	1
Bus	1	0.97	0.99
Dinosaur	1	1	1
Elephant	1	1	1
Flowers	1	0.98	0.96
Horses	1	0.98	1
Mountains	0.97	0.99	1
Food	0.92	1	1
<i>mAP</i>	0.975	0.977	0.99

Nevertheless, exceptionally high performance may raise concerns about potential overfitting and limited generalization. To address this issue, the proposed framework was additionally evaluated on the more challenging UCM Dataset, which contains higher inter-class similarity and greater scene variability. The experimental results demonstrate that the proposed method maintains strong performance across datasets with varying complexity, indicating that the framework is not specifically tailored to a single dataset and demonstrates consistent performance across the evaluated datasets.

This suggests that the observed performance enhancement is not only to dataset simplicity, but also reflects the robustness of the proposed feature representation and the clustering approach.

As shown in Table VI, classes such as Beaches, Buildings, Bus, Elephant, and Dinosaur produced relatively similar results for both the proposed AE-based method and the PCA-based approach. However, more noticeable performance differences were observed in

visually complex classes such as Africa and Food. A similar trend can also be observed in Table VII. Overall, the reported mAP values demonstrate the superiority of the proposed AE-based framework over PCA combined with optimized clustering. In Table VI, some classes such as Beaches, Buildings, Bus, Elephant, and Dinosaur, yield relatively close results with the two methods. Yet, noticeable differences are observed in other classes, including Africa and Food. The same scenario is shown in Table VII. Overall, the mAP highlights the superiority of the proposed AE-based method over PCA with optimized clustering. For extra visualization, Fig. 3 and Fig. 4 show the results of the mAP of the UCM and Corel-1k datasets, respectively.

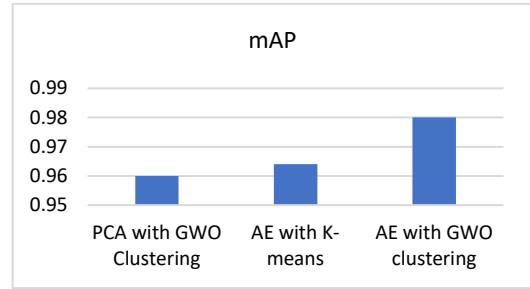


Fig. 3. The mAP results of the UCM dataset.

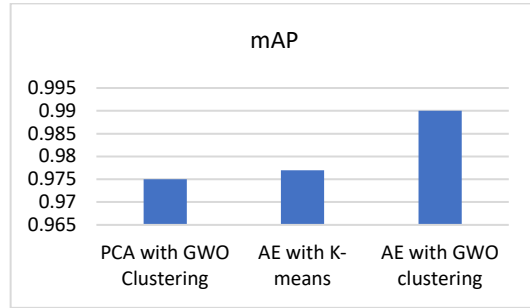


Fig. 4. The mAP results of the Corel-1K dataset.

B. Benchmarking with Previous Methods

In this subsection, the proposed method is benchmarked against recent deep learning-based approaches for CBIR. The proposed method is comparatively evaluated against several recent studies using standard datasets with consistent settings. Among these studies, Desai *et al.* [14] developed a hybrid framework that uses a pre-trained VGG16 network with a Support Vector Machine (SVM) for classification. Ranjith *et al.* [15] proposed DLIM, an Inception-based deep learning framework for Content-Based Image Retrieval (CBIR) that learns image features for discrimination and assesses similarity between queries and database examples within the feature space. Ghaleb *et al.* [17] examined three deep learning variants, and the CNN with GRU variant was reported to perform best. In the same way, Hassan *et al.* [18] applied CNN-based feature extraction with optimized hyperparameters and Euclidean distance as the measure of similarity. Shen *et al.* [29] combined contextual, hand-crafted and deep features, followed by Fuzzy C-Means clustering for retrieval and dimensionality

reduction. Furthermore, Tyagi *et al.* [45] proposed a hybrid model combination hand-crafted feature (color and texture) with deep learning features, resulting in a significant improvement in precision.

Theoretically, VGG16 offers high-level representations that deep spatial and contextual information often neglected by conventional descriptors. AEs can retain complex relationships and preserve more of the original data structure than linear PCA. K-means clustering minimizes the variance within clusters, but is highly sensitive to how the initial centroid is selected. This limitation is addressed through the incorporation of the GWO, which enhances the global search and reduces the potential for falling into local optima.

Given the results obtained on the Corel-1K and UC-Merced datasets using the mAP metric, as reported in Tables VIII and IX, some authors prefer to report results in terms of precision; as such, careful evaluation is required for a direct numerical comparison of systems. The mAP provides a more comprehensive picture of the evaluated system's performance by assessing the quality of the ranking and retrieval accuracy.

As illustrated in Table VIII, many studies exhibit robust results using the Corel-1K dataset. Tyagi *et al.* [45] reports a precision of 0.985, one of the highest results in the literature, and Shen *et al.* [29], which uses a hybrid feature clustering framework, reports a precision of 0.971. The latter is most relevant to the current study, as it

incorporated feature extraction, clustering, framework integration, and dimensionality reduction. Even with differences in assessment dimensions, the proposed approach achieves competitive results and is superior to others in mAP and overall retrieval effectiveness. The results also demonstrate that hybrid deep learning methods, such as those by Ranjith *et al.* [15] and Ghaleb *et al.* [17], outperform techniques based on a single feature-extraction approach, such as those by Desai *et al.* [14] and Hassan *et al.* [18]. This indicates the critical nature of combining many steps in the process, including feature extraction, feature transformation, and clustering, as in the proposed approach.

Table IX represents the assessment results for the UC-Merced dataset, which has fewer related studies. The proposed method achieves a precision of 0.98, compared to 0.95 for the ResNet50 model. This demonstrates the benefit of the two-stage feature representation approach, which incorporates VGG16 and an AE for dimensionality reduction. When hybrid methods, such as MALTP + ResNet50, are used, the performance is equivalent when extra hand-crafted features (such as texture and shape) are integrated. This indicates that integrating various features improves discrimination capability. The referred work is also worth mentioning, as it was evaluated on only 20 classes, whereas this work evaluated the complete dataset. This extensive evaluation exemplifies the scalability and robustness of the proposed framework.

TABLE VIII. PERFORMANCE COMPARISON WITH PREVIOUS METHODS IN TERMS OF MAP IN THE COREL-1K DATASET

Classes	Ref. [17]	Ref. [14]	Ref. [15]	Ref. [18]	Ref. [29]	Ref. [45]	Proposed Method
No. of retrieved images	10	10		10		10	10
Africa	0.9	0.84	0.87	0.91	-	-	0.99
Beaches	0.96	0.84	0.934	0.65	-	-	0.97
Buildings	0.94	0.835	0.844	0.95	-	-	1
Bus	1	0.827	0.913	0.90	-	-	0.99
Dinosaurs	1	0.83	0.932	1	-	-	1
Elephants	0.86	0.838	0.90	0.75	-	-	1
Flowers	0.94	0.83	0.917	1	-	-	0.96
Horses	0.86	0.84	0.92	0.99	-	-	1
Mountains	0.9	0.838	0.89	1	-	-	1
Food	0.945	0.837	0.90	0.65	-	-	1
mAP	0.93	0.836	0.903	0.88	0.971	0.985	0.99

TABLE IX. PERFORMANCE COMPARISON WITH PREVIOUS METHODS IN TERMS OF MAP IN THE UC-MERCED DATASET

Metric	RESNet50 [1]	MALTP + RESNet50 [1]	Proposed Method Dim = 30
mAP	0.95	0.98	0.98

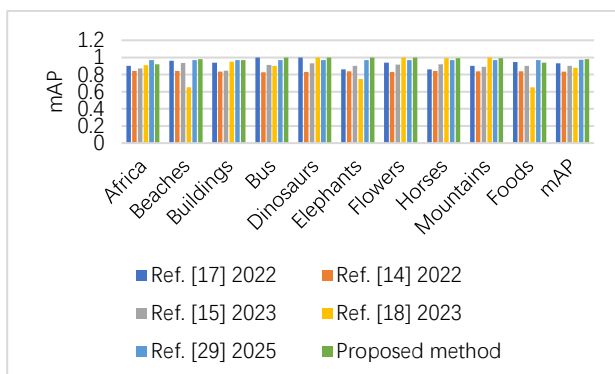


Fig. 5. The performance comparison of the Corel-1k dataset.

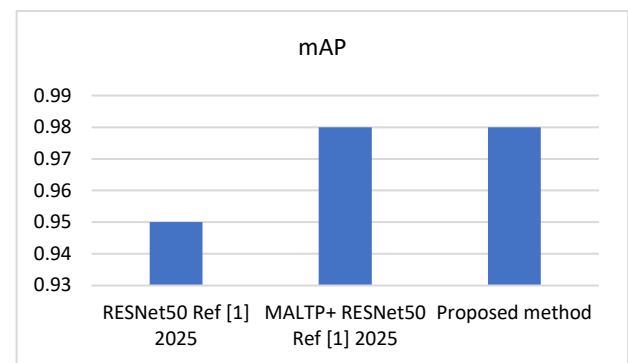


Fig. 6. The performance comparison of the UC-Merced dataset.

Using the UCM and Corel-1k datasets, Figs. 5 and 6 depict the results of the proposed method and previous approaches, studies showing the precision for each class across all the compared studies, including the proposed model. Sharp variation among individual class results is evident. Overall, the results reveal that the proposed method outperforms the compared studies in the majority of the cases, reflecting its ability to capture more discriminative feature representations than the compared approaches. Furthermore, as illustrated in Fig. 5, Shen *et al.* [29] presents results most comparable to the current work. This indicates that the incorporation of deep feature extraction and dimensionality reduction together with clustering significantly enhances feature discrimination compared to methods relying solely on deep features.

C. Clustering Evaluation

In this subsection, the clustering accuracy of image grouping is evaluated. The datasets consist of 10 and 21 predefined classes, for Corel-1K and UC-Merced respectively. Using the selected feature representations with dimensionality 10, 15, and 30, the proposed method successfully partitioned the dataset into 10 and 21 clusters using an optimized K-means algorithm. Three different metrics are used in this evaluation, namely, Purity, Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI). Purity is a measure of how well each cluster is classified with respect to a single ground-truth class. The values range from 0 to 1. 1 is the perfect value. The second metric is ARI. It compares the assigned cluster of samples with the truth table. The values of this metric range from -1 and 1 ; 1 is the perfect value. The last metric is NMI. It measures how well one can assess the utility of clusters. The range is between 0 and 1. Eqs. (16)–(19) present the formulations used to compute these metrics.

$$purity = \frac{1}{N} \sum_{i=1}^k \max_j |c_i \cap t_j| \quad (16)$$

where N indicates the number of data points, k refers to the number of clusters, c_i is a cluster in C , and t_j denotes the classification with the total number for the class c_i .

$$ARI = \frac{RI - \epsilon[RI]}{Max(RI) - \epsilon[RI]} \quad (17)$$

where $RI = a + b/\binom{n}{2}$, a indicates the number of pairs in the same cluster and true table, b refers to the number of pairs of different clusters and different classes, and $\binom{n}{2}$ is the total number of pairs.

$$NMI(U, V) = \frac{2 \times (U; V)}{H(U) + H(V)} \quad (18)$$

$$I(U; V) = \sum_{u \in U} \sum_{v \in V} p(u, v) \log \frac{p(u, v)}{p(u)p(v)} \quad (19)$$

$H(u)$ and $H(v)$ indicate the Entropy of cluster assignment and true labels.

Tables X and XI present comparison between the proposed method and a baseline approach that employs PCA instead of the AE. The evaluation is conducted across three feature dimensions: 10, 15, and 30.

As shown in Tables X and XI, clustering quality improves with increasing latent dimensionality in the case of AE (from 10 to 30 features), as nonlinear encoders preserve complex structures within greater representational capacity. Thus, AE consistently yields higher Purity, ARI, and NMI scores. PCA showed almost the same results using 10, 15, and 30 features, since the first principal components were sufficient to capture the essence of the variances and the subsequent components failed to add much new information pertaining to the important factor clustering. Findings show that dimension reduction through AE yields a more accurate approach compared to PCA-based reductions reported in the literature.

TABLE X. RESULTS OF COMPARISON BETWEEN THE PROPOSED METHOD AND BASE METHOD IN THE COREL-1K DATASET

Feature Dimensions	VGG16, AE and K-means with GWO			VGG16, PCA and K-means with GWO		
	Purity	ARI	NMI	Purity	ARI	NMI
10 features	0.977	0.954	0.959	0.959	0.912	0.918
15 features	0.979	0.955	0.95	0.958	0.91	0.917
30 features	0.984	0.965	0.964	0.959	0.912	0.918

TABLE XI. RESULTS OF COMPARISON BETWEEN THE PROPOSED METHOD AND BASE METHOD IN THE UC-MERCE DATASET

Feature Dimensions	VGG16, AE and K-means with GWO			VGG16, PCA and K-means with GWO		
	Purity	ARI	NMI	Purity	ARI	NMI
10 features	0.858	0.725	0.846	0.788	0.632	0.784
15 features	0.943	0.888	0.927	0.820	0.684	0.804
30 features	0.96	0.92	0.94	0.837	0.707	0.81

The study presents evidence of multiple factors that affect the accuracy and performance of image retrieval systems through the implementation of dimensionality reduction methods and clustering techniques. The PCA method produced identical clustering results across different dimensions because linear dimensionality reduction reaches its maximum capacity, after which it cannot handle higher-dimensional data accurately.

However, the AE-based features exhibit performance differences with dimensionality, enabling them to handle more intricate nonlinear patterns as their representational capacity increases. The impact of optimized clustering is more pronounced in datasets with multiple classes because feature nonlinearity interacts with the clustering initialization to yield different results. The results demonstrate that nonlinear dimensionality reduction

together with clustering optimization functions as an integrated system, which provides greater advantages even when it is used with more complex datasets.

D. Complexity Discussion

In this subsection, the computational complexity and execution time of the proposed framework are evaluated across different feature dimensions and compared with those of the corresponding model without Autoencoder-based dimensionality reduction.

The computational complexity of the proposed VGG16-AE-GWO-K-means approach can be divided into two main components. The first component corresponds to the complexity associated with the VGG16-AE stage. The complexity of VGG16 for all convolution layers represented by $O(\sum_{l=1}^{13} H_l W_l K^2 C_l^{in} C_l^{out})$, where H , W are the spatial dimensions of the feature map, K is the size of the kernel, C_{in} and C_{out} are the numbers of input channels and the number of output filters, respectively. The computational complexity of the AE can be expressed as $O(N \cdot d \cdot m)$.

In the context of image retrieval, the bottleneck (latent) dimension d and the input dimension N are the most significant factors, simplifying the complexity to

$O(N \cdot d \cdot m)$, where N refers to the size of the feature vector you are feeding into the AE, d is the size of the compressed representation at the center of the AE, and m is the number of images. Accordingly, the overall complexity of the VGG16-AE stage can be written as Eq. (20):

$$O(\sum_{l=1}^{13} H_l \times W_l \times K^2 \times C_l^{in} \times C_l^{out}) + o(N \times d \times m) \quad (20)$$

The second part includes the complexity of GWO-Kmeans. The computational complexity for GWO-Kmeans is dominated by the GWO algorithm. At each iteration in GWO, the fitness value for each particle is evaluated depending on N samples, and K clusters with D dimensions. Thus, the total complexity of GWO-Kmeans is $O(T \times P \times N \times K \times D)$, where T represents the GWO iterations, P represents the population size, N is the number of samples in the dataset, K represents the number of clusters, and D stands for the dimensions of the features.

Regarding computational efficiency, Table XII presents a comparison of the execution times of the proposed VGG16-AE-GWO-K-means framework under different feature dimensions.

TABLE XII. COMPUTATIONAL TIME (S) COMPARISON BETWEEN THE PROPOSED VGG16-AE-GWO-K-MEANS FRAMEWORK AND THE CORRESPONDING MODEL WITHOUT AE-BASED DIMENSIONALITY REDUCTION

#Features Datasets	10 Features	15 Features	30 Features	All Features without AEs
Corel 1K (1000 samples)	267	268.2	271.2	2559.51
UCM Dataset (2100 samples)	732	735	752.4	2633.20

As shown in Table XII, the Corel-1K dataset, which contains 1000 images distributed across 10 classes, required approximately 267 s, 268.2 s, and 271.2 s when the feature dimensions were reduced to 10, 15, and 30, respectively. In contrast, the execution time of the proposed framework without applying the Autoencoder (AE) was significantly higher, reaching approximately 2559 s.

Similarly, for the UCM Dataset, which contains 2100 images across 21 classes, the proposed framework required approximately 732 s, 735 s, and 752.4 s for feature dimensions of 10, 15, and 30, respectively. Conversely, the execution time without employing the Autoencoder reached approximately 2633 s.

These results clearly demonstrate that incorporating the Autoencoder significantly improves the computational efficiency of the proposed framework by reducing feature dimensionality and accelerating clustering and retrieval.

VI. CONCLUSIONS

The proposed retrieval framework is based on the VGG16 architecture. The model leverages the VGG16 neural network's ability to transform images into feature vectors and uses an Autoencoder with three latent dimensions to compress the vectors at 10, 15, and 30 dimensions. The framework employs the GWO, combined with K-means, to improve clustering by generating better

initial centroids and exploring the solution space, thereby reducing the effects of local optima.

Theoretically, K-means Clustering is sensitive to centroid initialization and often settles on a suboptimal solution. The merging of the Grey Wolf Optimizer with K-means Clustering enables the clustering mechanism to search a broader solution space and yields stable, semantically interpretable clusters. The combination of the local search capability of the K-means Clustering and the global search capability of the GWO is a pillar of the improvement in clustering quality, as verified by the experimental results.

The reduced feature representations were subsequently employed for image retrieval by identifying the most similar images to a query image within each cluster using the Pearson Correlation distance measure. The effectiveness of the proposed framework was evaluated using the Corel-1K and UCM datasets in terms of mAP. The experimental findings demonstrated that the VGG16-AE-GWO-K-means model with 30 latent features outperformed the VGG16-PCA-GWO-K-means model, achieving an mAP of 99% compared with 97.5%.

For clustering evaluation, three metrics were employed: Purity, Adjusted Rand Index (ARI), and Normalized Mutual Information (NMI). The proposed method achieved values of 98%, 96.5%, and 96.4% for Purity, ARI, and NMI, respectively, compared with 95.9%, 91.2%, and 91.8% obtained by the VGG16-PCA-GWO-K-means model on the Corel-1K dataset. Overall, the

obtained results indicate that combining the pre-trained VGG16 deep neural network for feature extraction with the hybrid Grey Wolf Optimizer–K-means clustering approach significantly improves Content-Based Image Retrieval performance across different image categories when compared with existing approaches.

Additionally, AE-based method outperformed other methods, especially PCA, in the sense that it could capture the complex, nonlinear structures in higher class datasets. The obtained results validate the fact that the nonlinear dimensionality reduction of AEs can extract more discriminative features, which enhances the image retrieval and clustering performance. But dimension reduction with AE incurs an additional computational overhead over linear methods.

For a more advanced future quest, other optimization techniques can be adopted such as Whale optimization algorithm and Ant colony optimization to improve clustering quality. Also, alternative dimensionality reduction approaches may be investigated to enhance feature representation quality and the reliability of the image retrieval process.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

A. M. A. Al-Sabaawi conducted the methodology, software of clustering and deep learning, original draft preparation. B. T. Hasan conducted the visualization statistical test, methodology, Optimization software. Zaid J. Al-Araji wrote the review, editing and analysis. M. H. Hussein contributed to conceptualization, validation, analysis and supervision. All authors had approved the final version.

ACKNOWLEDGMENT

Thanks to Ninevah University for supporting this work.

REFERENCES

- [1] P. Ugale and P. Railkar, "Content-based remote sensing image retrieval based on two-way feature representation using ResNet50 and modified multiscale local ternary pattern," *Math. Model. Eng. Probl.*, vol. 12, no. 9, pp. 2969–2980, 2025.
- [2] Y. Liu, D. Zhang, G. Lu *et al.*, "A survey of content-based image retrieval with high-level semantics," *Pattern Recogn.*, vol. 40, no. 1, pp. 262–282, 2007.
- [3] E. E. Blanchard, B. Oner, A. Allgood *et al.*, "Evolution of simulation scholarship: A text mining exploration," *Clin. Simul. Nurs.*, vol. 96, 101620, 2024.
- [4] X. Lian, N. Xia, G. Dai *et al.*, "An efficient joint training model for monaural noisy-reverberant speech recognition," *Appl. Acoust.*, vol. 228, 110322.
- [5] I. A. Al-Mani, A. M. A. Al-Sabaawi, and M. H. Hussien, "A review paper of model based collaborative filtering techniques," in *Proc. Int. Conf. Data Sci. Intell. Comput.*, 2022, pp. 52–57.
- [6] M. A. Alkahya, H. O. Alreahan, and Z. Y. Algamal, "Classification of breast cancer histopathological images using adaptive penalized logistic regression with Wilcoxon rank sum test," *Electron. J. Appl. Stat. Anal.*, vol. 16, no. 3, pp. 507–518, 2023.
- [7] X. Sun and H. Qian, "Chinese herbal medicine image recognition and retrieval by convolutional neural network," *PLOS ONE*, vol. 11, no. 6, e0156327, 2016.
- [8] A. N. Mon, W. P. Pa, and Y. K. Thu, "Exploring the effect of tones for Myanmar language speech recognition using convolutional neural network," in *Proc. Comput. Linguist.*, 2018, pp. 314–326.
- [9] Y. Yang, Q. J. Wu, and Y. Wang, "Autoencoder with invertible functions for dimension reduction and image reconstruction," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 48, no. 7, pp. 1065–1079, 2016.
- [10] M. S. Ghaleb, H. M. Ebied, H. A. Shedeed *et al.*, "Content-based image retrieval based on convolutional neural networks," in *Proc. 10th Int. Conf. Intell. Comput. Inf. Syst.*, 2021, pp. 149–153.
- [11] A. Babenko, A. Slesarev, A. Chigorin *et al.*, "Neural codes for image retrieval," in *Proc. Comput. Vis.*, 2014, pp. 584–599.
- [12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [13] K. Lin, H. F. Yang, J. H. Hsiao *et al.*, "Deep learning of binary hash codes for fast image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2015, pp. 27–35.
- [14] P. Desai, J. Pujari, C. Sujatha *et al.*, "Hybrid approach for content-based image retrieval using VGG16 layered architecture and SVM: An application of deep learning," *SN Comput. Sci.*, vol. 2, 170, 2021.
- [15] E. Ranjith, L. Parthiban, T. P. Latchoumi *et al.*, "An effective content-based image retrieval system using deep learning based inception model," *Wireless Pers. Commun.*, vol. 133, pp. 811–829, 2023.
- [16] D. K. Sudhish, L. R. Nair, and S. Shailesh, "Content-based image retrieval for medical diagnosis using fuzzy clustering and deep learning," *Biomed. Signal Process. Control*, vol. 88, 105620, 2024.
- [17] M. S. Ghaleb, H. M. Ebied, H. A. Shedeed *et al.*, "Image retrieval based on deep learning," *J. Syst. Manag. Sci.*, vol. 12, no. 2, pp. 477–496, 2022.
- [18] R. Q. Hassan, Z. N. Sultani, and B. N. Dhannoon, "Content-based image retrieval based on corel dataset using deep learning," *Int. J. Artif. Intell.*, vol. 12, no. 4, pp. 1854–1863, 2023.
- [19] L. Zhuo, B. Cheng, and J. Zhang, "A comparative study of dimensionality reduction methods for large-scale image retrieval," *Neurocomputing*, vol. 141, pp. 202–210, 2014.
- [20] I. M. Hameed, S. H. Abdulhussain, and B. M. Mahmmod, "Content-based image retrieval: A review of recent trends," *Cogent Eng.*, vol. 8, no. 1, 2021.
- [21] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemom. Intell. Lab. Syst.*, vol. 2, no. 1–3, pp. 37–52, 1987.
- [22] I. Jolliffe, "Principal Component Analysis," in *Encyclopedia of Statistics in Behavioral Science*, 2nd ed. New York, NY, USA: Springer, B. S. Everitt and D. C. Howell, Eds., Hoboken: Wiley, 2005.
- [23] S. Rifai, P. Vincent, X. Muller *et al.*, "Contractive auto-encoders: Explicit invariance during feature extraction," in *Proc. 28th Int. Conf. Mach. Learn.*, 2011, pp. 833–840.
- [24] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, 2013.
- [25] T. Chen, S. Kornblith, M. Norouzi *et al.*, "A simple framework for contrastive learning of visual representations," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.
- [26] K. Wangi and A. Makandar, "Autoencoder for image retrieval system using deep learning technique with TensorFlow and Keras," in *Proc. IEEE Int. Conf. Integr. Circuits Commun. Syst.*, 2023, pp. 1–5.
- [27] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint, arXiv:1409.1556, 2014.
- [28] L. M. Abualigah, A. T. Khader, and E. S. Hanandeh, "A new feature selection method to improve document clustering using particle swarm optimization," *J. Comput. Sci.*, vol. 25, no. 1, pp. 456–466, 2018.
- [29] Z. Y. Shen, H. D. Shen, and F. He, "A hybrid deep learning-based model for enhanced feature representation in image retrieval systems," *Egypt. Inform. J.*, vol. 31, 100717, 2025.
- [30] A. M. A. Al-Sabaawi, H. Karacan, and Y. E. Yenice, "SVD++ and clustering approaches to alleviating the cold-start problem for recommendation systems," *Int. J. Innov. Comput. Inf. Control*, vol. 17, no. 2, pp. 383–396, 2021.
- [31] N. T. Mohammed, M. H. Hussein, and A. J. Rashid, "PAM clustering aided Android malicious apps detection," in *Proc. IOP Conf. Ser. Mater. Sci. Eng.*, 2020, pp. 1–9.

- [32] M. H. Hussein, H. N. Nawaf, and W. S. Bhaya, "Exploiting the shared neighborhood to improve the quality of social community detection," in *Proc. 2017 Annu. Conf. New Trends Inf. Commun. Technol. Appl.*, 2017, pp. 52–56.
- [33] S. Jabeen, Z. Mehmood, T. Mahmood *et al.*, "An effective content-based image retrieval technique for image visuals representation based on the bag-of-visual-words model," *PLOS ONE*, vol. 13, no. 4, e0194526, 2018.
- [34] I. M. S. Bimantara and I. M. Widiartha, "Optimization of k-means clustering using particle swarm optimization algorithm for grouping traveler reviews data on Tripadvisor sites," *J. Ilm. Kursor*, vol. 12, no. 1, pp. 1–10, 2023.
- [35] M. Kumar, R. Singh, and P. Mukherjee, "VTHSC-MIR: Vision transformer hashing with supervised contrastive learning based medical image retrieval," *Pattern Recogn. Lett.*, vol. 184, pp. 28–36, 2024.
- [36] L. Zhu, Y. Wang, Y. Hu *et al.*, "Cross-modal contrastive learning with spatiotemporal context for correlation-aware multiscale remote sensing image retrieval," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–21, 2024.
- [37] H. Ren, S. Cheng, L. Wang *et al.*, "Convolution-based visual transformer with dual space shifting attention hashing for image retrieval," *Appl. Soft Comput.*, vol. 171, 112752, 2025.
- [38] C. S. Chiao and D. T. Lin, "ELICE: Embedding language through informative contrastive-encoder," *Expert Syst. Appl.*, vol. 229, 120523, 2023.
- [39] H. Hu, X. Wang, Y. Zhang *et al.*, "A comprehensive survey on contrastive learning," *Neurocomputing*, vol. 610, 128645, 2024.
- [40] K. Wangi and A. Makandar, "Autoencoder for image retrieval system using deep learning technique with tensorflow and kears," in *Proc. 2023 IEEE Int. Conf. Integr. Circuits Commun. Syst.*, 2023, pp. 1–5.
- [41] D. Li, A. Shukla, S. Chandaka *et al.*, "Autoencoder-based representation learning for similar patients retrieval from electronic health records: comparative study," *JMIR Med. Inform.*, vol. 24, e68830, 2025.
- [42] Y. Qiu, X. Yang, and S. Chen, "An improved gray wolf optimization algorithm solving functional optimization and engineering design problems," *Sci. Rep.*, vol. 14, 14190, 2024.
- [43] J. Z. Wang, J. Li, and G. Wiederhold, "SIMPLIcity: Semantics-sensitive integrated matching for picture libraries," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, pp. 947–963, 2001.
- [44] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geogr. Inf. Syst.*, 2010, pp. 270–279.
- [45] S. Tyagi, P. Shukla, P. Singh *et al.*, "Enhanced contentbased image retrieval via hybrid color, texture, and deep learning features," *Sci. Rep.*, vol. 16, 14888, 2026.

Copyright © 2026 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).