Kurdistan Journal of Applied Research (KJAR)
Print-ISSN: 2411-7684 | Electronic-ISSN: 2411-7706

Original Article

Deep Metric Learning with MobileNetV2 and Triplet Loss for Efficient Image Retrieval

Hersh M. Hama [a], **Kamaran H. Manguri** [a *], **Saman M. Omer** [a]

[a] Software Engineering Department, College of Engineering, University of Raparin, Ranya, Sulaymaniyah, Iraq.

Submitted: 04 May 2026
Revised: 23 August 2026
Accepted: 27 August 2026

*** Corresponding Author:**
kamaran@uor.edu.krd

Keywords: Content-based image retrieval, MobileNetV2, Deep metric learning, Dimensional feature reduction.

How to cite this paper: H. M. Hama, K. H. Manguri, S. M. Omer," Deep Metric Learning with MobileNetV2 and Triplet Loss for Efficient Image Retrieval", KJAR, vol. 11, no. 2, pp. 88-101, December 2026, doi: 10.24017/science.2026.2.4

Abstract: In the context of today's digital system applications, content-based image retrieval (CBIR) is an essential tool for managing large visual repositories. Current CBIR systems, however, face a trade-off between retrieval accuracy and computational efficiency, caused by high-dimensional representations and their heavy memory and query-time requirements. To address this problem, this paper proposes an efficient CBIR system based on deep metric learning. The system integrates the MobileNetV2 network for feature extraction, a trainable linear projection head, and triplet-loss optimization. The aim of this integration is to learn compact, discriminative embeddings that improve retrieval accuracy while maintaining a fast search time. The system was trained and tested with the Corel-1K database, and tested with 200 query images. The results show that training the projection head with triplet loss improves embedding quality while reducing the embedding dimensionality from 1280 to 512. This reduction not only improves retrieval accuracy, but also reduces query latency by more than 59%. The proposed model achieved a precision at 10 (P@10) of 97.65% and an average precision at 10 (AP@10) of 97.11%. Moreover, the framework maintained performance at greater retrieval depth, achieving P@20 and AP@20 scores of 97.30% and 96.74%, respectively, exceeding those of several more complex methods. The learned embeddings are also robust across image classes, as further confirmed by the qualitative and class-wise analyses. Overall, the results confirm that compact metric-learning representations provide a practical, scalable, and efficient approach for contemporary CBIR applications.

1. Introduction

The dynamic and rapid growth of digital multimedia content has greatly raised the need of efficient and intelligent image retrieval systems. Traditional text-based image retrieval systems are highly dependent on manual annotations, which are also difficult, subjective, time-consuming and subject to manual errors, and are neither scalable nor accurate [1]. To address this set of shortcomings, content-based image retrieval (CBIR) systems have been developed to directly process visual information for the purpose of image retrieval that can be automatic, objective and semantic driven [2-4]. The typical CBIR pipeline comprises of two main phases: feature extraction and similarity matching [5] where the visual material is transcoded into feature vectors, followed by ranking them by distance from a query image.

The more recent systems use deep learning techniques, such as convolutional neural network (CNN), to obtain high-level semantic feature representations more similar to human perception than handcrafted feature representation systems [6, 7]. However, despite of their success, there are bottlenecks in their practical deployment. In general, high dimensional feature representations

significantly increase storage requirements and computation time and create severe limits in real-time retrieval applications [8]. Thus, the goal of optimizing retrieval accuracy, computational efficiency, and memory usage remains a critical issue in today's visual search engines [9-11].

The most important identified research gap from the literature is that although heavy backbone networks are accurate, they suffer from too many memory requirements, making them impractical for edge applications with limited resources or real-time requirements [12, 13]. Moreover, the standard deep learning frameworks usually train the models based on the standard cross-entropy loss, which is not necessarily well-suited for multi-class similarity distance metrics [14, 15]. In order to fill these gaps, this paper presents a novel compact distance-aware deep metric learning framework for efficient CBIR. The proposed architecture seamlessly combines an ultra-lightweight feature extractor, MobileNetV2 [1], with a learnable linear projection head. An advantage of the proposed method over traditional approaches is the ability to optimize dimensionality reduction and similarity learning at the same time by using triplet-loss-based metric learning [5]. The embedding space is reduced from 1280 to 512 dimensions in this integration; thus, it specifically learns to embed the images compactly while maintaining high discriminative power for cosine similarity ranking. This is a summary of the main contributions of this work:

- An ultra-lightweight MobileNetV2 backbone is used to extract high-quality semantic features from images with minimal computational complexity, suitable for resource-limited environments.
- A trained linear projection head is applied to dynamically compress the feature space from 1280 dimensions down to 512 dimensions, greatly reducing the storage size of the feature index and the latency of searching it.
- The seamlessly integrated distance-aware deep metric learning with triplet margin loss directly optimizes the embedding space, where images with similar visual appearance are embedded in the space as closer vectors, and images with dissimilar appearance are embedded in the space as farther vectors.
- A comprehensive evaluation is carried out using standard benchmarking datasets, showing that the proposed framework can give a better trade-off between computational efficiency and retrieval accuracy than existing state-of-the-art approaches.

The rest of this paper is structured as follows. A thorough summary of related work on this topic is presented in Section 2. The materials and methods used are detailed in Section 3. The results are presented and analyzed in section 4. A discussion is presented in Section 5 and the concluding remarks are included in Section 6.

2. Related Works

CBIR has developed beyond systems that compared images on raw pixel statistics to systems that leverage the representational capability of deep learning. The necessity to bridge the semantic gap, to minimize memory usage, and speed up the retrieval process have been the guiding forces of this development and are the very challenges that propel our suggested framework.

The first CBIR systems used hand-made low-level descriptors following Atlam *et al.* [16] Color histograms calculated in the hue, saturation, and value color space were more perceptually consistent compared to red, green, and blue representations with precision rates of up to 0.77 combined color-texture pipelines. Color features were supplemented with texture descriptors based on the grey-level co-occurrence matrix and discrete wavelet transform, but combinations of these had higher dimensionality and storage requirements [17]. The local feature descriptors scale-invariant feature transform (SIFT) and speeded-up robust features enhanced resistance to geometric distortions and slight occlusion. Zheng *et al.* [18] provided a detailed decade-long overview of instance retrieval techniques, outlining the development from SIFT-based techniques to CNN-based features and demonstrating how the Bag-of-Words technique based on SIFT was dominant in CBIR for more than a decade because of its transformation invariance. Madduri [19] utilized the SIFT algorithm, demonstrating that the 128-dimensional per-keypoint vectors imposed significant memory and computation costs. This limitation directly influenced the subsequent choice of a dimensionality

reduction approach in this study. Furthermore, Li and Wang [20] introduced a statistical model for automatic linguistic indexing of images, establishing fundamental principles to bridge the semantic gap from low-level visual cues to high-level semantic interpretations. Wan *et al.* [21] provided a thorough empirical evaluation of deep learning models used in CBIR tasks, which confirmed that CNNs' feature representations significantly surpass traditional BoW and manually designed descriptor techniques.

Srivastava *et al.* [22] conducted an extensive review of more than 215 CBIR papers, mapping out the paradigm shift toward machine learning representations. In another study, Naveen and Parvathi [23] demonstrated that multi-feature fusion techniques fusing color, texture and shape using automatic weighting schemes consistently outperform single-descriptor techniques; however, the complex automation of weight assignment for non-expert users remains a significant challenge. Furthermore, Singh and Batra [24] introduced architectures featuring a bi-layer separation between coarse and fine matching across processing layers, illustrating that utilizing Zernike moments in the second layer to represent rotation-invariant shapes provides a highly effective cost-accuracy trade-off.

CBIR accuracy has been revolutionized with the use of deep convolutional neural networks. Krizhevsky *et al.* [25] showed that training large deep convolutional neural networks on ImageNet data achieved results significantly better than conventional handcrafted descriptors, thus setting deep feature learning as the de facto standard for visual recognition and the method that allowed transfer learning to be used in later CBIR systems. Li *et al*. [26] reported that the emergence of very deep architectures with the use of small-sized 3×3 stacked convolution filters greatly enhanced image classification and representation learning on a large scale. The Visual Geometry Group (VGG) architecture, which utilized depth increases along with small filter sizes, proved that the deeper the network, the more discriminating and complex the learnt semantic features become. Therefore, pretrained VGG models on ImageNet were widely used as feature extractors in early deep CBIR systems, as it became possible to use semantic representations without retraining the model for each specific task. Jabnoun *et al*. [27] utilized the VGG-16 architecture, demonstrating that its 4096-dimensional fully connected embeddings achieved an average precision of 83.10% on standard benchmarks, thereby directly addressing the semantic gap through the learning of high-level object patterns. He *et al*. [28] proposed deep residual networks (ResNets) where identity shortcuts help to train deeper networks successfully, reaching the highest accuracies on ImageNet data set and serving as the basic architecture in numerous CBIR models thereafter because of their excellent representation abilities. Bibi *et al.* [29] further optimized using genetic algorithm feature selection and an extreme learning machine classifier, where VGG-19 reached 96.57% accuracy on the Wang-A dataset, showing that optimization after extraction can significantly increase retrieval accuracy and also removes redundant dimensions and reduces memory usage. Chen *et al.* [30] undertook a comprehensive survey of deep learning progress on CBIR by discussing recent contributions categorized according to deep network architectural classes, deep learning features, feature embedding and aggregation techniques, and network fine-tuning strategies and showed that deep learnt features always outperform hand-crafted features.

Although these benefits exist, it has massive architectures like VGG-16/19 have, it is associated with large computational costs, and therefore not relevant in resource-constrained or real-time applications. EfficientNet was presented as a compound scaling approach which optimizes the network in terms of its depth, width and resolution in a single run, using a predefined scaling coefficient, thereby showing the superiority of such a principled scaling approach in achieving higher accuracy with reduced numbers of parameters as compared to the conventional approach [31]. This fact compels us to select Sandler *et al*. [1], which is a lightweight backbone that can attain competitive representational capacity despite incurring a small fraction of the memory and inference cost of its more profound counterparts. By freezing pre-trained MobileNetV2 weights and decapping to extract 1280-dimensional global average pooled features, we are able to encode rich ImageNet semantics with little computational cost.

Although transfer learning enhances the quality of features, the generic ImageNet features are not best-suited in tasks of fine-grained image retrieval. This is dealt with by metric learning techniques

which directly attempt to manipulate the embedding space in such a manner that related semantically similar images are geometrically proximate. Schroff *et al*. [5] developed the triplet loss structure which imposes the following constraint on any anchor image: the distance between the image and a positive (same-class) sample (between the image and another sample) should be less than the distance between the image and a negative (different-class) sample (between the image and a different sample) by a margin of at least 0.01. This constraint promotes compactness intra-class and separation between classes in the embedding space, which directly enhances precision in retrieval without necessarily adding labelled annotations other than class membership. According to Mohan *et al*. [32], triplet loss and other related contrastive loss functions have been shown to yield high performance in terms of retrieval accuracy on visual datasets when used with light weight backbones, which makes them ideal for use in edge-based retrieval applications. Yousefzadeh *et al.* [33] added another validation to the advantage of metric learning in retrieval using a triplet-loss dilated residual network, which increases the receptive field while maintaining the same complexity of the model in image retrieval datasets.

Besides feature level efficiency, retrieval speed is also governed by indexing strategy and storage organization. Ma *et al.* [34] introduced a privacy-preserving approach for CBIR using deep learning techniques in cloud computing with maximum average precision of 69.27% and 64.53% in the Corel-10k and Inria Holiday datasets, respectively. It is evident from their results that the demands of accuracy and security can be fulfilled simultaneously. Raghukumar and Naveen [35] developed a domain-specific framework for electrocardiogram report retrieval, utilizing wavelets, cross-correlation, and artificial neural networks to achieve an accuracy between 95% and 98.6 %. By adapting the feature representation to restricted visual distributions, these specialized systems significantly reduce the search complexity as compared to general purpose retrieval engines.

There are, however, three persistent gaps that have been found in the literature review. Firstly, most of the state-of-the-art algorithms utilize expensive deep learning architectures such as VGG-16/19 and ResNets. Such architectures normally use a lot of computation power as well as being bulky in size, making them unsuitable for resource-limited environments. Hence, there is a need for a lightweight architecture. Second, the two algorithmic methods of dimensionality reduction and discriminative optimization are usually used independently without the mutual advantage of learning them simultaneously. Third, Sayed *et al*. [36] and Ahmed [37] demonstrated upper limits of 95–96% for Corel-1K retrieval experiments without the application of metric learning techniques, implying the ability of structured embedding spaces to take performance to the next level.

All three gaps are tackled by the proposed framework of MobileNetV2, a lightweight feature extractor, dimensionality reduction head that is jointly trained, and triplet loss to learn discriminative metrics.

3. Materials and Methods

This work proposes a deep metric learning framework that learns compact, discriminative image embeddings to support CBIR. The pipeline presented in figure 1 has four primary steps: feature extraction based on a pre-trained MobileNetV2 backbone, embedding refinement based on a linear projection head, metric learning based on triplet loss [5], and similarity-based retrieval based on cosine similarity.

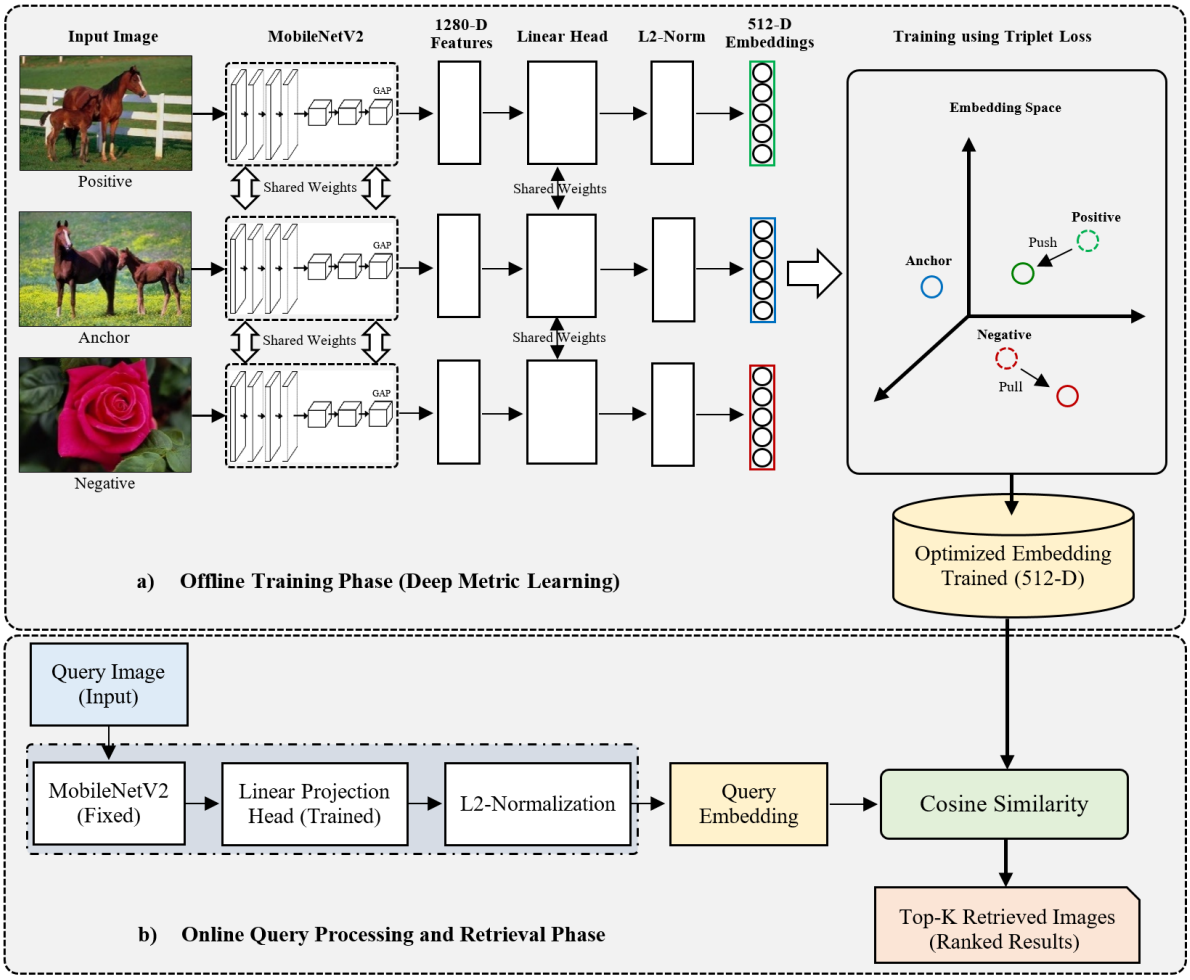

Figure 1: Proposed deep metric learning pipeline for image retrieval.

3.1. Dataset

The proposed retrieval system was trained on the Corel-1K dataset [20], which consists of 1,000 images of either 256 × 384 pixels or 384 × 256 pixels. The data are divided into ten different categories, each of which consists of 100 pictures: African people, beaches, buses, dinosaurs, elephants, flowers, food, horses, monuments, and snow-capped mountains.

For a full assessment, the entire Corel-1K dataset was used to train the model and to build the retrieval database. For the test set, 20 query images per category were collected separately from online sources, giving 200 previously unseen images in total. This protocol measures retrieval performance on images that were not part of the training set and therefore provides a strong indication of how well the system generalizes.

3.2. Feature Extraction Using MobileNetV2

The feature extractor is a pre-trained MobileNetV2 model that was trained on ImageNet [1] because it is lightweight and efficient in computation. The classification layer is removed and the convolutional backbone is frozen during training to minimize computational cost and reduce overfitting. Because deep models tend to overfit on smaller benchmarks such as Corel-1K, the ImageNet-pretrained backbone is kept frozen and only the lightweight projection layer is optimized.

The input images were resized to 224 × 224 pixels to match the standard input size of the pre-trained MobileNetV2 network, thereby preserving its pre-trained feature-extraction capabilities. After resizing, the images are converted into tensors and normalized before processing. The backbone produces a 1280-dimensional feature vector through global average pooling (GAP). The resultant feature representation is L2-normalized to be scale-invariant so that similarity computation is more stable.

3.3. Linear Projection Head

To map the high-dimensional features into a compact embedding space, a fully connected linear projection head is integrated directly after the MobileNetV2 backbone. Let $x \in \mathbb{R}^{1280}$ represent the global average pooled feature vector $h \in \mathbb{R}^{512}$ via a learnable weight matrix $W \in \mathbb{R}^{512\times1280}$ and bias $b \in \mathbb{R}^{512}$. To ensure scale-invariant similarity matching, an L_2-normalization is subsequently applied to project the final embedding ($\|z\|_2 = 1$). This entire operation is mathematically formulated as:

$$z = \frac{W_x + b}{\|W_x + b\|_2} \quad (1)$$

For training, the parameters {W, b} are optimized end-to-end using backpropagation and triplet margin loss. This dense mapping is done for two reasons: first, by reducing the feature dimensionality from 1280 to 512, the number of features stored and the query latency can be significantly reduced and second, the embedding space is restructured to keep critical relationships across features, which enables efficient cosine similarity retrieval.

3.4. Metric Learning with Triplet Loss

To optimize the embedding space for similarity-based retrieval, triplet margin loss is employed [5]. For each anchor sample x_a positive sample x_p is selected from the same class, while a negative sample x_n is selected from a different class. The corresponding embeddings are denoted as f_a, f_p, and f_n, respectively. The triplet loss function is defined as:

$$L(a,p,n) = max\left(0, \|f_a - f_p\|_2^2 - \|f_a - f_n\|_2^2 + \alpha\right) \quad (2)$$

where α is a margin hyperparameter that enforces a minimum separation between positive and negative pairs.

In the learnt embedding space, this goal promotes inter-class separability and intra-class compactness. The Adam optimizer is used to improve the model with a batch size of 16, margin α=0.5, and learning rate of 1×10^{-4} (Table 1).

Triplets are constructed offline at the dataset level before entering the training pipeline. For each anchor instance in the dataset, a positive sample is randomly selected from the same class, and a negative sample is randomly drawn from a distinct class. These pre-constructed triplets are then shuffled and fed into the network using a mini-batch size of 16. Although online advanced triplet mining strategies (such as hard or semi-hard mining) are critical for training deep networks from scratch on massive datasets, they introduce significant computational latency due to frequent pairwise distance updates. Given that the proposed framework leverages a highly robust pre-trained MobileNetV2 backbone on a standardized benchmark (Corel-1K), offline random triplet generation provided an optimal trade-off—minimizing training complexity while preventing vanishing gradients, as evidenced by the smooth loss convergence and high retrieval accuracy.

Table 1: Hyperparameter configurations and training setup for the proposed deep metric learning framework.

Hyperparameters	Value
Margin	0.5
Batch size	16
Number of epochs	20
Learning rate	0.0001

3.5. Retrieval Process

All images in the database are pre-calculated and stored as embeddings once trained. The query image is passed through the same feature-extraction pipeline to obtain its embedding vector. The similarity between query and database embeddings is calculated by cosine similarity:

$$Cosine\ Similarity(f_q, f_i) = \frac{f_q \cdot f_i}{\|f_q\|\|f_i\|} \quad (3)$$

where f_q represents the 512-dimensional embedding vector of the query image, and f_i denotes the embedding vector of the i-th image in the database repository. The notation $\cdot$ signifies the vector dot product, while $\|\cdot\|$ represents the L_2-norm (magnitude) of the respective vectors. This cosine similarity metric is utilized to quantify the semantic closeness between the query and database vectors. Database images are subsequently ranked in descending order based on these computed similarity scores, and the most similar images are returned as the final retrieval results.

3.6. Retrieval Evaluation Metrics

In order to evaluate the performance of the proposed CBIR system quantitatively, popular information retrieval measures are utilized, specifically precision (P) and average precision (AP). These metrics are calculated based on the top-10 retrieved images to provide a comprehensive assessment of the system's search accuracy and retrieval quality.

Precision is the ratio of the retrieved images which are pertinent to the query. A crucial measure of the capability of a system to give correct answers, it is estimated through the following equation:

$$Precision = \frac{|relevant\ images\ \cap\ retrieved\ images|}{|retrieved\ images|} \quad (4)$$

AP gives a one-number overview of the precision-recall curve of a system, which has a more complete perspective of the system performance at various retrieval ranks. In this study, focus is placed on AP within the top-10 results to specifically evaluate the effectiveness of the framework in retrieving relevant images in the first set of returned results. The formula for average precision is given as:

$$AP = \sum_{k=1}^{n} P(k) \cdot \Delta r(k) \quad (5)$$

Here, $P(k)$ is the precision at cut-off k in the list, and $\Delta r(k)$ is the change in recall from item $k-1$ to k.

3.7. Experimental Setup

Python 3.12 with PyTorch and Torchvision were used to implement the proposed model. Experiments were run on a machine with an Intel Core i3-10110U central processing unit (CPU) and 16 GB of random-access memory.

Owing to hardware limitations, all experiments were performed on the CPU alone. Although this restricts scalability to larger datasets, it demonstrates the effectiveness of the proposed lightweight architecture. Data processing and result management were handled with supporting libraries, including NumPy, Pandas, and OpenPyXL.

The evaluation on an Intel Core i3-10110U CPU serves as a baseline to demonstrate the framework's viability for low-power edge applications. In terms of scalability, the 512-dimensional feature embedding yields a minimal memory footprint (approx. 2 KB per image index), enabling efficient in-memory indexing for larger datasets. Furthermore, the entire architecture is structurally compatible with GPU acceleration and vector indexing libraries (e.g., Facebook AI Similarity Search), ensuring high throughput and minimal latency when scaled to large-scale deployment environments.

4. Results

To evaluate the proposed CBIR system, a series of experiments was carried out to examine retrieval accuracy and efficiency. The test was conducted on an external query set of 200 unseen images (20 images per category) collected outside the Corel-1K dataset as indicated in Section 3.1. Precision at top-10 and top-20 results (P@10 and P@20), average precision at top-10 and top-20 results (AP@10 and AP@20), and average retrieval time per query were used to estimate performance. Statistical reliability was verified by computing performance measures on each query separately, reporting the category mean over the 20 queries, and presenting the overall mean across all queries.

4.1. Retrieval Performance

The proposed full architecture combines the MobileNetV2 backbone, a 512-dimensional linear projection head, and metric optimization via triplet margin loss. Optimizing the linear projection head using triplet margin loss achieved the highest retrieval accuracy across all evaluated depths, yielding a P@10 of 97.65%, an AP@10 of 97.11%, a P@20 of 97.30%, and an AP@20 of 96.74%, while maintaining a low average retrieval latency of 1.475 ms/query. Table 2 compares the retrieval performance, feature dimensionality, and computational latency of the different model configurations.

Table 2: Performance and computational metrics across feature representation configurations.

Model	Feature Vector	Average Retrieval Time (ms/query)	P@10 (%)	AP@10 (%)	P@20 (%)	AP@20 (%)
Baseline (MobileNetV2 only)	1280	3.611	94.45	92.32	92.45	89.88
MobileNetV2 + Linear Head (Untrained)	512	1.429	94.15	92.83	92.83	90.90
MobileNetV2 + Trained Linear Head with Triplet Loss	512	1.475	97.65	97.11	97.30	96.74

The baseline configuration (MobileNetV2 with a 1280-dimensional feature representation) achieved a P@10 of 94.45% and an AP@10 of 92.32%, with a P@20 of 92.45% and an AP@20 of 89.88%, at an average retrieval time of 3.611 ms/query.

A linear projection head was then added to reduce the feature dimensionality from 1280 to 512 and improve efficiency. Although it involved no loss optimization, this structural modification markedly accelerated retrieval, reducing the average retrieval time from 3.611 ms/query to 1.429 ms/query (a computational efficiency gains of over 60%). Notably, this configuration preserved retrieval accuracy almost entirely (P@10 = 94.15%, AP@10 = 92.83%, P@20 = 92.83%, and AP@20 = 90.90%).

The principal performance gain arises from optimizing the projection head with the triplet margin loss. Unlike traditional classification losses which only consider each instance in isolation, the triplet loss explicitly models the relationship between instances in the embedding space (a triplet is a group of three images, an anchor, a positive, and a negative).

The loss function enforces a predefined margin, thereby reducing intra-class distances while increasing inter-class distances. As shown in table 2, this distance-aware optimization yields a notable performance increase, improving P@10 from 94.15% to 97.65% (+3.50 percentage points) and AP@10 from 92.83% to 97.11% (+4.28 percentage points). Importantly, this structured embedding space also sustains high performance at greater retrieval depths (P@20 = 97.30% and AP@20 = 96.74%) and demonstrates that metric learning can improve overall precision with a negligible difference in inference cost (1.475 ms/query).

4.2. Qualitative Analysis

To complement the quantitative assessment, a qualitative analysis was carried out to illustrate the retrieval behavior of the proposed CBIR system. Figure 2 presents example queries from four categories (buses, dinosaurs, flowers, and food) alongside their corresponding top-10 retrieved results. Across all sample queries, the retrieved results exhibit high visual and semantic consistency with the query images.

Figure 2: The ten most similar images retrieved by query image using the proposed model.

Figure 3 details the class-specific precision at both short-range (Top-10) and extended (Top-20) retrieval depths. High accuracy is maintained across individual categories, with several classes, such as elephant, flower, food, horse, monument, and snow-capped mountains, and achieving perfect retrieval scores.

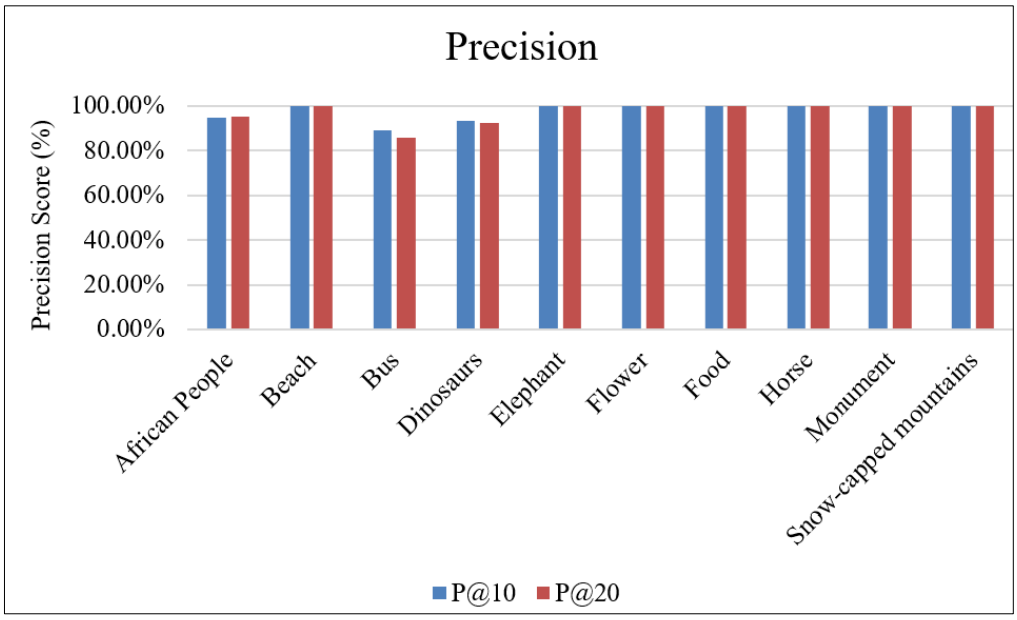

Figure 3: Precision for each image class.

Table 3 reports the precision (P@10, P@20) and average precision (AP@10, AP@20) for each class, corresponding to figures 3 and 4, across all ten evaluated classes.

Table 3: Category-specific performance metrics across all evaluated classes.

Class Name	P@10 (%)	AP@10 (%)	P@20 (%)	AP@20 (%)
African People	94.50	92.95	95.25	93.30
Beach	100.00	100.00	99.75	99.61
Bus	89.00	86.38	85.50	83.06
Dinosaur	93.00	91.79	92.50	91.43
Elephant	100.00	100.00	100.00	100.00
Flower	100.00	100.00	100.00	100.00
Food	100.00	100.00	100.00	100.00
Horse	100.00	100.00	100.00	100.00
Monument	100.00	100.00	100.00	100.00
Snow-capped mountains	100.00	100.00	100.00	100.00
Average Overall	97.65	97.11	97.30	96.74

Under the extended evaluation setting (Top-20), a slight decrease in precision is observed for some classes, including the bus and dinosaur classes, as detailed in table 3. These classes exhibit larger precision drops than the more visually homogeneous categories, such as food and flowers.

Figure 4 presents the per-class average precision at both AP@10 and AP@20. The evaluation demonstrates consistently high values across categories, with minimal performance drop observed when extending the search depth from Top-10 to Top-20.

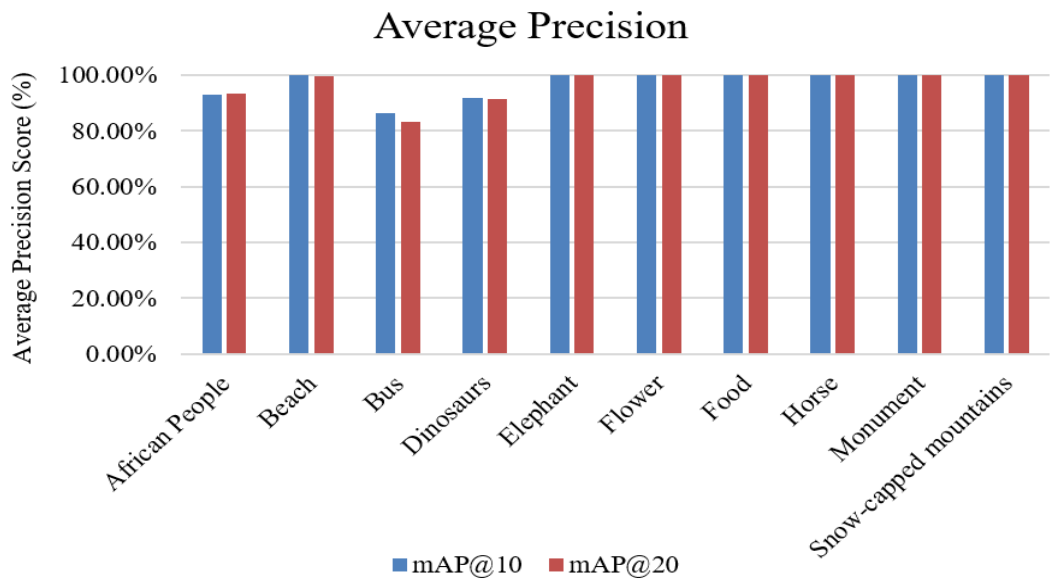

Figure 4: Average precision for each image class.

5. Discussion

The results presented in Section 4 show the proposed framework delivers strong performance for CBIR on the Corel-1K dataset, consistently outperforming established state-of-the-art architectures. Notably, the proposed model outperforms ResNet18 [37] despite using a lighter feature extractor. This improvement highlights that using a lightweight MobileNetV2 backbone in conjunction with a trained linear projection head and distance-aware constraints can achieve higher retrieval accuracy than structurally heavier networks. This result is consistent with the findings reported by Zheng *et al.* [18], which indicate that changing the geometric structure of the embedding space is more important than scaling only the depth of the feature extractor for fine-grained retrieval.

To further confirm the effectiveness of the proposed framework, its performance was compared with several state-of-the-art CBIR methods evaluated on the Corel-1K dataset. The precision and average precision at both the top-10 and top-20 results (P@10, AP@10, P@20, and AP@20) are compared using the values reported in the corresponding studies. Table 4 summarizes the results, where "-" indicates that the respective metric was not reported in the original literature.

Table 4: Performance comparison of the proposed framework against state-of-the-art CBIR methods on the Corel-1K dataset.

Model	P@10 (%)	AP@10 (%)	P@20 (%)	AP@20 (%)
[36]	93.90	-	92.80	-
[37]	95.50	89.40	-	-
[38]	89.00	91.00	-	-
[39]	-	88.00	-	-
[40]	85.27	-	-	-
[41]	82.27	-	76.13	-
Proposed Method	**97.65**	**97.11**	**97.30**	**96.74**

As shown in table 4, the proposed method achieves the highest performance among the compared methods, with a P@10 of 97.65% and an AP@10 of 97.11%. Specifically, the proposed model increases AP@10 by a significant margin over the best-reported baseline in literature AP@10 = 91% [38]. Furthermore, it consistently outperforms other established benchmark methods, such as those reported in [39] (AP@10 = 88.00%) and [40] (P@10 = 85.27%).

Importantly, the superiority of the proposed framework remains highly consistent when evaluated at a deeper retrieval depth. At the top-20 threshold, the proposed method achieves a P@20 of 97.30% and an AP@20 of 96.74%. This represents a precision difference of +4.50 percentage points compared to CCBIR [36] (P@20 = 92.8%) and a +21.17% improvement over the baseline model in [41] (P@20 = 76.13%).

It should be noted that most of the compared methods limit their evaluation to short-range retrieval and do not report both metrics, which in some cases prevents a fully comprehensive comparison. However, across all available metrics, the proposed method consistently achieves strong performance and establishes a robust benchmark for future evaluations.

A key indicator of the framework's robustness is its ability to preserve performance at deeper retrieval depths. Whereas conventional systems reported in the literature often suffer from a sharp decrease in accuracy due to feature drift and semantic overlap [41], the proposed architecture successfully mitigates this degradation. The optimization pipeline explicitly enforces both intra-class compactness and inter-class separability through the use of triplet margin loss ($\alpha = 0.5$), which aligns with the fundamental premise of metric learning under structured margin constraints [15]. Moreover, the high AP@10 and AP@20 scores indicate that the method is highly effective at ranking relevant images among the top positions of the retrieved results.

Overall, these findings directly address three key gaps identified in current CBIR literature. First, regarding the high dimensionality versus computational overhead gap, while conventional models rely on dense feature representations, compressing features into a 512-dimensional subspace significantly lowers latency and indexing overhead without sacrificing retrieval accuracy. Second, concerning the long-range retrieval instability gap, unlike existing methods that degrade rapidly at extended search boundaries, the proposed triplet loss optimization preserves clean semantic margins even when the retrieval pool doubles. Third, addressing the inconsistent metric protocol gap, while prior studies [36, 37, 39] often omit long-range evaluations or average precision metrics, this study establishes a standardized evaluation framework across both Precision and AP at multiple depth thresholds.

By combining targeted dimensionality reduction and deep metric learning, the system achieves high retrieval precision while remaining computationally efficient, requiring only 1.475 ms per query. This lightweight profile makes the framework highly suitable for real-time, resource-constrained edge

applications. Nevertheless, a remaining limitation is that the model was evaluated on a curated benchmark dataset; future work will focus on testing generalization across large-scale, uncurated web datasets and exploring feature quantization for low-power hardware.

6. Conclusions

The work presents an efficient and robust CBIR architecture that combines a lightweight convolutional neural network backbone with a dimensionality-reducing projection head, trained using deep metric learning with a triplet margin loss. The principal contribution of this work is the demonstration that enforcing geometric distance constraints within a lower-dimensional subspace is sufficient to produce a highly discriminative embedding space. Moreover, this structural modification achieves fast inference and substantially reduces computational and indexing overhead, providing a favorable trade-off between retrieval accuracy and architectural compactness. The framework avoids excessive memory and hardware requirements at the retrieval stage, making it well-suited for real-time applications in resource-limited environments.

The system achieved high retrieval accuracy and computational efficiency on a standard benchmark dataset, but had some limitations. More complex or deeper hybrid backbones were not explored, owing to the limited computational resources available during training. Furthermore, although the framework performs well, validation was carried out on structured benchmark images with limited background clutter and well-defined classes. To fully evaluate the limits of generalization of this framework under extreme visual variations, future work should focus on scaling the system to larger, more diverse, and cluttered real-world datasets. In addition, alternative distance-based loss functions, cross-attention feature-fusion schemes, and vision transformer backbones should be explored to further enhance the robustness and scalability of the architecture.

Author contributions: Hersh M. Hama: Conceptualization, Methodology, Software, Validation, Formal Analysis, Visualization, Writing – original draft. **Kamaran H. Manguri:** Project administration, Investigation, Writing – review & editing. **Saman M. Omer:** Supervision, Writing – review & editing,

Data availability: Data will be available upon reasonable request by the authors.

Conflicts of interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding: The authors did not receive support from any organization for the conducting of the study.

References

[1] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.

[2] A. Latif *et al.*, "Content-based image retrieval and feature extraction: a comprehensive review," *Mathematical Problems in Engineering*, vol. 2019, art. no. 9658350, pp. 1–21, Aug. 2019, doi: 10.1155/2019/9658350.

[3] Y. Liu, D. Zhang, G. Lu, and W. Y. Ma, "A survey of content-based image retrieval with high-level semantics," *Pattern Recognition*, vol. 40, no. 1, pp. 262–282, Jan. 2007, doi: 10.1016/j.patcog.2006.04.045.

[4] I. M. Hameed, S. H. Abdulhussain, and B. M. Mahmmod, "Content-based image retrieval: a review of recent trends," *Cogent Engineering*, vol. 8, no. 1, art. no. 1927469, pp. 1–37, Jun. 2021, doi: 10.1080/23311916.2021.1927469.

[5] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 815–823, doi: 10.1109/CVPR.2015.7298682.

[6] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: an astounding baseline for recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Columbus, OH, USA, Jun. 2014, pp. 806–813, doi: 10.1109/CVPRW.2014.131.

[7] S. R. Dubey, "A decade survey of content based image retrieval using deep learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 2687–2704, May 2022, doi: 10.1109/TCSVT.2021.3080920.

[8] M. A. Belarbi, S. Mahmoudi, G. Belalem, S. A. Mahmoudi, and A. Cools, "A new comparative study of dimensionality reduction methods in large-scale image retrieval," *Big Data and Cognitive Computing*, vol. 6, no. 2, p. 54, May 2022, doi: 10.3390/bdcc6020054.

[9] F. Yang *et al.*, "Visual search at eBay," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Halifax, NS, Canada, Aug. 2017, pp. 2101–2110, doi: 10.1145/3097983.3098162.

[10] V. Khullar *et al.*, "Multiple model visual feature embedding and selection method for an efficient pest classification supporting precision agriculture," *Scientific Reports*, vol. 15, no. 1, art. no. 31791, Aug. 2025, doi: 10.1038/s41598-025-16942-1.

[11] A. Gordo, J. Almazan, J. Revaud, and D. Larlus, "Deep image retrieval: Learning global representations for image search," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Amsterdam, The Netherlands, Oct. 2016, pp. 241–257, doi: 10.1007/978-3-319-46466-4_15.

[12] A. Sharshar, L. U. Khan, W. Ullah, and M. Guizani, "Vision-language models for edge networks: a comprehensive survey," *IEEE Internet of Things Journal*, vol. 12, no. 16, pp. 32701–32724, Aug. 2025, doi: 10.1109/JIOT.2025.3579032.

[13] S. Naveen and M. R. Kounte, "Memory optimization at Edge for Distributed Convolution Neural Network," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 12, art. no. e4648, Sep. 2022, pp. 1–18, doi: 10.1002/ett.4648.

[14] K. Musgrave, S. Belongie, and S. N. Lim, "A metric learning reality check," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Glasgow, UK, Aug. 2020, pp. 681–699, doi: 10.1007/978-3-030-58595-2_41.

[15] M. Kaya and H. S. Bilge, "Deep metric learning: a survey," *Symmetry*, vol. 11, no. 9, art. no. 1066, Aug. 2019, doi: 10.3390/sym11091066.

[16] H. F. Atlam, G. Attiya, and N. El-Fishawy, "Integration of color and texture features in CBIR system," *International Journal of Computer Applications*, vol. 164, no. 3, pp. 23–29, Apr. 2017, doi: 10.5120/ijca2017913600.

[17] N. K. Rout, M. Atulkar, and M. K. Ahirwal, "A review on content-based image retrieval system: present trends and future challenges," *International Journal of Computational Vision and Robotics*, vol. 11, no. 5, pp. 461–485, Sep. 2021, doi: 10.1504/IJCVR.2021.117578.

[18] L. Zheng, Y. Yang, and Q. Tian, "SIFT meets CNN: A decade survey of instance retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 5, pp. 1224–1244, May 2018, doi: 10.1109/TPAMI.2017.2709749.

[19] A. Madduri, "Content based image retrieval system using local feature extraction techniques," *International Journal of Computer Applications*, vol. 183, no. 20, pp. 16–20, Aug. 2021, doi: 10.5120/ijca2021921549.

[20] J. Li and J. Z. Wang, "Automatic linguistic indexing of pictures by a statistical modeling approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1075–1088, Sep. 2003, doi: 10.1109/TPAMI.2003.1227984.

[21] J. Wan *et al.*, "Deep learning for content-based image retrieval: a comprehensive study," in *Proceedings of the 22nd ACM International Conference on Multimedia*, Orlando, FL, USA, Nov. 2014, pp. 157–166, doi: 10.1145/2647868.2654948.

[22] D. Srivastava, S. S. Singh, B. Rajitha, M. Verma, M. Kaur, and H.-N. Lee, "Content-based image retrieval: a survey on local and global features selection, extraction, representation, and evaluation parameters," *IEEE Access*, vol. 11, pp. 95410–95431, Aug. 2023, doi: 10.1109/ACCESS.2023.3308911.

[23] K. Naveen and R. M. S. Parvathi, "AI powered multi feature fusion framework for retrieving images using color, texture and shape descriptors," *Scientific Reports*, vol. 15, art. no. 45601, Nov. 2025, pp. 1–16, doi: 10.1038/s41598-025-29719-3.

[24] S. Singh and S. Batra, "An efficient bi-layer content based image retrieval system," *Multimedia Tools and Applications*, vol. 79, no. 25, pp. 17731–17759, Jul. 2020, doi: 10.1007/s11042-019-08401-7.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, Jun. 2017, doi: 10.1145/3065386.

[26] Y. Li, J. Yang, and J. Ma, "Recent developments of content-based image retrieval (CBIR)," *Neurocomputing*, vol. 452, pp. 675–689, Sep. 2021, doi: 10.1016/j.neucom.2020.07.139.

[27] J. Jabnoun, N. Haffar, A. Zrigui, S. Nsir, H. Nicolas, and A. Trigui, "An image retrieval system using deep learning to extract high-level features," in *Proceedings of the International Conference on Computational Collective Intelligence (ICCCI)*, Hammamet, Tunisia, Sep. 2022, pp. 167–179, Springer, doi: 10.1007/978-3-031-16210-7_13.

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[29] R. Bibi, Z. Mehmood, A. Munshi, R. M. Yousaf, and S. S. Ahmed, "Deep features optimization based on a transfer learning, genetic algorithm, and extreme learning machine for robust content-based image retrieval," *PLOS ONE*, vol. 17, no. 10, art. no. e0274764, Oct. 2022, doi: 10.1371/journal.pone.0274764.

[30] W. Chen, Y. Liu, W. Wang, E. M. Bakker, T. Georgiou, P. Fieguth, L. Liu, and M. S. Lew, "Deep learning for instance retrieval: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 6, pp. 7270–7292, Jun. 2023, doi: 10.1109/TPAMI.2022.3218591.

[31] H. Alhichri, A. S. Alswayed, B. Bazi, N. Ammour, and N. A. Alajlan, "Classification of remote sensing images using EfficientNet-B3 CNN model with attention," *IEEE Access*, vol. 9, pp. 14078–14094, Jan. 2021, doi: 10.1109/ACCESS.2021.3051085.

[32] D. D. Mohan, B. Jawade, S. Setlur, and V. Govindaraju, "Deep metric learning for computer vision: A brief overview," *Handbook of Statistics*, vol. 48, pp. 59–79, Jan. 2023, doi: 10.1016/bs.host.2023.01.003.

[33] S. Yousefzadeh, H. Pourreza, and H. Mahyar, "A triplet-loss dilated residual network for high-resolution representation learning in image retrieval," *Journal of Signal Processing Systems*, vol. 95, pp. 529–541, May 2023, doi: 10.1007/s11265-023-01865-9.

[34] W. Ma, T. Zhou, J. Qin, X. Xiang, Y. Tan, and Z. Cai, "A privacy-preserving content-based image retrieval method based on deep learning in cloud computing," *Expert Systems with Applications*, vol. 203, art. no. 117508, Oct. 2022, doi: 10.1016/j.eswa.2022.117508.

[35] B. S. Raghukumar and B. Naveen, "Analysis on CBIR system for ECG reports," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 5, pp. 586–592, Jun. 2020, doi: 10.35940/IJEAT.C6229.069520.

[36] M. S. Sayed, A. A. A. Gad-Elrab, K. A. Fathy, and K. R. Raslan, "Unsupervised content based image retrieval using pre-trained CNN and PCNN features extractors," *International Journal of Intelligent Engineering and Systems*, vol. 16, no. 1, pp. 584–596, Feb. 2023, doi: 10.22266/ijies2023.0228.50.

[37] A. Ahmed, "Pre-trained CNNs models for content based image retrieval," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 7, pp. 200–206, Jul. 2021, doi: 10.14569/IJACSA.2021.0120723.

[38] R. Kumar and N. Murthy, "Relevance-aware content-based image retrieval using deep hybrid feature extraction," *Engineering, Technology & Applied Science Research*, vol. 15, no. 3, pp. 22976–22982, Jun. 2025, doi: 10.48084/etasr.10767.

[39] R. Q. Hassan, Z. N. Sultani, and B. N. Dhannoon, "Content-based image retrieval based on corel dataset using deep learning," *IAES International Journal of Artificial Intelligence*, vol. 12, no. 4, pp. 1854–1863, Dec. 2023, doi: 10.11591/ijai.v12.i4.

[40] A. Orman, "Content-based image retrieval using composite feature vectors with edge features based on color and pixel similarity," *Traitement du Signal*, vol. 41, no. 4, pp. 1945–1952, Aug. 2024, doi: 10.18280/TS.410424.

[41] S. F. Salih and A. A. Abdulla, "An improved content based image retrieval technique by exploiting bi-layer concept," *UHD Journal of Science and Technology*, vol. 5, no. 1, pp. 1–12, Jan. 2021, doi: 10.21928/uhdjst.v5n1y2021.