

Research Article

An Efficient Feature Extraction and Image Retrieval Based on Convolutional Neural Networks

Sawsan Mousa Mahmoud¹ , Hanan A. S. Al-Jubouri^{1,*} 

¹ Department of Computer Engineering, College of Engineering, Mustansiriyah University, Baghdad, 10052, Iraq

*Corresponding Author: Hanan A. S. Al-Jubouri, E-mail: hananaljubouri@uomustansiriya.edu.iq

Article Info	Abstract
Article History	Image retrieval has become increasingly important in many application domains, including medicine, forensics, biology, astronomy, fashion, and military systems, where large volumes of digital images are generated and stored daily. Consequently, there is a growing need to retrieve images based on their visual content, such as color, texture, and shape, rather than relying on keywords or textual annotations that may be incomplete or unavailable. Content-Based Image Retrieval (CBIR) has been developed to address this challenge. However, designing effective image features remains difficult due to the semantic gap between human perception and machine-level representation. This research proposes a novel hybrid feature extraction and fusion approach for image retrieval, termed Fuse Convolutional Neural Network Models (FCNNM). The proposed method leverages multiple layers of Convolutional Neural Networks (CNNs) to learn discriminative image representations while avoiding the high dimensionality and computational complexity associated with dense layers in traditional Deep CNNs. Instead, the model relies on sequential convolutional and pooling layers. Two CNN architectures with three and four layers, respectively, are designed, and their extracted features are fused to improve retrieval performance. The proposed approach is evaluated using benchmark datasets, namely Corel-1K and Corel-5K. Experimental results demonstrate that fusing 2D CNN 3- and 4-layer models significantly enhances retrieval accuracy, achieving 97% on Corel-1K and 95% on Corel-5K, while maintaining high computational efficiency.
Received Nov01, 2025	
Revised Dec 08, 2025	
Accepted Dec 14, 2025	
Keywords	
Content-Based Image Retrieval (CBIR)	
Deep learning models	
Feature extraction	
Image classification	
Convolutional Neural Networks	



Copyright: © 2026 Sawsan Mousa Mahmoud, and Hanan A. S. Al-Jubouri. This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY 4.0) license.

1. Introduction

A large volume of digital images is acquired and stored in databases daily using various photography devices across communities, such as researchers, doctors, police, and individuals. In traditional image retrieval methods, images are manually annotated with keywords or phrases, which may be unavailable or incomplete. This process is also time-consuming, especially in large-scale image databases. Consequently, there is a need to retrieve images automatically. In the 1990s, CBIR emerged, in which images are indexed based on visual content (e.g., colour, texture, and shape). However, CBIR suffers from the so-called

semantic gap between the high-level meaning humans perceive and the low-level machine representation. This gap arises from the lack of sufficiently discriminative features and from limitations of the similarity measures used to compare query images with database images [1]. Hence, this paper aims to extract effective image features using CNNs and to develop an algorithm that enhances image retrieval effectiveness.

Typically, a CBIR system involves three stages: preprocessing, extracting descriptors, and measuring similarity [2]. These stages are traversed by loaded images and query images in the offline and online phases, respectively, as illustrated in Figure 1. A ranked list of retrieved images is produced from the third stage. However, the list may include some images that are not similar to the query image due to a common object, shortcomings of the similarity measure, and perceptual differences in image descriptors (i.e., the semantic gap). Thus, exploring accurate descriptors and similarity measures remains an open topic for interested scientific researchers.

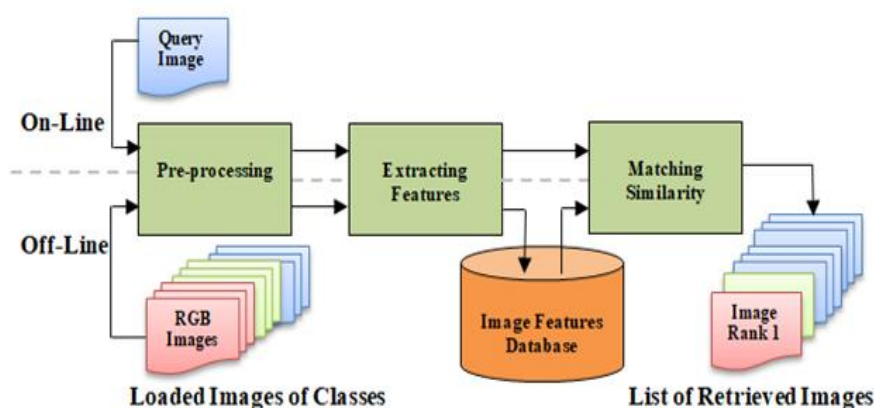


Figure 1. A typical CBIR system

The fundamental contribution of this research is the proposed methodology, which presents a novel strategy for image classification and retrieval by combining different CNN models with varying convolutional layer configurations. Image features are extracted using several convolutional layers, and the combination of various CNN designs is compared to the existing approaches. The suggested model is primarily designed to simplify standard CNN approaches while improving average precision and relevance in image retrieval. Also, the model can learn and extract effective features that discriminate between images. Another contribution is a proposed novel algorithm for image retrieval, so-called Fuse CNN Models (FCNNM). The algorithm's strategy is to combine the performance of two proposed CNN models (i.e., 2DCNN-3 layers and 2DCNN-4 layers) to increase image retrieval accuracy. This study tries:

- To use the 2DCNN-3 and 2DCNN-4 models to extract features and improve the effectiveness of the image retrieval system.
- To apply the Fuse CNN Models (FCNNM) to combine the performance of the two CNN models and increase retrieval accuracy.
- To evaluate the models on the Corel-1K and Corel-5K datasets using Precision (P), Average Precision (AP), and mean Average Precision (mAP).
- To compare the proposed model with previous methods and demonstrate its advantages in retrieval performance.
- To test the hypothesis that shallow architectures can achieve precision comparable to deep CNNs while using fewer parameters.

2. Literature Review

Many studies on CBIR focusing on hand-crafted aspects have been published [1, 3-5]. A computer's low-level features are still inferior to a human's perception of images. For example, the histogram representation of the image cannot retain spatial information, which is important for accurately reflecting the image's content. In CBIR, intelligent techniques are used to extract significant features and help accurately retrieve images. Using artificial neural networks in CBIR alters how traditional CBIR methods operate [6]. For example, an algorithm was presented in Irtaza, et al. [7] to retrieve images, which first applied a wavelet packet tree to extract the first feature. Then, a detailed Gabor analysis was performed on the smallest approximation image of the wavelet packet tree to obtain the second feature, selecting a set of Eigenvalues for each Gabor response image. Both features were concatenated to obtain a single texture feature. Next, Bags of Images (BOIs) were produced, and each class was trained on the sample images in its relevant BOI using a back-propagation neural network trained on those BOIs. The query image is classified into the appropriate semantic class using these trained neural networks, taking into account its top "K" nearest neighbours, determined via Pearson correlation.

Since 2012, researchers have increasingly focused on CNNs and Deep Learning Networks (DLNs) because their layered architectures can learn effective image representations [8]. DLNs are a type of machine learning commonly used to recognise objects, find patterns, understand natural language, and process images, such as classifying different objects. A suggested approach used triplet Siamese networks

within a more complex architecture that pooled and aggregated across several regions [9]. A method used by DLNs to train large-scale database images. To obtain discriminative image features, bi-linear root pooling was proposed to address the issue of long feature vectors. In addition, back-propagation training was performed to produce a robust architecture and to obtain image retrieval features in Alzu'bi, et al. [10]. A framework [11] uses a retraining methodology in which a DLN is first applied to obtain a feature representation, and the network is then retrained based on information availability, fully unsupervised relevance information, and relevance feedback. The aim was to develop more efficient image descriptors that perform well on retrieval tasks and have low memory requirements.

Many CNN models have been used for image classification, and features extracted from fully connected layers. Also, different CNN models are compared and evaluated, such as VGG-16, VGG-19, and ResNet-50 [6, 10]. Deep learning has recently been adopted as a superior choice for hierarchical feature learning, preserving both low- and high-significance features while significantly increasing complexity [12]. For example, a new deep CNN model was constructed in Khan [12] using three convolutional layers (Conv-3 DCNN) to diagnose three diseases in apple plants from leaf images. The CNN structure here has fewer layers, thereby requiring less storage and computation. Also, Khan, et al. [13], three CNNs at each step of the proposed process to identify the cause of pneumonia on radiographs, whether COVID-19 or other causes. The results show that, with only three CNNs and a small database, the method used improves pneumonia detection and reveals the mechanisms underlying its cause. In Pipanmekaporn, et al. [14] deep image retrieval model, is proposed. It is a layer used to extract the image pyramid structure. After that, it generates a compact global descriptor for visual landmark search. This descriptor is improved using triplet loss. Also, Hu and Bors [15] proposed an efficient co-attention method for large-scale CBIR. Local feature selection and clustering methods are used to reduce the computational cost caused by query sensitivity.

Recent advancements in CBIR have also used multimodal feature representations and large pre-training datasets (e.g., ImageNet-21K and LAION-400M). In addition, Vision Transformers (ViTs), DINOv2, and CLIP are examples of large-scale transformer designs and self-supervised learning techniques that offer significant potential for further advancement. Although they have shown impressive retrieval performance, they require more computing power and training time and are highly GPU-intensive. For example, Radford, et al. [15] achieved a high retrieval accuracy of about 99%, but it requires many GPUs for inference.

ViT-B/16 and DINO were used by Oquab, et al. [16], where they achieved high retrieval precision (about 98%-99%), but required complex model designs. The performance of a high-performance image overlay descriptor was examined by El Mehdi, et al. [17]. The suggested descriptor, GDF-HOG, was developed by combining several Gaussian derivative filters. For dimension reduction, the principal component analysis technique and an enhanced AlexNet CNN were employed. According to the current analysis, the suggested descriptor yields analog image retrieval results comparable to those of existing descriptors. Another feature has been suggested: a combination of hand-crafted features (e.g., color histogram, color moments, wavelet transform, and LBPs) and features extracted from known DLNs (e.g., Vgg16, Vgg19, MobileNet, MobileNetv2, and ResNet50). In terms of data privacy and model complexity, for example, Aung and Aung and Khan [18] demonstrate that adding an attention mechanism can enhance feature representation in lung nodule segmentation, improving accuracy and quality without resorting to deeper or more sophisticated networks. In addition, decentralised and privacy-preserving learning frameworks can reduce the risks associated with data sharing while maintaining a competitive level of performance, as presented by Khan, et al. [19].

In this paper, we use multiple CNN models with varying numbers of layers to extract effective features and retrieve images, rather than relying on a single, more complex CNN architecture. Fusing the two models increases the mean average precision by 6% on Corel-5K and 4% on Corel-1K. In addition, the models require less computing power and training time. This shows that our retrieval model is efficient and easy to use. In addition, our model improves accuracy and feature representation while maintaining computational efficiency and flexibility in data privacy protection. The explanation is that our work employs two CNN models with different three- and four-layer configurations, then combines their outputs without additional processing, even for networks with greater depth or complexity. The model performs well at retrieval and can be adjusted to protect privacy, as shown by comparison with current research.

3. Method

This section provides a detailed description of our proposed CBIR algorithm. The algorithm uses multiple CNN models to learn image features, which is an issue in the CBIR field, as mentioned earlier. Thus, the algorithm starts by loading images and using the proposed 2DCNN-3-layer and 2DCNN-4-layer models to extract features, as shown in Figure 2.

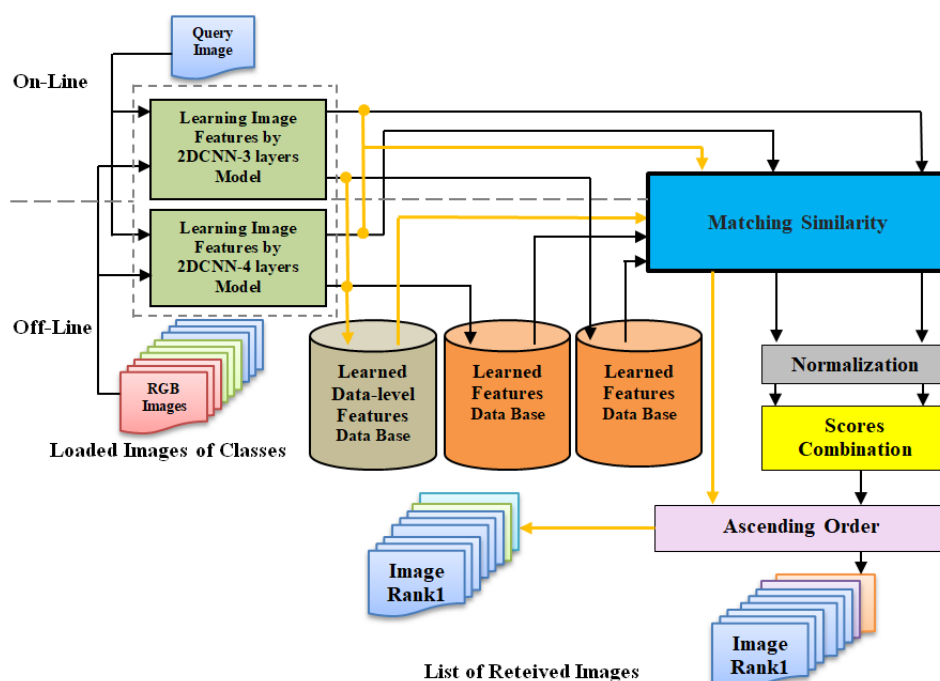


Figure 2. Proposed framework of the CBIR algorithm

Basically, the central component of a CNN is the convolutional layer. Using a set of learnable filters, it applies a convolution operation to the input image. At each stage, the filters, which are tiny matrices that slide over the input image, compute a dot product between the filter and a small portion of the input. The convolutional layer's output is a collection of feature maps that capture various aspects of the input image [20]. Hence, the suggested model is built using a 2D CNN with varying numbers of 2D convolutional layers. The parameters of the suggested CNN model are illustrated in Table 1.

Table 1. Parameters of the suggested CNN model

Layer	Layer Type	Configuration
1	Image Input	28 x 28 x 3 image input
2	Conv2D	15 filters 25 x 25 kernel, [1 1] stride, valid padding
3	Normalisation	Batch normalisation
4	Max Pooling 2D	2 × 2 max pooling, stride 1 × 1, valid padding
5	Conv2D+ReLU	20 filters, 25 × 25 kernel, stride 1 × 1, valid padding
6	Normalisation	Batch normalisation
7	Max Pooling 2D	2 × 2 max pooling, stride 1 × 1, valid padding
8	Conv2D+ReLU	25 filters, 25 × 25 kernel, stride 1 × 1, valid padding
9	Normalisation	Batch normalisation
10	Max Pooling 2D	2 × 2 max pooling, stride 1 × 1, valid padding
11	Fully Connected	50 neurons
12	SoftMax	Softmax activation
13	Classification	2-class output

Mathematically, the suggested model architecture can be represented as follows:

- 1) *Input layer*: Its function is to receive images, resize them, and then pass them to subsequent layers for feature extraction. In this work, the image dimensions are $28 \times 28 \times 3$.
- 2) *Convolutional layers* are made up of several CNN layers, each containing $n \times m$ convolution kernels, where n represents the size of the filter and m is the number of filters conducting convolution operations on the input images. Convolutional layer functions use filters to identify features in images, making them useful for detecting feature points during testing [21]. In addition, using stride [1 1] allows us to preserve more detailed information than other methods that use longer strides could in environments with high down-sampling rates (where a large portion of the image data is lost), while still maintaining dimensionality-reduction techniques. Followed by normalisation layers, ReLU layers, and max pooling layers.
- 3) *The ReLU layer* sets all negative values to 0. This ensures the CNN's mathematical stability by preventing learned values from becoming stuck near 0 or rapidly increasing toward infinity [18]. The convolution operation produces feature maps with a high-level vector representation. To reduce this representation, we added a max pooling layer after the convolution layer, which helps select significant information by eliminating weak activations. This helps minimise overfitting due to noisy data [22].
- 4) *A fully connected layer* that takes the high-level filtered images and transforms them into labelled categories [21].
- 5) *SoftMax layer*.
- 6) *Classification layer*.

For 2DCNN-3 layers, there are three convolutional 2D layers (Conv2D), while for 2DCNN-4 layers, there are four Conv2D layers. Then, for image classification and retrieval, 2DCNN-3 layers and 2DCNN-4 layers are combined with varying convolutional layer configurations (see Table 1 for parameter configurations). Table 2 compares our proposed method with other CBIR baselines. It shows the backbone, pooling, feature dimensionality, datasets, and evaluation metrics between these methods.

The model is trained on all training sets, and the parameters of each layer are defined for image classification and feature extraction. In CNN models with many deep layers, overfitting and vanishing gradients are two problems that may arise. We employ a shallow design (2DCNN-3 layers and 2DCNN-4 layers), not as a simple approach, but because the deployment environment has severe latency, computation, and memory limitations. Each additional layer, channel, or feature map dimension directly raises FLOPs,

memory footprint, and data movement costs. On the other hand, deep models sometimes require storing large amounts of high-dimensional intermediate activations and executing many layers serially.

Table 2. A comparison between our proposed method and other CBIR baselines

Method	Backbone	Pooling	Feature Dimensionality	Datasets	Evaluation Metrics
VGG + MAC	VGG-16	Max	512	Corel-1K, Corel-5K	P@20, mAP@20
ResNet-50 + GeM	ResNet-50	Generalised Mean	2048	Oxford5K, Paris6K	mAP
MobileNet + R-MAC	MobileNetV2	Regional Max	1024	Corel-1K	mAP
Densenet + AvgPool	DenseNet-121	Average	1024	Corel-5K	P@10, mAP@10
Proposed 2DCNN-3 (Ours)	Shallow CNN (3 conv layers)	Max	128	Corel-1K, Corel-5K	P@10, mAP@10
Proposed 2DCNN-4 (Ours)	Shallow CNN (4 conv layers)	Max	256	Corel-1K, Corel-5K	P@20, mAP@20

The next step in the algorithm is storing the learned features in databases for later use. When the features of a query image are extracted, they are compared with those in the database using the cosine distance function (i.e., a similarity measure). This process uses both 2DCNN-3-layer and 2DCNN-4-layer models. Obtained scores are normalised and then combined to produce a sorted list in ascending order. Finally, a ranked list of images is retrieved. For CBIR, the retrieval depends on the learned feature representations rather than the classification layer. The learned feature layer used for retrieval is obtained from the last Conv2D layer before the fully connected and SoftMax output layers. In our work, the 2DCNN-3 layers and 2DCNN-4 layers extract features, which are then used to calculate image similarity.

4. Dataset

We have used datasets characterised by the following attributes: Corel-5K [23] contains 5000 images divided into 50 classes, where images are (126x187) or (187x126) in size. Meanwhile, the dataset Corel-1K [23], which is a standard and known as WANG to compare with others' approaches, contains 1000 images divided into 10 classes (African people, Beach, Buildings, Buses, Dinosaurs, Elephants, Flowers, Horses, Mountains, and Foods), where images are (256 x 384) or (384 x 256) in size, each class contains 100 images. For data preprocessing and generalisation, data augmentation is implemented to crop, scale, and flip images.

Each image is initially sized at about 256×384 pixels. In our work, the images are cropped, centred,

and resized to $28 \times 28 \times 3$ to match the input layer size of our suggested fusion of the 2DCNN-3 and 2DCNN-4 layers. The main reason for using such a small image size in the input layer is to maintain low computational complexity. As a result, the images can be retrieved in real time.

Although higher resolutions can improve retrieval performance somewhat, these gains must be carefully balanced against the far higher computational cost. Thus, the 28×28 design remains an efficient and reasonable choice, but higher resolutions may be preferable when computational resources allow.

In our experimental results, this small size of images does not affect the retrieval performance of the images, and their attributes, i.e., colour, shape, and texture, are still well preserved. After that, the images are passed on to the next layers of the neural network to extract features. Different samples from both datasets are shown in Figures 3 and 4 (a-j), respectively.



Figure 3. Samples of Corel-5K images

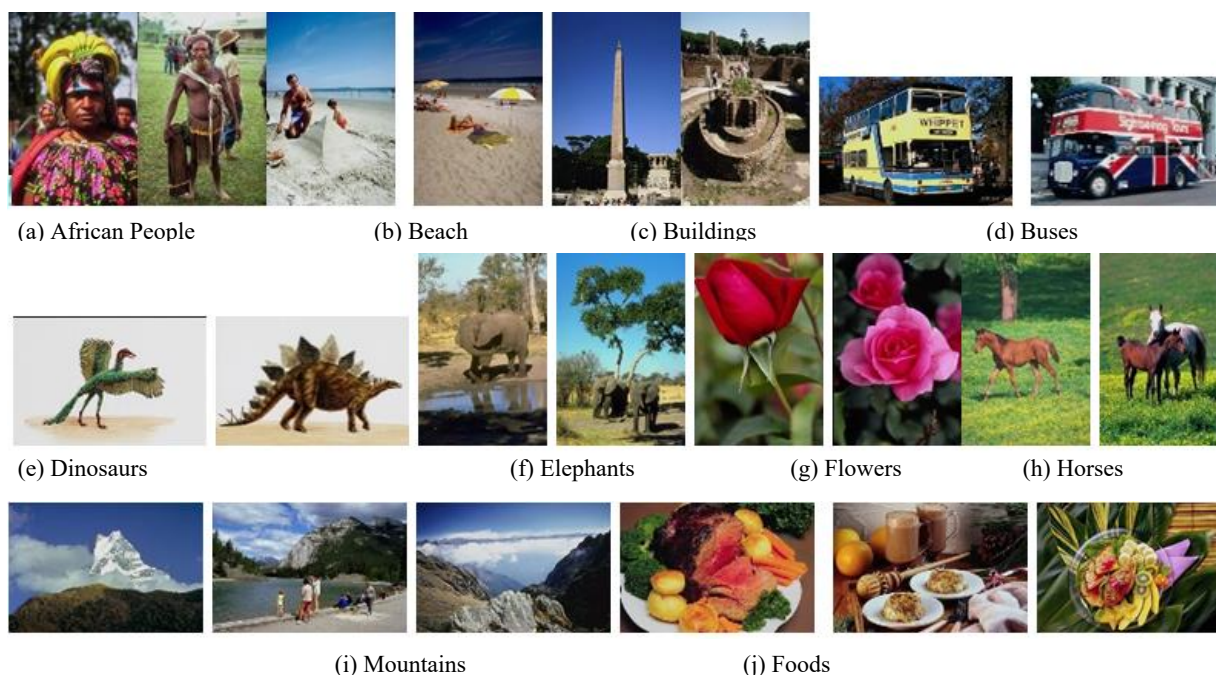


Figure 4. Samples of Corel-1K images

5. Evaluation of Performance

This section illustrates the measurements used to evaluate the suggested model. In other words, Precision (P), Average Precision (AP), and mean Average Precision (mAP) are all important metrics [24].

$$P = \frac{I_M}{T_{IM}} \quad (1)$$

where P is the precision of image retrieval, IM is the number of relevant retrieved images, and TIM is the total number of retrieved images [7].

$$AP = \frac{\sum_{i=1}^n P_i}{T} \quad (2)$$

where AP is the average precision of image retrieval, P_i is the precision of the i image in the class, and T is the total number of images in the class [7].

$$mAP = \frac{\sum_{j=1}^c AP_j}{c} \quad (3)$$

where mAP is the mean average precision of image retrieval, AP_j is the average precision of the j class image, and c is the total number of classes in the database.

6. Results and Discussions

In this section, we present the results of the experimental tests we performed on the models. Additionally, we provide model analysis on filter size configuration. Feature extraction was carried out on a PC on MATLAB with 16 GB, an Intel Core i7 processor (8550U), and a CPU speed of 1.80 GHz. It used Stochastic Gradient Descent with Momentum (SGDM) with momentum = 0.9, weight decay (L2) = 0.0005, initial learning rate = 0.01, number of epochs = 100, data augmentations were random flipping, scaling, cropping, and change of color, mini batch size was 128, and early stopping was done: Stop when validation loss was not changing for five epochs, checkpoint selection: lowest validation loss checkpoint, feature layer: last convolutional layer before global pooling, feature storage format: MAT-files, single-precision floating point. For both Corel-1K and Corel-5K, the datasets are divided into three parts: a training set, a validation set, and a test set, with 70% used for training and 30% for validation and testing. CNN models are trained on the training set. The query set, on the other hand, is taken from the test set. The remaining test images are the gallery set. To avoid data leakage, we keep all query images separate from the gallery set. So, we can measure retrieval performance only on the test set that has been kept aside. No images from the training or validation sets are involved in the evaluation. The method allows us to measure retrieval performance without any bias. This, in turn, helps us figure out which model is performing the best. For

model generalisation or early stopping during training, the data is split into a validation set.

The proposed framework shown in Figure 2 can be described as follows: In the first stage, features or descriptors are learned and extracted by implementing the proposed 2DCNN-3 and 2DCNN-4 supervised learning models to represent images. The names 2DCNN-3 and 2DCNN-4 refer to CNNs with 3 and 4 layers, respectively. The main difference is the number of convolutional layers, which determines how deep the network is and how complex the learnt features may be. Also, regarding training time, 2DCNN-3 has a lower training time than 2DCNN-4, but it can learn more complex features and capture more complex patterns. In general, 2DCNNs like 2DCNN-3 and 2DCNN-4 train more quickly. In our work, multiple layers of CNNs are used to learn image features, and training is faster than with other deeper networks. Moreover, it provides better image retrieval capabilities and significant efficiency gains. Obtained features are stored in an Excel file during the offline stage for later use during the online stage. Hence, query image features are learned and extracted by using the 2DCNN-3 layers and 2DCNN-4 layers. Next, similarity measures are calculated between the learned query image features and those of the database using the Cosine Distance (Cd; see Eq. (4)). The resulting similarity values are sorted in ascending order, and the retrieved images are ranked accordingly in the case of data-level fusion (FCNNM1).

Meanwhile, in the case of score-level fusion (FCNNM2), the resulting similarity values are normalised, the resulting scores are summed, and the resulting list is sorted in ascending order. Finally, a ranked list of images retrieved based on those score values. In this image retrieval process, the image label is not used to measure similarity. So, unsupervised learning is useful when the labels are unknown and the dataset is large. Therefore, the effectiveness of the extracted features can be assessed. Thus, the leave-one-out image strategy was used to evaluate experiments [7]. In other words, the number of image queries will be compared to the remaining images.

$$Cd = 1 - \frac{x_{DI_m} x_{DBI_m}^T}{\sqrt{(x_{DI_m} x_{DI_m}^T)(x_{DBI_m} x_{DBI_m}^T)}} \quad (4)$$

where DI_m is a demanded image feature, and DBI_m is the database image features.

In training a 2D CNN for image classification, the optimiser selection significantly influences generalisation, convergence speed, and performance. In our work, we have used the SGDM optimiser for its strong generalisation, with momentum helping it escape local minima. A momentum of 0.9000 was used, and the initial learning rate was 0.0100. The first image retrieval experiment was conducted on the

Corel-1K standard dataset using the proposed 2DCNN-3-layer model, which produced a learned feature of 10 dimensions. Figure 5 illustrates the precision of each class, while the Top 10–50 retrieved images are used to show the behaviour of each class. The Dinosaurs and Horses classes achieved 100% precision, whereas the Mountains class achieved between 84% and 87% due to the effect of a common object, especially in the Beach class, as well as colour and texture (See Figure 5). Overall, the model achieved a mean average precision of 93%.

In conclusion, CBIR suffers from a semantic problem between high-level meaning and low-level features. Subsequently, retrieval performance will be negatively affected. For instance, a query image of "Mountains" can be confused with "Beach", and vice versa, due to common objects (e.g., the blue sky, people, and water) and overlapping colour and texture. Therefore, the confusion between the two classes is high, even though the images semantically belong to the beach or mountain class (see Figure 4).

In the second image retrieval experiment, we used features extracted from the proposed 2DCNN-4-layer model, which yielded a learned feature vector of 10 dimensions. Figure 6 shows results for precision, and we can see that precision in the Mountains and Food classes improves to 88% and 97%, respectively, compared to the 2DCNN-3-layer model. However, other classes registered some degradation. As shown in Figure 6, the precision for the food and mountain classes increases to 88% and 97%, respectively, compared with the 2DCNN-3 layers model. Other classes, though, showed considerable deterioration. This shows that each model can produce different features and yield different relevant images, although the mean average precision is 93%, the same as that obtained in the first experiment. Therefore, we combined the performance of both models to increase the number of relevant images in the ranked retrieved image list.

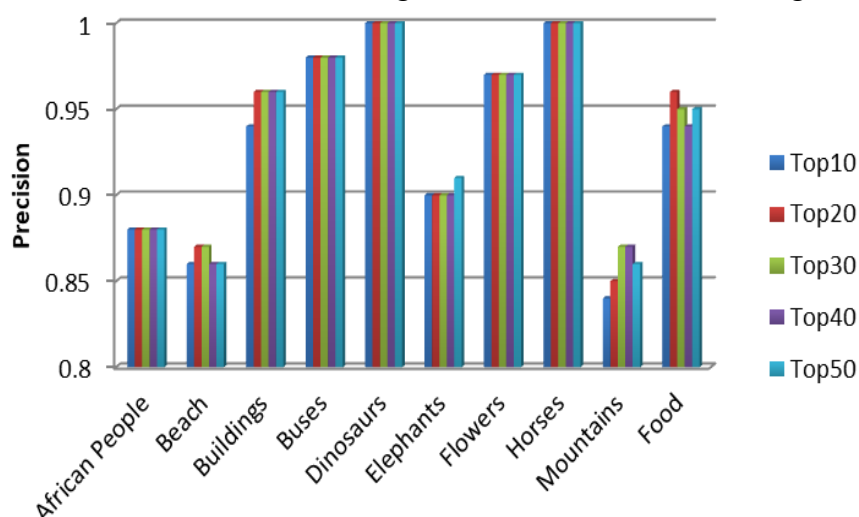


Figure 5. Precision values of the Top 10-50 retrieved images for the proposed 2DCNN-3 layers model

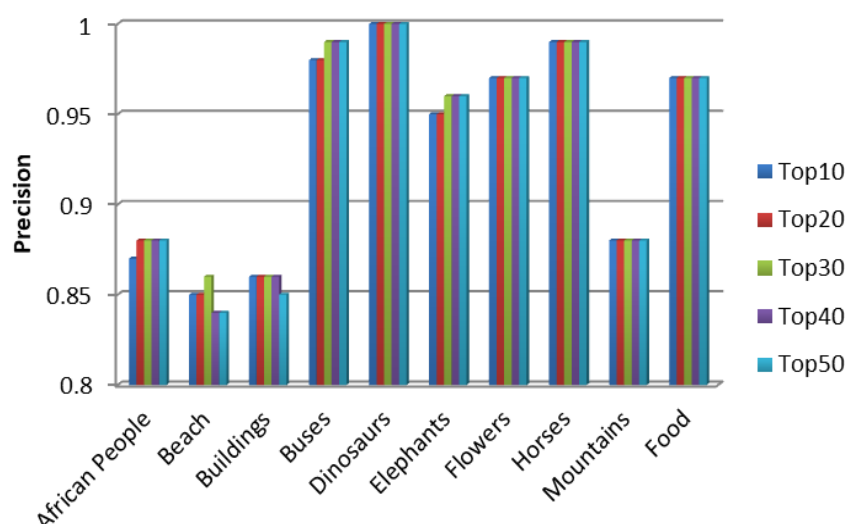


Figure 6. Precision values of the Top 10-50 retrieved images for the proposed 2DCNN-4 layers model

In the original fusion scheme for information retrieval, the goal is to increase the number of relevant documents retrieved. Subsequently, the idea was exploited in image retrieval. There are two main types of fusion methods: data-level and score-level. Data-level fusion involves concatenating two distinct features to represent an image and storing the resulting feature vector in the database. Meanwhile, the score-level fusion means summing the score values (i.e., normalised similarity measures) and then retrieving images based on the resulting score. The score values could be weighted or not [25]. We first tested the proposed 2DCNN models using data-level fusion (FCNNM1), concatenating the learned features from 2DCNN-3 layers (10D) and 2DCNN-4 layers (10D). This means the new concatenated feature is 20-dimensional. FCNNM1 is shown in Figure 2 in orange, where the concatenated features of database images are stored at the Offline stage. Then, a comparison is made between a query image concatenated feature and those of the database using the Cosine distance function. The obtained similarity values are subsequently sorted in ascending order. Next, retrieved images are ranked accordingly. Figure 7 shows that the greatest improvements for the African People, Elephants, and Mountains classes are about 7%, 8%, and 9%, respectively. Meanwhile, the improvement for Food, Buses, and Flowers is about (4, 2, 2) %, respectively. This is evidence that concatenated features increase the discriminability of images.

Secondly, we have tested the proposed 2DCNN models using the score-level fusion (FCNNM2). The steps are illustrated in Figure 2, where features are extracted and learned from the 2DCNN-3 and 2DCNN-4 models; each feature has 10 dimensions. After that, they are stored in the database at the Offline stage. When a query image is requested, features are extracted and learned from each of the above-proposed models, and these features are compared to those in the two databases using the Cosine distance function.

The resulting similarity values are normalised; the resulting scores are summed (Eq. (5)) and then sorted in ascending order for use with retrieved images.

$$S_f = S_{c_1} + S_{c_2} \quad (5)$$

where S_f is the result of fusion score, S_{c_1} is equal to normalised Cd_1 resulting from using 2DCNN-3 layers, and S_{c_2} is the normalised Cd_1 resulting from 2DCNN-4 layers.

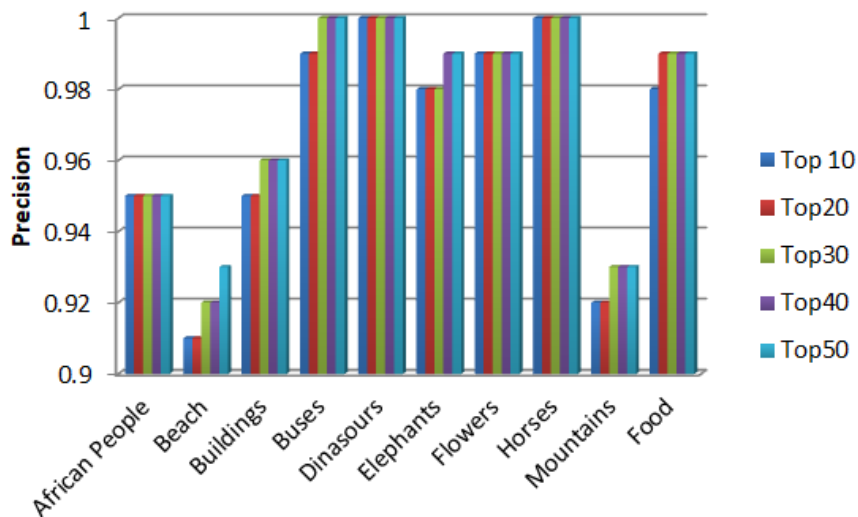


Figure 7. Precision values of the Top 10-50 retrieved images for Data-level fusion

Three scenarios of the fusing method were conducted: without weights, with fixed weights ($w_1=0.6$ and $w_2=0.4$), and with adaptive weights generated by linear regression, as in our previous work [25]. Basically, linear regression finds the best-fit straight line through the points. This line is called a regression line and minimises the sum of squared errors of the estimate. Hence, a straight line, as in Eq. (6), where Z is the input matrix, y is the output, and x (coefficients) is the demand. A straight-line equation was used to estimate weights, where Z denotes dissimilarity values, x denotes weights, and y denotes the fusion vector.

$$Zx + y = 0 \quad Zx = y \quad (6)$$

where Z is the input matrix, y is the output, and x (coefficients) is the demand.

Suppose v and g are dissimilarity values based on two different extracted features and $\bar{v}$ and $\bar{g}$ are means, then the best line model (y) can be computed by:

$$y = slope * v + intercept \quad (7)$$

$$slope = \frac{n \sum vg - \sum v \sum g}{n \sum v^2 - (\sum v)^2} \quad (8)$$

$$intercept = \bar{v} - slope * \bar{g} \quad (9)$$

Now, we can find w weights from the following multiple linear regression instruction in MATLAB: $x = \text{regress}(y, Z)$, where $w_1 = x(1)$ and $w_2 = x(2)$, then use them in the fusion formula as follows:

$$S_f = w1 \times Sc_1 + w2 \times Sc_2 \quad (10)$$

Table 3 presents the results for each fusion method, expressed as mean average precision. In general, the fusion methods increase the mean average precisions by about 4% compared to 2DCNN-3 layers and 2DCNN-4 layers. Specifically, we see that the fixed-weight method achieved a 1% lower mean average precision than the other two fusion methods. Meanwhile, SUM and Regression achieved similar mean average precisions, up to Top 10-50.

Table 3. mAP values for Top 10-50 retrieved images using fusion methods on Corel-1K

Fusion Method	Top 10	Top 20	Top 30	Top 40	Top 50
2DCNN-3 layers	0.93	0.93	0.94	0.93	0.93
2DCNN-4 layers	0.93	0.93	0.94	0.93	0.93
Data level	0.97	0.97	0.97	0.97	0.97
SUM	0.96	0.96	0.97	0.97	0.97
Regression	0.96	0.96	0.97	0.97	0.97
Fixed weights	0.95	0.95	0.96	0.96	0.96

A paired t-test, as shown in Eq. (11), tests the significance of the difference between two hypotheses: the null hypothesis (H_0), where $\mu_1 - \mu_2 = 0$, and the alternative hypothesis (H_1), where $\mu_1 - \mu_2 \neq 0$. The P-value of the test is the probability of observing the test statistic under the null hypothesis. Small values of P indicate rejection of the null hypothesis at the 0.05 significance level [20].

$$t = \frac{\bar{d}}{\sigma/\sqrt{n}} \quad (11)$$

where $\bar{d}$ refers to the sample meaning of the differences between paired observations, σ is the sample standard deviation of the differences, and n is the size of the sample.

In our case, we calculated the paired t-test by regarding the sample mean of the differences between average precision values produced from using the 2DCNN-3 for 10 classes of Corel-1K and those produced from fusing both 2DCNN-3 and 2DCNN-4 models, the sample standard deviation of the differences, and the size of the sample equal to ten, so the degree of freedom (df) is nine because $df = n - 1$. Table 4 shows the obtained results in terms of t , P-value, and standard deviation (σ). As we can see, the P values of 0.0105, 0.0237, 0.0098, 0.030, and 0.030 for Top 10, 20, 30, 40, and 50, respectively, are less than the significance level of 0.05. This indicates a significant improvement after the fusion process (i.e., the null hypothesis is rejected). In addition, Cohen [26], which refers to the effect size (small=0.2, medium=0.5, large=0.8), was computed for the sample and illustrated in Table 4 and defined in Eq. (12). We can interpret the obtained

effect size values, 0.64, 0.66, 0.71, 0.71, and 0.70 for Top 10, 20, 30, 40, 50, respectively, as large because they exceed Cohen's d convention for the medium effect. Hence, Cohen's d also indicates a benefit of the fusing process [26].

$$Cohd = (M1 - M2)/poold Sd \quad (12)$$

where $M1$ is group 1 and $M2$ is group 2 of the sample of the same size [26].

$$poold Sd = \sqrt{\frac{Sd1^2 + Sd2^2}{2}} \quad (13)$$

where $Sd1$ and $Sd2$ are standard deviations of the group1 and group2 respectively.

Table 4. The paired t-test results for the Corel-1K dataset

Measures	Top 10	Top 20	Top 30	Top 40	Top 50
t	3.2225	2.7168	3.2646	3.3739	3.3607
P	0.0105	0.0237	0.0098	0.0082	0.0084
σ	0.029	0.032	0.029	0.030	0.030
Cohen's d	0.6412	0.6637	0.7078	0.7058	0.6965

Furthermore, our proposed models and framework were trained on a larger image database, namely Corel-5K. Table 5 shows retrieval results from the 3-layer and 4-layer CNN models, with the first achieving 89% accuracy on the Top 10-50 retrieved images and the second achieving 1% more. On the other hand, the retrieval results were also improved by about 6% compared to the 3-layer CNN using data-level (FCNNM1) and score-level (FCNNM2-SUM) fusion methods. Hence, there is a positive effect from using the fusion methods.

Table 5. mAP values for the Top 10-50 retrieved images using fusion methods on Corel-5K

Models & Fusion	Top 10	Top 20	Top 30	Top 40	Top 50
CNN 3-layer	0.89	0.89	0.89	0.89	0.89
CNN 4-layer	0.91	0.91	0.91	0.91	0.91
FCNNM1	0.94	0.94	0.95	0.95	0.95
FCNNM2	0.94	0.94	0.95	0.95	0.95

The proposed 2DCNN-3 and 2DCNN-4 models were used to learn features that can distinguish images. Then, the models were incorporated into the CBIR framework (see Figure 2), where a fusion method was used to improve image retrieval accuracy. Experiments showed that both data-level and score-level fusion methods (i.e., can similarly enhance the retrieved image list by increasing the number of relevant images). However, a feature vector has higher dimensionality in a data-level fusion method. Table 6 compares the recent methods mentioned in Section 2 regarding feature dimension and number of layers

for mAP@10 and mAP@20 on retrieved images. The comparison was based on the Corel-1K dataset collected by Wiederhold [23], which is widely used as a standard database. We can see that the proposed FCNNM1, with a 20-dimensional feature vector, three and four layers, and an accuracy of 97% at Top 10. Also, the proposed FCNNM2 has a feature vector length of 10 dimensions, 3 or 4 layers, and an accuracy of 96% at the Top 10 level. This means it is higher than or equal to the recent works of El Mehdi, et al. [17] and Shamsipour, et al. [27], which use 128D and 2323D feature vectors, respectively, with 22 and 50 layers. Moreover, our method FCNNM1 achieves a +3% mAP@20, and FCNNM2 achieves a +2% mAP@20, compared to the work of Shamsipour, et al. [27].

Additionally, 2DCNN-3 layers require less training time than 2DCNN-4; however, 2DCNN-4 can learn more intricate features and identify more complex patterns. Generally, 2DCNNs such as 2DCNN-3 and 2DCNN-4 exhibit faster training times. In our research, we employ multiple layers of CNNs to extract image features, resulting in a training process that is quicker than that of other deeper networks. Furthermore, it enhances image retrieval capabilities and demonstrates considerable efficiency.

Table 6. mAP comparison between the proposed method and recent approaches in the literature for image retrieval using the Corel-1K dataset

Method	Feature Dimension	CNN layers	Top 10	Top 20
Proposed in [14]	128D	22	96%	-
Proposed in [24]	2323D	50	96%	94%
Proposed 2DCNN-3 layers	10D	3	93%	93%
Proposed 2DCNN-4 layers	10D	4	93%	93%
Proposed FCNNM1	20D	3&4	97%	97%
Proposed FCNNM2	10D	3&4	96%	96%

In conclusion, this work uses a 2D CNN with grouping into three- and four-layer groups to produce a user-friendly, compact design suitable for applications requiring fast image retrieval. Overfitting and vanishing gradients are problems that current DCNN models, such as ResNet and EfficientNet, can overcome. These models have high processing and memory requirements, which might be limited. A comparison of our proposed 2DCNN with three- and four-layer models and other complex DCNN models indicates that the proposed model achieves superior performance in image retrieval on the Corel-1K and Corel-5K datasets. The experimental results obtained in this study demonstrate that the proposed model requires fewer parameters and shorter training time. To summarise, comparing our proposed 2DCNN with

3 and 4 layers and a deeper CNN model, our model performs very well on image retrieval. Our experiment results also show that our model needs fewer parameters and less training time.

7. Conclusions and Future Works

This paper aimed to develop a CBIR algorithm that improves the effectiveness and efficiency of image retrieval by enhancing image representation (i.e., color, texture, and shape features), an issue that remains challenging in the CBIR field. Therefore, two CNN models with two dimensions, 2DCNN-3 layers and 2DCNN-4 layers, are proposed and implemented to extract image features. The image retrieval accuracy for Corel-1K and Corel-5K was 93% and 91%, respectively. Furthermore, the proposed algorithm exploits fusion by integrating the performance of both 2DCNN-3 and 2DCNN-4 models to increase the number of relevant images in the retrieved list. The experimental results indicated that each model could yield different relevant images, which can be integrated to improve image retrieval accuracy. Consequently, performance on the Corel-1K and Corel-5K datasets improves significantly to 97% and 95%, respectively. Thus, a hybrid image retrieval method is proposed that combines supervised and unsupervised learning techniques, along with a fusion scheme that raises relevant images in the ranked retrieval list. Our models, 2DCNN-3 and 2DCNN-4, achieve promising results and are implemented on both Corel-1K and Corel-5K. However, there are some weaknesses due to common objects and visual content across different classes, which remain a challenge in the Corel-1K and Corel-5K datasets, as well as in similar diverse image databases. For instance, blue sky, green grass, trees, people, dogs, wolves, mountains, etc., are common across various classes.

While the results of this study have been encouraging, it must address several limitations to improve future research. The first limitation concerns the datasets used for evaluation. In addition, the two sets of data used to evaluate the algorithm's performance were both Corel-1K and Corel-5K, which are relatively small and no longer exist. Therefore, future research should evaluate the effectiveness of the proposed algorithm using larger, more recent datasets, such as Caltech-256, and other databases, such as DeepFashion. Furthermore, in addition to evaluating the proposed algorithm against only two baseline methods, additional stakeholders must include baseline methods that represent a broader range of methodologies to provide a fair comparison and better delineate where it fits within the existing literature. Also, this work is centred on the development of efficient 2DCNNs with the fusion of 3 and 4 layers for feature extraction, and thus

does not consider large, modern pre-trained baselines such as ResNet-50+GAP/GeM or CLIP ViT-B/32. Although these models, in all probability, could deliver better retrieval performance, they also require substantial computation and resources. Finally, there should be comprehensive analyses of the various efficiency metrics (including the number of parameters, FLOPs, memory consumption, query response time, etc.) associated with each of the methods; this will allow for a more complete understanding of how to apply this method in practice and to develop an estimate of its computational expense.

Acknowledgment: The author gratefully acknowledges the support of the College of Engineering, Mustansiriyah University, Baghdad, Iraq, for providing institutional support during this study.

Data Availability Statement: The datasets analysed during the current study are publicly available benchmark datasets. Additional data generated during this study are available from the corresponding author on reasonable request.

Funding: The authors declare that no funding was received for this research.

Declaration of Competing Interest: The authors declare that they have no known competing interests.

Research Involving Human/Animal Participants: This article does not involve any studies conducted by the authors on animals or human participants.

References

- [1] H. Du, H. Al-Jubouri, and H. Sellaheewa, "Effectiveness of image features and similarity measures in cluster-based approaches for content-based image retrieval," presented at the Mobile Multimedia/Image Processing, Security, and Applications 2014, 2014, doi: <https://doi.org/10.1117/12.2057721>
- [2] A. Latif *et al.*, "Content-Based Image Retrieval and Feature Extraction: A Comprehensive Review," *Mathematical Problems in Engineering*, vol. 2019, p. 9658350, 2019, doi: <https://doi.org/10.1155/2019/9658350>.
- [3] R. Vishraj, S. Gupta, and S. Singh, "A comprehensive review of content-based image retrieval systems using deep learning and hand-crafted features in medical imaging: Research challenges and future directions," *Computers and Electrical Engineering*, vol. 104, p. 108450, 2022, doi: <https://doi.org/10.1016/j.compeleceng.2022.108450>.
- [4] F. Imdad, S. U. Haq, I. Rasool, M. Wasimuddin, and M. O. Farooq, "Content-Based Image Retrieval Established on Deep and Handcrafted Features," *The Sciencetech*, vol. 5, no. 2, pp. 1-12, 2024. [Online]. Available: <https://journals.qurtuba.edu.pk/ojs/index.php/tst/article/view/85932>.
- [5] M. Singh and M. K. Singh, "Content based medical image retrieval using deep learning and handcrafted features in dimensionality reduction framework," *Computers and Electrical Engineering*, vol. 127, p. 110581, 2025, doi: <https://doi.org/10.1016/j.compeleceng.2025.110581>.
- [6] S. M. Mahmoud, H. A. S. Al-Jubouri, and T. E. Abdoulabbas, "Chest radiographs images retrieval using deep learning networks," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 3, pp. 1358-1369, 2022, doi: <https://doi.org/10.11591/eei.v11i3.3478>.
- [7] A. Irtaza, M. A. Jaffar, E. Aleisa, and T.-S. Choi, "Embedding neural networks for semantic association in content based

- image retrieval," *Multimedia Tools and Applications*, vol. 72, no. 2, pp. 1911-1931, 2014, doi: <https://doi.org/10.1007/s11042-013-1489-6>.
- [8] R. Sathya and B. Saleena, "A survey on content based image retrieval using convolutional neural networks," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 5, pp. 7387-7396, 2020, doi: <https://doi.org/10.30534/ijatcse/2020/70952020>.
- [9] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "Deep Image Retrieval: Learning Global Representations for Image Search," presented at the Computer Vision – ECCV 2016, 2016, doi: https://doi.org/10.1007/978-3-319-46466-4_15
- [10] A. Alzu'bi, A. Amira, and N. Ramzan, "Content-based image retrieval with compact deep convolutional features," *Neurocomputing*, vol. 249, pp. 95-105, 2017, doi: <https://doi.org/10.1016/j.neucom.2017.03.072>.
- [11] M. Tzelepi and A. Tefas, "Deep convolutional learning for Content Based Image Retrieval," *Neurocomputing*, vol. 275, pp. 2467-2478, 2018, doi: <https://doi.org/10.1016/j.neucom.2017.11.022>.
- [12] V. K. Vishnoi, K. Kumar, B. Kumar, S. Mohan, and A. A. Khan, "Detection of Apple Plant Diseases Using Leaf Images Through Convolutional Neural Network," *IEEE Access*, vol. 11, pp. 6594-6609, 2023, doi: <https://doi.org/10.1109/ACCESS.2022.3232917>.
- [13] A. A. Khan *et al.*, "Detection of Omicron Caused Pneumonia from Radiology Images Using Convolution Neural Network (CNN)," *Computers, Materials & Continua*, vol. 74, no. 2, pp. 3743-3761, 2023, doi: <https://doi.org/10.32604/cmc.2023.033924>.
- [14] L. Pipanmekaporn, S. Kamonsantiroj, C. Ratanavilisagul, and S. Prasomphan, "Spatial Pyramid Attention Enhanced Visual Descriptors for Landmark Retrieval," *Journal of Image and Graphics*, vol. 11, no. 4, pp. 359-366, 2023, doi: <https://doi.org/10.18178/joig.11.4.359-366>.
- [15] A. Radford *et al.*, "Learning Transferable Visual Models From Natural Language Supervision," presented at the Proceedings of the 38th International Conference on Machine Learning, 2021, doi: <https://doi.org/10.48550/arXiv.2103.00020>
- [16] M. Oquab *et al.*, "DINOv2: Learning Robust Visual Features without Supervision," doi: <https://doi.org/10.48550/arXiv.2304.07193>.
- [17] E. A. E. Mehdi, B. Latifa, and S. Hassan, "A Mighty Image Retrieval Descriptor Based on Machine Learning and Gaussian Derivative Filter," *International Journal of Electrical and Computer Engineering Systems*, vol. 15, no. 5, pp. 427-435, 2024, doi: <https://doi.org/10.32985/ijeces.15.5.5>.
- [18] T. M. M. Aung and A. A. Khan, "Enhanced U-Net with Attention Mechanisms for Improved Feature Representation in Lung Nodule Segmentation," *Current Medical Imaging*, vol. 21, p. e15734056386382, 2025, doi: <https://doi.org/10.2174/0115734056386382250902064757>.
- [19] A. A. Khan, M. Driss, W. Boulila, G. A. Sampedro, S. Abbas, and C. Wechtaisong, "Privacy Preserved and Decentralized Smartphone Recommendation System," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 4617-4624, 2024, doi: <https://doi.org/10.1109/TCE.2023.3323406>.
- [20] A. Ross and V. L. Willson, *Basic and Advanced Statistical Tests: Writing Results Sections and Creating Tables and Figures*. Rotterdam: SensePublishers, 2018, doi: <https://doi.org/10.1007/978-94-6351-086-8>
- [21] J. P. Bharadiya, "Convolutional Neural Networks for Image Classification," *International Journal of Innovative Science and Research Technology*, vol. 8, no. 5, pp. 673-677, 2023, doi: <https://doi.org/10.5281/zenodo.7952031>.
- [22] A. Ghourabi, M. A. Mahmood, and Q. M. Alzubi, "A Hybrid CNN-LSTM Model for SMS Spam Detection in Arabic and English Messages," *Future Internet*, vol. 12, no. 9, p. 156, 2020, doi: <https://doi.org/10.3390/fi12090156>.
- [23] 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, doi: <https://doi.org/10.1109/34.955109>.
- [24] H. A. Al-Jubouri and S. M. Mahmmod, "A comparative analysis of automatic deep neural networks for image retrieval,"

- TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 19, no. 3, pp. 858-871, 2021, doi: <https://doi.org/10.12928/TELKOMNIKA.v19i3.18157>.
- [25] H. A. Al-Jubouri, "Integration Colour and Texture Features for Content-Based Image Retrieval," *International Journal of Modern Education and Computer Science*, vol. 12, no. 2, pp. 10-18, 2020, doi: <https://doi.org/10.5815/ijmecs.2020.02.02>.
- [26] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*. New York: Routledge, 2013, doi: <https://doi.org/10.4324/9780203771587>
- [27] G. Shamsipour, S. Fekri-Ershad, M. Sharifi, and A. Alaei, "Improve the efficiency of handcrafted features in image retrieval by adding selected feature generating layers of deep convolutional neural networks," *Signal, Image and Video Processing*, vol. 18, no. 3, pp. 2607-2620, 2024, doi: <https://doi.org/10.1007/s11760-023-02934-z>.