Content based Image Retrieval using Fine-tuned Deep Features with Transfer Learning

Meqdam A. Mohammed, Zakariya A. Oraibi*, Mohammed Abdulridha Hussain
Department of Computer Science, College of Education for Pure Sciences, University of Basrah
Basrah, 61004, Iraq
*Corresponding Author Email: zakaria_au@uobasrah.edu.iq

***Abstract*—This paper introduces a deep learning approach to efficiently retrieve images using a robust deep features extracted from VGG-19 architecture. Our work involves fine-tuning this pre-trained network to adapt it to our dataset by replacing the final layers, we can then train the network so that it learns feature representation for the content based image retrieval task. After applying transfer learning, then, the new network is trained on our dataset after performing augmentation to the set of training images to increase the number of images in order to improve the training accuracy. Augmentation techniques involve using shifting, shearing, and flipping to the original input images. Finally, features are extracted from the 'fc7' layer that has 4096 bins for each input image. Euclidean distance is applied to calculate the closest distance between the query image and the features database. Experiments are conducted on a standard dataset called Corel-1k with 1000 images and 10 different categories. Results show that our approach generates high precision accuracy that outperforms traditional image retrieval methods and is in level with deep learning based methods.**

***Index Terms*—Deep Learning, CBIR, Image Features, Transfer Learning.**

I. Introduction

Storing, retrieving, manipulating a specific image from a large scale repository of information is considered a very difficult machine learning task due to the growing increase in the number of digital data. A content-based image retrieval system (CBIR) uses the content of an image to retrieve images from datasets having similar visual representations. In this case, visual representation depicts the color, texture, or shape of an image. A typical CBIR system works by transforming training images into corresponding feature vectors through techniques that can be both hand-crafted and based on deep learning approaches [1]. A query image is then fed to the system where its feature vector is compared against the feature database and similar images are retrieved based on the similarity scores. Factors including the appropriate selection of image representation methods, classifiers, and similarity measures are crucial for the success of image retrieval systems [2].

The main parts in any CBIR system are the features extracted from the images and the choice of similarity measure. As a result, the majority of researchers focus on these two parts when designing a CBIR model. These features include handcrafted features which require a specific combination of texture, color, and shape features in order to generate better image retrieval [3]–[8]. In addition, deep features that rely on Convolutional Neural Networks (CNNs) to enhance the performance of a CBIR system since they provide a generic image representation [9], [10]. The performance of CBIR systems using deep features usually outperforms handcrafted features systems due to their powerful ability to represent the input images. The second part in CBIR system is the choice of similarity measure. Researchers applied many techniques to measure the similarity between two feature vectors including Manhattan distance, Euclidean distance, Jaccard similarity Canberra distance, and Cosine similarity [2]. The majority of researches use these metrics directly without making any changes and focus on developing framework of features for strong CBIR systems. Deep learning approaches applied for CBIR in the literature include the work of Lin et al. [11] in which CNNs where used to create a framework that uses binary hash codes with fine-tuning. Hamming distance was applied in their approach. Hamreras et al. [12] proposed using CNN features to represent input images by extracting features from the pre-trained Alexnet using transfer learning. An approach called CRB-CNN was proposed by Alzu'bi et al. [13] which uses two CNNs in parallel for feature extraction from specific layers to create a compact feature vector with Euclidean distance employed for image retrieval. Shah et al. [14] proposed using CNN features for CBIR along with Euclidean distance to retrieve images. Furthermore, Pathak et al. [15] introduced a feature fusion system that consists of deep and handcrafted features for image retrieval. An improved version of DarkNet-53 was used along with shape and HSI color space. The work of Desai et al. also involved using a hybrid model of deep learning features using VGG-16 and classification using SVM for fast and robust image retrieval [16]. The majority of the approaches in the literature relied on pre-trained deep learning architectures with powerful features extracted from the final network layers.

In this paper, we introduce an approach for image retrieval using robust features extracted from a pre-trained VGG-19 architecture. Our technique starts by training the network on the set of training images of our database. Hence, the network will be familiar with the intended type of images so that features extracted from it will be a powerful representation of the image objects. Before training, transfer learning is applied by replacing the final three layers of VGG-19 with the newly fine-tuned layers which take 10 classes instead of 1000. Then, features are extracted from the 'fc7' layer with 4096 dimensions for each input image. All these features are used for image retrieval. Euclidean distance is applied on the

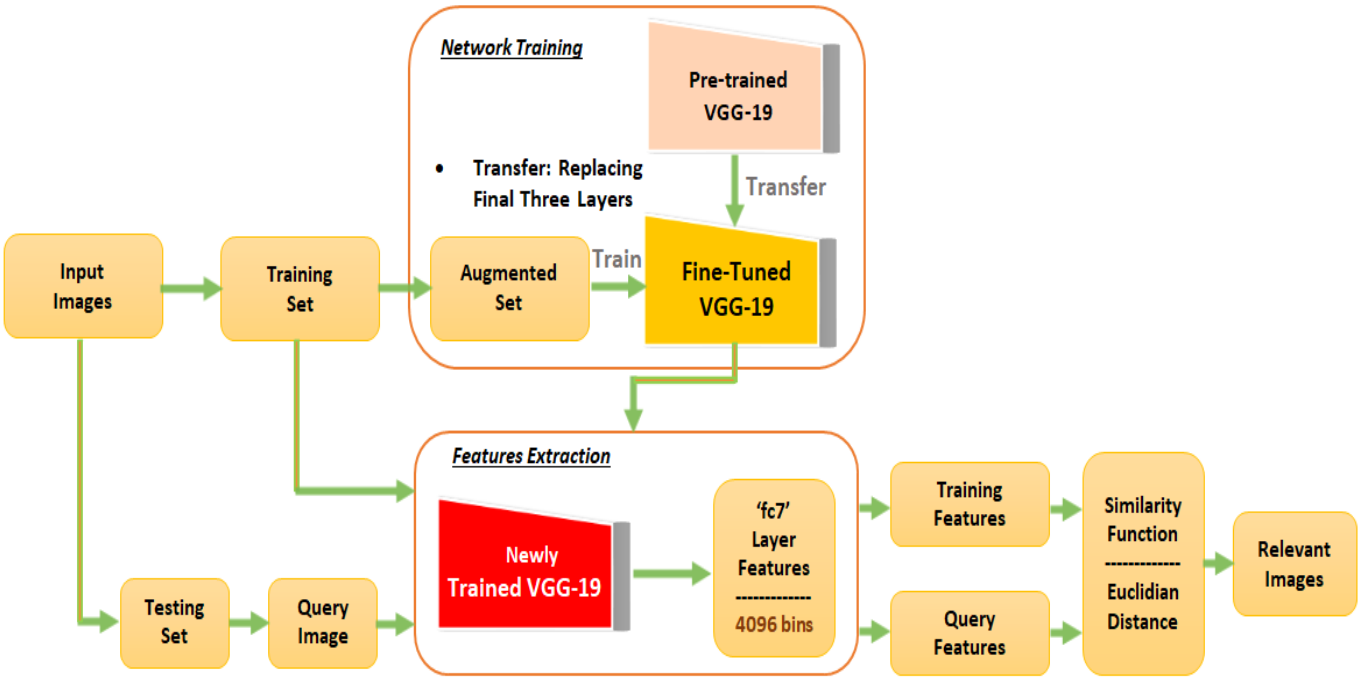

Fig. 1. The proposed image retrieval scheme using VGG-19 features with Euclidean similarity function.

query image and the set of features extracted from the rest of images that represent the database features. We used these deep features successfully in previous image classification task which proved to be reliable to improve the mean class accuracy and achieve superior performance [17].

The rest of the paper is organized as follows: Section II introduces our methodology to retrieve images using deep features. Section III details the results of experiments. Section IV concludes the paper and highlights future work.

II. Methodology

The use of CBIR ranges from portable devices to broad medical applications, satellite applications, etc. It relies mainly on analyzing the content of the given image which is more desirable than relying on associated information like tags. Our proposed architecture for a strong CBIR system is shown in Figure 1. The main steps of the model include network training, features extraction, and similarity metric computation.

A. Network Training

The original dataset used to test our model must be split first into two sets: training and testing. Hence, training set images are used to train the VGG-19 model after performing the transfer learning. However, since the sample images provided for training are low, we need to augment this set of images before sending them for training. Many data augmentation techniques are used to enlarge datasets and make network training more efficient. In our work, we derived new images from the original datasets by making small changes such as translation, flipping, and shearing. Since the problem is to match images, we did not rely on generating totally new images (synthetic data) to train our network. After performing augmentation, we increased the number of images from 900 samples to 3600 samples. Those newly generated images are used only to train the model. During the training stage, we used 50 epochs to get a good convergence of both model accuracy and model loss as shown in Figure 2.

B. Deep Features Extraction

The VGG-19 CNN consists of 19 convolutional layers and it accepts an input image of size $224 \times 224 \times 3$ [18]. In addition, it comprises of a receptive field of 3×3 with a convolution stride of 1 pixel and 2 pixels in the later layers. Furthermore, there are five maximum pooling layers and three Fully-Connected (FC) layers at the end. The first two FC lyaers have 4096 channels each, the third one has 1000 channels since the network is trained on ImageNet dataset which consists of 1000 classes.

In this paper, we fine-tuned VGG-19 architecture by applying transfer learning to train it on our data which consists of 10 classes. As shown in Figure 1, the augmented dataset is fed to the fine-tune architecture after that the newly trained VGG-19 architecture is used to extract deep features. After successfully training the network, each image from the original training set is fed to the network and features are extracted from 'fc7' layer. Hence, each image is transformed into a 1D feature vector of 4096 bins to be used in the next step which is the similarity metric computation between the query feature and features database.

It is worthy to mention that these features are considered global features and were used successfully in many computer vision applications including texture classification [19]. In addition, we found during experiments that the time consumed to compute the similarity measure between two feature vectors

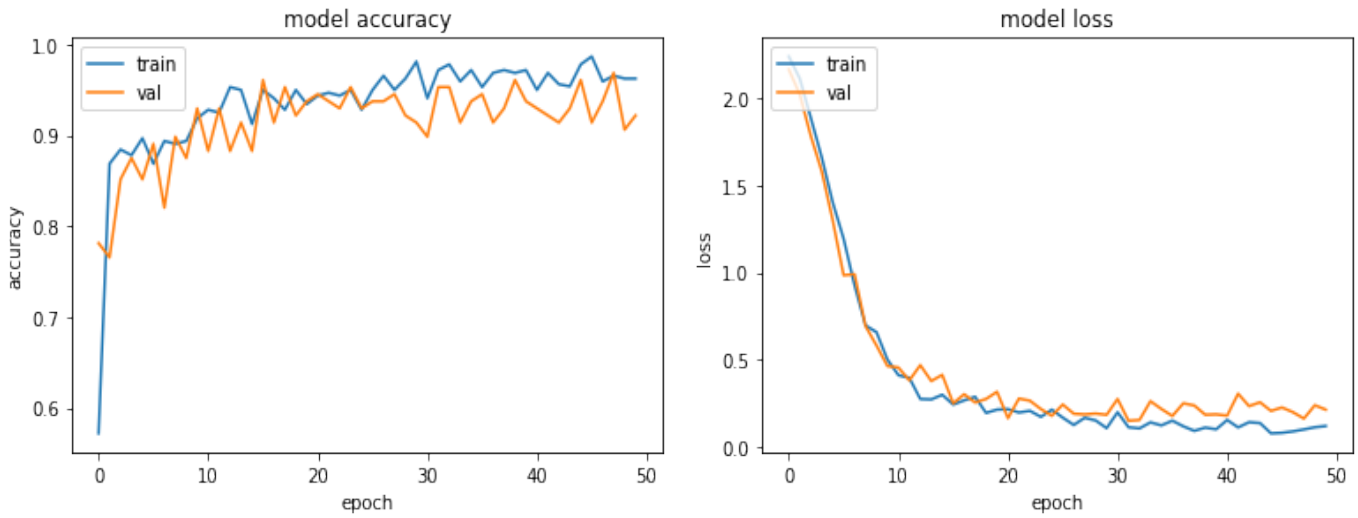

Fig. 2. Model accuracy vs model loss after 50 epochs using fine-tuned VGG-19 architecture.

using these global features is very trivial. Hence, once we get the trained fine-tuned model, we can store it and use it whenever we need to extract features from any dataset.

C. Similarity Metric Computation

Once features are extracted for both training features (features database) and the query image, the next step is to preform features matching. Hence, a similarity technique is used to measure these attributes. In this paper, we used Euclidean distance similarity function between the query features and database features. The formula used to calculate the Euclidean distance is as shown below:

$$Ed\,(p,q) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2} \tag{1}$$

q and p are the corresponding features of the query image and one of the database images.

III. Experimental Results

A. Corel-1k Dataset

In the experiments, we used Corel-1k dataset which consists of 10 classes: Beaches, Bus, Dinosaurs, Elephants, Flowers, Foods, Horses, Monuments, Mountains, and People. Each class consists of 100 images divided into 90 images for training and 10 for testing stored in a JPEG format. Before we apply VGG-19 architecture, images are resized to $224 \times 224 \times 3$. Figure 3 shows sample images of the dataset. The 10 images provided for testing are used as query images in each category.

B. Results of Experiments

The results of applying our CBIR approach on Corel-1k dataset is introduced in this section. In addition, a comparison is made between our results and the results of state-of-the-art methods applied on the same dataset. All experiments and analysis related to our model are conducted on the Google Colaboratory platform with a Tesla GPU. The metrics used to measure the performance of our approach are Precision (P) and the Recall (R) which are widely used in this field. Both metrics are calculated using the equations below:

$$P = \frac{Number\ of\ Relevant\ Images\ Retrieved}{Total\ Number\ of\ Retrieved\ Images} \tag{2}$$

$$R = \frac{Number\ of\ Relevant\ Images\ Retrieved}{Total\ Number\ of\ Relevant\ Images} \tag{3}$$

The results of our proposed approach are shown in Table I using the precision metric. As we can observe, in each category, the precision is high achieving 1.0 in six categories out of ten. Overall, the average precision of our approach is 0.929. In terms or recall, it was computed by dividing the total number of correctly retrieved images by 80 since we are retrieving 10 images. Figure 4 shows the recall result for each class. The classes with a precision of 1.000 get a recall of 0.125. The average recall of all classes is 0.118.

Figure 5 shows the images retrieved after applying our approach using two query images from the Dinosaurs and Foods classes.

Table II shows a comparison in terms of precision between our method and state-of-the-art approaches. Ramanjaneyulu et al. [20] used VGG-16 model to extract deep features and create the feature database as well as using Euclidean distance between the query image features and the features database. Their approach achieved high precision of 94.58%. Li et al. [21] proposed a hybrid method of spatial convolutional neural network and extreme machine learning and applied the framework to Corel-1k dataset and achieved a precision of 79.70%. Other methods that rely on local features like texture and color include the work of Pavithra et al. [22] which achieved a precision of 83.22% as well as the work of Charles et al. [7] which achieved a precision of 76.5%. It is worthy to mention that Hamreras et al. [12] extracted features using both

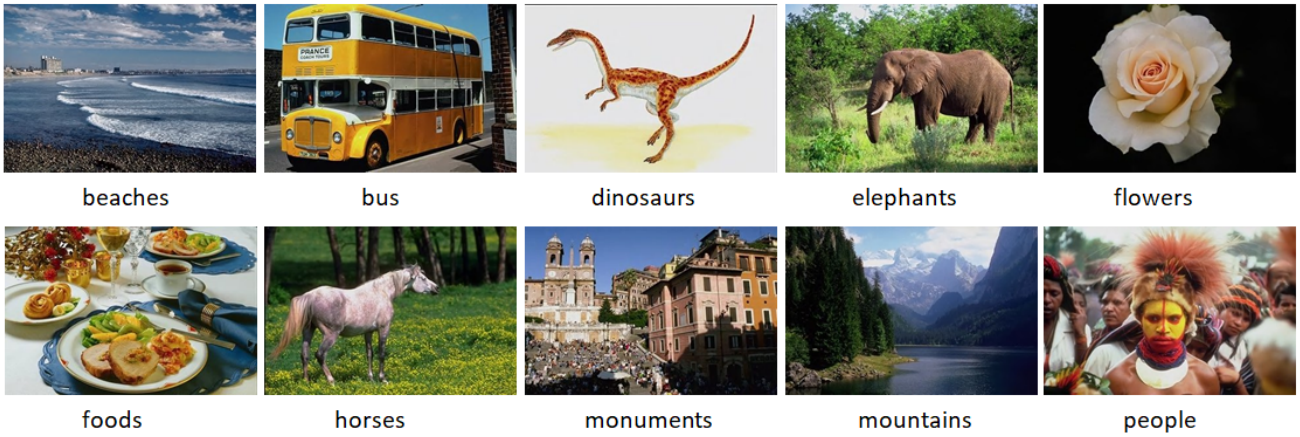

Fig. 3. Sample images of Core-1k database.

TABLE I
PERFORMANCE OF OUR APPROACH MEASURED USING PRECISION. THE NUMBER OF RETRIEVED IMAGES PER QUERY IS 10.

Category	Precision
Beaches	0.710
Bus	1.000
dinosaurs	0.980
Elephants	1.000
Flowers	1.000
Foods	0.680
Horses	1.000
monuments	1.000
Mountains	1.000
Africa	0.920
Overall	0.929

CNN and transfer learning with AlexNet pre-trained network and achieved a precision of 0.9583. In our work, we achieved a high precision of 92.90% using only deep features extracted from VGG-19 architecture.

The number of images retrieved per query varies between systems in the literature. While most of the systems retrieve 20 images per query, the recent work of sikandar et al. [27] retrieved only 5 images and reported a precision rate of 100%. In our paper, we retrieved 10 images.

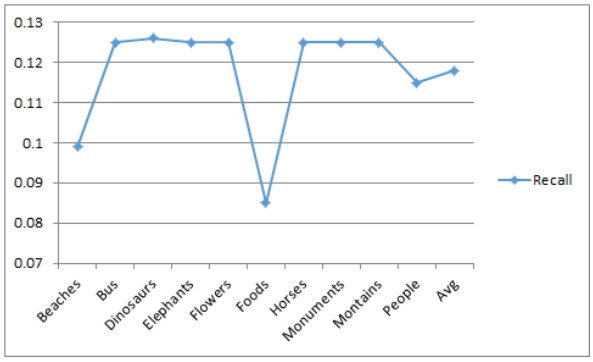

Fig. 4. Recall results for each class.

TABLE II
COMPARISON BETWEEN OUR METHOD AND THE STATE-OF-THE-ART APPROACHES.

Method	Year	Precision (%)
DDBTC [5]	2015	79.20
LMCTP [7]	2016	76.50
SCNN-ELM [21]	2018	79.70
CM-LBP-CED [22]	2018	83.22
Singh et al. [24]	2020	92.2
Salih et al. [26]	2023	86.06
Proposed	-	**92.90**

IV. CONCLUSIONS

In this paper, we present a simple frame work for content-based image retreiaval using a powerful features extracted from VGG-19 architecture. We start by using the subset of training images of our databse to train the VGG-19 network. Before training is applied, images are to be augmented in order to increase the number of training samples, then, transfer learning is applied by changing the final three VGG-19 layers with the newly trained layers. Finally, features are extracted from the 'fc7' layer which has a dimension of 4096 bins for each image sample. These features will be used during the retrieval stage which involves applying the Euclidean distance between the query image and the sample features database. Results show that our approach generates superior precision figures in comparison with state-of-the-art methods. As a result, these deep features extracted from the fine-tuned VGG-19 architecture has the potential to further improve the precision when combined with other robust features. Furthermore, our results outperformed traditional CBIR approaches that rely on local features.

The future work will include designing a new deep learning architecture and train it from scratch to efficiently recognize various image classes. After that, deep features will be extracted from the new architecture and used to retrieve images correctly. In addition, other distance measures will be used like Manhattan distance and Cosine distance. Moreover, Other datasets will be used to test the system and to provide more comprehensive results.

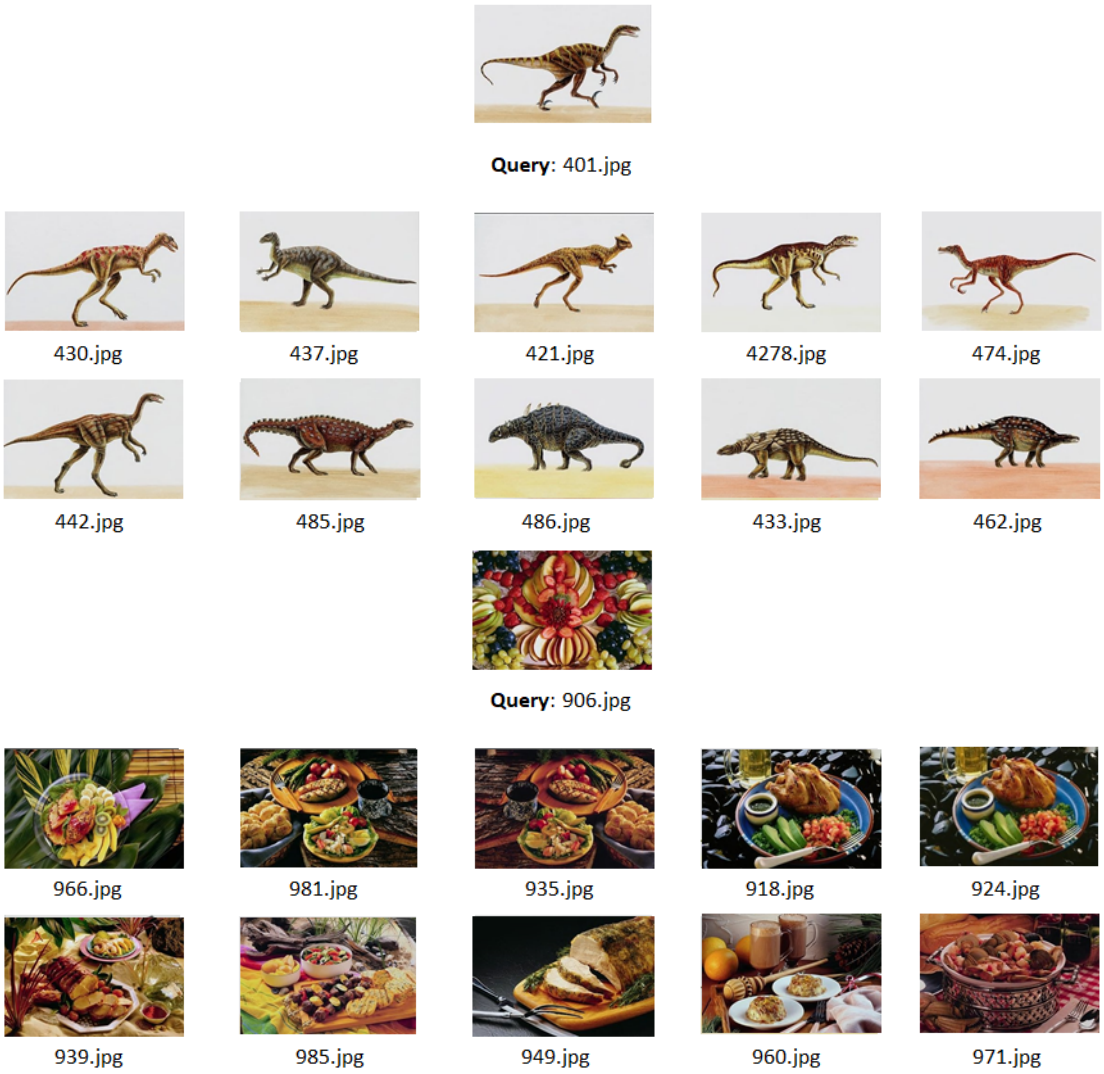

Fig. 5. Two query images from the Dinosaurs and Foods classes and the corresponding retrieved images using our framework.

REFERENCES

[1] Z. Tianyu, M. Zhenjiang, and Z. Jianhu, "Combining cnn with hand-crafted features for image classification," in *2018 14th ieee international conference on signal processing (icsp)*, pp. 554–557, IEEE, 2018.

[2] 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, p. e0274764, 2022.

[3] E. Walia and A. Pal, "Fusion framework for effective color image retrieval," *Journal of Visual Communication and Image Representation*, vol. 25, no. 6, pp. 1335–1348, 2014.

[4] J. Annrose *et al.*, "An efficient image retrieval system with structured query based feature selection and filtering initial level relevant images using range query," *Optik*, vol. 157, pp. 1053–1064, 2018.

[5] J.-M. Guo, H. Prasetyo, and N.-J. Wang, "Effective image retrieval system using dot-diffused block truncation coding features," *IEEE Transactions on Multimedia*, vol. 17, no. 9, pp. 1576–1590, 2015.

[6] X.-Y. Wang, B.-B. Zhang, and H.-Y. Yang, "Content-based image retrieval by integrating color and texture features," *Multimedia tools and applications*, vol. 68, no. 3, pp. 545–569, 2014.

[7] Y. R. Charles and R. Ramraj, "A novel local mesh color texture pattern for image retrieval system," *AEU-International Journal of Electronics and Communications*, vol. 70, no. 3, pp. 225–233, 2016.

[8] Z. A. Oraibi, M. Irio, A. Hafiane, and K. Palaniappan, "Texture classification using multiple local descriptors," in *2017 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pp. 1–7, IEEE, 2017.

[9] C. Bai, L. Huang, X. Pan, J. Zheng, and S. Chen, "Optimization of deep convolutional neural network for large scale image retrieval," *Neurocomputing*, vol. 303, pp. 60–67, 2018.

[10] Q. Zhang, D. Liu, and H. Li, "Deep network-based image coding for simultaneous compression and retrieval," in *2017 IEEE International Conference on Image Processing (ICIP)*, pp. 405–409, IEEE, 2017.

[11] K. Lin, H.-F. Yang, J.-H. Hsiao, and C.-S. Chen, "Deep learning of binary hash codes for fast image retrieval," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 27–35, 2015.

[12] S. Hamreras, R. Benítez-Rochel, B. Boucheham, M. A. Molina-Cabello, and E. López-Rubio, "Content based image retrieval by convolutional neural networks," in *International Work-Conference on the Interplay Between Natural and Artificial Computation*, pp. 277–286, Springer, 2019.

[13] A. Alzu'bi, A. Amira, and N. Ramzan, "Content-based image retrieval with compact deep convolutional features," *Neurocomputing*, vol. 249, pp. 95–105, 2017.

[14] A. Shah, R. Naseem, S. Iqbal, M. A. Shah, *et al.*, "Improving cbir accuracy using convolutional neural network for feature extraction," in *2017 13th International Conference on Emerging Technologies (ICET)*, pp. 1–5, IEEE, 2017.

[15] D. Pathak and U. Raju, "Content-based image retrieval using feature-fusion of groupnormalized-inception-darknet-53 features and handcraft features," *Optik*, vol. 246, p. 167754, 2021.

[16] P. Desai, J. Pujari, C. Sujatha, A. Kamble, and A. Kambli, "Hybrid

approach for content-based image retrieval using vgg16 layered architecture and svm: An application of deep learning," *SN Computer Science*, vol. 2, no. 3, pp. 1–9, 2021.

[17] Z. A. Oraibi, H. Yousif, A. Hafiane, G. Seetharaman, and K. Palaniappan, "Learning local and deep features for efficient cell image classification using random forests," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, pp. 2446–2450, IEEE, 2018.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[19] M. Cimpoi, S. Maji, and A. Vedaldi, "Deep filter banks for texture recognition and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3828–3836, 2015.

[20] K. Ramanjaneyulu, K. V. Swamy, and C. S. Rao, "Novel cbir system using cnn architecture," in *2018 3rd International conference on inventive computation technologies (ICICT)*, pp. 379–383, IEEE, 2018.

[21] D. Li, X. Qiu, Z. Zhu, and Y. Liu, "Criminal investigation image classification based on spatial cnn features and elm," in *2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 2, pp. 294–298, IEEE, 2018.

[22] L. Pavithra and T. S. Sharmila, "An efficient framework for image retrieval using color, texture and edge features," *Computers & Electrical Engineering*, vol. 70, pp. 580–593, 2018.

[23] A. Sarwar, Z. Mehmood, T. Saba, K. A. Qazi, A. Adnan, and H. Jamal, "A novel method for content-based image retrieval to improve the effectiveness of the bag-of-words model using a support vector machine," *Journal of Information Science*, vol. 45, no. 1, pp. 117–135, 2019.

[24] S. Singh and S. Batra, "An efficient bi-layer content based image retrieval system," *Multimedia Tools and Applications*, vol. 79, no. 25-26, pp. 17731–17759, 2020.

[25] O. Sikha and K. Soman, "Dynamic mode decomposition based salient edge/region features for content based image retrieval.," *Multim. Tools Appl.*, vol. 80, no. 10, pp. 15937–15958, 2021.

[26] S. F. Salih and A. A. Abdulla, "An effective bi-layer content-based image retrieval technique," *The Journal of Supercomputing*, vol. 79, no. 2, pp. 2308–2331, 2023.

[27] S. Sikandar, R. Mahum, and A. Alsalman, "A novel hybrid approach for a content-based image retrieval using feature fusion," *Applied Sciences*, vol. 13, no. 7, p. 4581, 2023.