IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 30 September 2023, accepted 9 October 2023, date of publication 13 October 2023, date of current version 18 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3324133

RESEARCH ARTICLE

Deep Semantic Feature Reduction for Efficient Remote Sensing Image Retrieval

RAJESH YELCHURI[1], ALAA O. KHADIDOS[2], ADIL O. KHADIDOS[3], ABDULRHMAN M. ALSHAREEF[2], GANDHARBA SWAIN[4], AND JATINDRA KUMAR DASH[1]

[1]Department of Computer Science Engineering, SRM University-AP, Neerukonda, Mangalagiri, Amaravati, Andhra Pradesh 522502, India
[2]Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[3]Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[4]Department of Artificial Intelligence and Data Science, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh 522302, India

Corresponding author: Jatindra Kumar Dash (jatindrakumar.d@srmap.edu.in)

ABSTRACT Content-Based Remote Sensing Image Retrieval (CBRSIR) is used to find relevant images from large collections of remote sensing images. CBRSIR works by indexing each image in the database with a feature vector. Deep semantic features generated using convolutional neural networks (CNNs) are more powerful than low-level features for CBRSIR tasks because they can comprehend the context and content within an image. However, the major problem with the deep features is its large vector size which in turn can impact the performance of the retrieval system and are more susceptible to noise and outlier data. Therefore, in this work, a modified ResNet50 architecture is proposed that serves as a powerful feature extractor, benefiting from its deep learning capabilities. Specific modifications are introduced to enhance its discriminative power and generalization ability, enabling it to extract more robust deep features for image indexing. The proposed method achieves a mean average precision (mAP) of 0.899 surpassing the popular competing methods ResNet50 and GoogleNet by a substantial margin of 22.02%, 26.79% respectively. Moreover, to address the curse of dimensionality, this study also proposes a novel approach that combines a modified ResNet50 architecture with Linear Discriminant Analysis (LDA) and Maximum Relevance and Minimum Redundancy (MRMR) technique. The proposed approach achieves 85.45% reduction in size of the feature vector using MRMR and 98.19% using LDA, thereby improving retrieval efficiency without impacting the performance.

INDEX TERMS Remote sensing, remote sensing image retrieval, deep learning, convolutional neural networks, minimum redundancy-maximum relevance (mRMR).

I. INTRODUCTION

Remote sensing is the science and technology that make it possible to recognize, quantify, and assess specific properties of objects, regions, or events without coming into direct contact with them. Over the past decade, remote sensing has undergone several technological advancements that have resulted in the capture of high-resolution spatial images. Earlier, aircraft or earth-orbiting satellites were used for this purpose based on the nature of the job, but now, with technological advancement, UAVs (unmanned aerial vehicles), also called drones, are used for this purpose. There are a vast variety of applications that fall under this remote sensing technology, such as weather forecasting, studying the environment and natural disasters, resource utilization, pollution studies, identifying areas of fossil fuel resources, etc. With this increase in technological advancements, huge amounts of data are acquired from time to time for processing using satellites or drones, and this increase in demand has opened up new challenges.

Content-based remote sensing image retrieval (CBRSIR) [1], [2], [3], [4] is highly significant in remote sensing and is a vital tool in facilitating rapid access to satellite and aerial imagery. CBRSIR is used in various diverse

The associate editor coordinating the review of this manuscript and approving it for publication was Nazar Zaki.

fields such as agriculture, disaster management, geology and mining, climate research and forest management etc by providing quick access to relevant remote sensing imagery. Environmental scientists use CBRSIR to monitor changes in landscape, natural resources and vegetation, while agriculture benefits from crop health assessment and yield production. Disaster management relies on CBRSIR for rapid image access during natural calamities and urban planning for infrastructure. Overall, CBRSIR emerges as an important tool which facilitates decision making and empowers resource management across various domains.

Content-based remote sensing image retrieval (CBRSIR) focuses on retrieving images based on image similarity. This is accomplished by indexing the images in the database with certain features like color, shape, texture, etc. Feature extraction plays a vital role in any type of content-based image retrieval system, as the performance of the system greatly relies on the type of features selected for retrieval. Remote sensing images can have a wide range of content, from images with fine-grained textures to images with coarse textures or images with objects. As a result, it is unclear in this domain which descriptor should be used to describe images with such variability. Over the years, various researchers have suggested different methods to extract relevant features, also known as feature descriptors, which can represent the content of the images well in the feature space.

Initially, low-level features, often termed hand-crafted features [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], are proposed and used over the years for feature extraction, which can be either local or global based on the feature extraction technique. These techniques are usually called ''unsupervised'', as they don't use any class or label information corresponding to an image. Global features try to represent the whole image as a feature vector based on characteristics like color, shape, and texture. For instance, to differentiate between the forest and the ocean, features extracted using color information can accurately differentiate between the ocean and a forest as they produce different feature vectors for different colors. At the same time, the color information can be relatively similar for two images that have objects of different content but share the same color information, which isn't accurate. On the other hand, the local features are represented by a set of vectors that are extracted from different patches of an image called the ''region of interest''.

Color and texture features are being used more frequently in Remote sensing image retrieval than shape features, as these images have spectral information which is important for remote sensing image analysis. Bosilj et al. [19] investigated both global and local pattern spectral features for geographical image retrieval and, for the first time, used a dense strategy to implement pattern spectral features. They evaluated their proposed method with other state-of-the-art approaches, which proved to be better. Further more, texture features have also been used to acquire the spatial changes of pixel intensity, which has played great on a variety of remote sensing tasks, including RSIR [20], [21], [22], [23]. Common texture features include GLCM, also called as gray-level-co-occurrence matrices [24], wavelet [6], [7], [12], gabor filters [5], [25] and LBP [9]. However, most of the texture feature descriptors are extracted from gray-scale images discarding crucial color information. So, Shao et al. [26] proposed an improved color texture descriptor which uses color information and performs way better than popular texture features like LBP [9] and Gabor filters. Other works [27] focus on combining color and textural features to improve the performance of image retrieval systems.

SIFT [28] is one of the most popular local feature descriptors and has been widely used for various remote sensing tasks, including scene classification, RSIR, etc. Using SIFT, it is possible to identify prominent patches that surround chosen key points in the images, and the number of selected key points within the image determines how big the feature vector will be. As a result, the retrieval complexity increases if the feature vectors are large in size, which is not suitable for any retrieval system. Bag-of-visual-words [29] can be used to reduce the size of the feature vector by encoding it into a compact image representation. Moreover, these features can be used in combination with other features so as to extract yet more robust representations of an image. Yang and Newsam [30] investigated the use of local invariant features to perform an extensive evaluation of geographic image retrieval on the UCMD data, which was, at the time, the only publicly available remote sensing benchmark dataset. Shape features are also important for content recognition of remote-sensing images [31], [32], [33] because these features primarily define the shape of the objects but are not very good at capturing their spatial relationship. Other popular local features include the histogram of oriented gradient (HOG) [10] and its variant, descriptor pyramid histogram of oriented gradient (PHOG) [13]. The extraction of low-level features still remains an active research area as these features do have some limitations because they are sensitive to scale, rotation, translation, and noise; moreover, they do not represent all the characteristics of an image.

Latter, with the emergence of CNNs, the focus shifted from hand-crafted features to deep features. There are a variety of CNNs available with varying network width and depth. Among them, AlexNet [34] is the first CNN that has shown good improvement on the ImageNet dataset. With this success, several researchers started proposing a variety of CNNs that vary in network width and depth.VGG [35], GoogleNet [36] and ResNet [37] are the most popular CNNs proposed and are considered state-of-the-art CNNs till today. Among these CNNs, ResNet has gained more popularity because it mainly addresses the gradient vanishing problem that arises with the increase in the number of convolutional layers in the CNN. This ResNet is composed of deep residual blocks, which could even break the barrier of a hundred layers and reach over a thousand layers.

Few researchers have explored the use of the deep features from CNNs for various tasks. Agrawal et al. [38] have used the deep features extracted from the popular CNNs like VGG19 and ResNet50 to retrieve the chest CT images with feature vector sizes of 4096 and 2048. The chest CT images are trained on CNNs using transfer learning. Mohammed et al. [39] used VGG19 to retrieve the images from the fully connected layer which have feature vector size in the order of thousands. Latter, few researchers have proposed retrieval techniques by fusing the features in combination with the deep features. Pathak and Raju [40] used both the deep and hand-crafted (low-level) feature to perform the image retrieval on most popular datasets like Corel and Colour-Brodatz. To extract the deep features, GoogleNet is used in combination with the low-level feature HOG, which is used to represent the shape of the image. Similarly, in [41], fusion of features is used to get the high-level representation of the image. The features are extracted from the output of the average pooling layer of the Inception-Darknet CNN. In addition to this, the low-level features extracted from RGB and HSI color space are used to perform the retrieval. Although the feature vector representation is high-level, but the concatenated vector is large. Liu et al. [42] used fusion technique to combine deep and low-level features. To perform the retrieval, features from two CNNs are used along with the gabor and DWT features. The above mentioned methods uses deep features either independently or in combination with low-level features to enhance the image retrieval efficiency. However, a noteworthy concern arises as these yield large feature vectors which can impact the latency of retrieval system.

Recently, several methods have emerged that rely on fuzzy rules [43], [44], deep metric learning, and attention mechanisms [45], [46], which use the discriminative ability of the deep features of the CNN. Deep metric learning is used in several research areas like natural image retrieval [47], person re-identification [48] and face recognition [49], which has proven to be effective. Using this technique, features can be represented in the feature space in such a way that the objects that are semantically similar lie close to each other, and those that are different are kept far away. Cao et al. [46] proposed a triplet deep metric Convolutional Neural Network (CNN) method that can extract representative features of an image such that images within the same class come together and those belonging to different classes move far away. However, methods that use triplet learning require a large number of triplets for training, which can be challenging to generate for large datasets and can limit the scalability of these triplet learning algorithms. Ye et al. [43], used fuzzy rules and fuzzy distance to improve the retrieval accuracy. To do this, two fuzzy class memberships are used; one is used to determine the classification confidence, and the other is used to determine whether an image belongs to either of the three fuzzy sets, i.e., 'medium confidence,' 'low confidence,' or 'high confidence,' based on the classification confidence.

Furthermore, Yelchuri et al. [44] proposed an image retrieval system for texture image retrieval which uses the strength of the CNN in calculating the fuzzy class membership of the query image for all the available output classes and uses weighted distance metric to retrieve the images from the wavelet feature space. Apart from this, these fuzzy methods are fully supervised in nature and need the class label information which should be indexed in the database. Coming to the attention mechanism, Noh et al. [45] used key points based on the attention mechanism to select the most prominent deep local features whereas Chaudhuri et al. [50] proposed a graph CNN that used edge attention and node attention mechanisms to emphasize important visual context by giving more weight to the significant nearby regions that highlight a key node. At the same time, these attention mechanisms are computationally expensive and are sensitive to noise in the input images.

Overall, the researchers have leveraged the power of deep learning, particularly CNNs for feature extraction. The CNNs have emerged as a tool for automatic feature extraction and to achieve this, researches used popular state-of-art CNNs such as ResNet [37], VGG [35], Inception [36], DenseNet [51], Xception [52] etc. In addition, fusion based methods [38], [39], [40], [41], [42] are used to represent the image which use deep and low-level features to extract the high-level representation of an image to improve the performance of the retrieval. Moreover, the adaption of fuzzy logic [43], [44] and deep metric learning [46], [47] has yielded powerful feature representations, which further enhanced the discriminative capabilities of the CBRSIR systems. A detailed survey of the applications of deep learning for content-based image retrieval can be found in Zhou et al. [53].

The main drawback of the CNNs is, that it requires a lot of labeled data and training time in order to train the network. Moreover, the features extracted using these trained CNNs are often big in size i.e., highly dimensional in nature, and may contain redundant information. Convolutional neural networks (CNN) are very popular in the fields of image classification and object detection due to their ability to learn minute image features. Many CBIR systems have also been implemented over the last few years that take the help of CNN to extract the image features for indexing the images. However, the feature vector obtained by most CNNs is typically large in size. In addition, not all the features obtained in the flattening layer of a CNN may be useful, and a few may be redundant. The above drawbacks limit the performance of any CBIR system in terms of retrieval speed and retrieval efficacy (average precision and recall). The proposed approach investigates a popular CNN architecture and modifies the architecture to obtain a reduced-length feature vector. The reduced feature vector is further investigated using two popular feature selection techniques for better retrieval accuracy in the field of satellite image retrieval known as CBRSIR

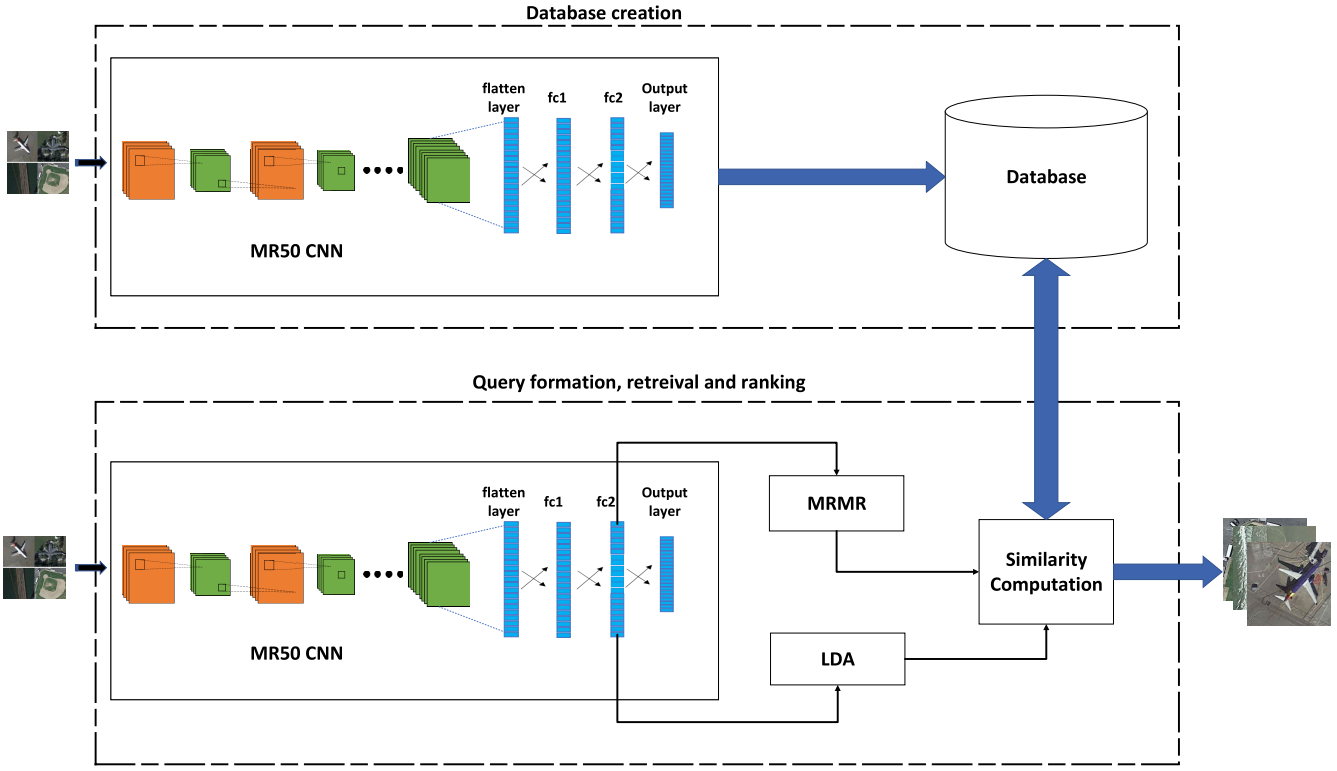


FIGURE 1. Block diagram of proposed method.

(Content-based remote sensing image retrieval). The proposed system presented in this paper considers performance along with the feature vector size, which plays an important role in deciding the latency of the system. The rest of the paper is organized as follows: Section II describes the proposed method, and Section III provides details about the metrics and the dataset used in the evaluation of the system. Section IV provides the details of the evaluation of the proposed system with other competing methods. Section V provides the conclusion of the work presented in the paper.

II. PROPOSED METHOD

The block diagram of the proposed method is shown in Figure.1. The three main components of the proposed method are:

1) Training the modified ResNet MR50.
2) Creation of the database with the deep features extracted from the modified ResNet MR50 CNN.
3) Query formation and retrieval.
   a) Extraction of deep features using the trained CNN models.
   b) Identifying an effective low-dimensional representation of high-level information for image retrieval using the popular techniques 'LDA' and 'mRMR'.
   c) Retrieval and ranking of the images using the city block or manhattan distance metric.

A. TRAINING THE MODIFIED RESNET MR50 CNN

According to Basha et al. [54], to achieve good classification performance with deeper CNNs, i.e., CNNs with a higher number of convolutional layers, there is no need for a large number of neurons in the fully connected (FC) layers, irrespective of the dataset. Moreover, the problem with the deep features is that they are large in size, which can affect the retrieval performance of the system. Therefore, to improve the classification ability of the ResNet50 CNN and to keep the feature vector moderate in size, the classification layer of the ResNet50 CNN is modified as shown in Fig. 2 and named 'MR50'. The modified CNN MR50 has three layers; two of them are used as the dense layers named FC1 and FC2 having 1024 and 512 neurons respectively and the third layer i.e., the output layer, consists of a total of 38 neurons because the chosen dataset consists of 38 classes. To train the MR50 CNN, 'transfer learning' is employed because it has several benefits, such as accelerating the training process and consuming fewer computing resources. Transfer learning is often described as using a model that has already been trained to accelerate the learning process for a new task, which in turn improves the model's overall accuracy and performance. Therefore, at first, the ResNet50 CNN weights trained on the 'Imagenet' dataset are transferred to the MR50 CNN, which is then trained on the 'PatternNet' image dataset using Keras (with TensorFlow as the backend). The images of the PatternNet dataset are processed to have a size of 224 $\times$ 224 x 3 (W, H, C), as per the requirement

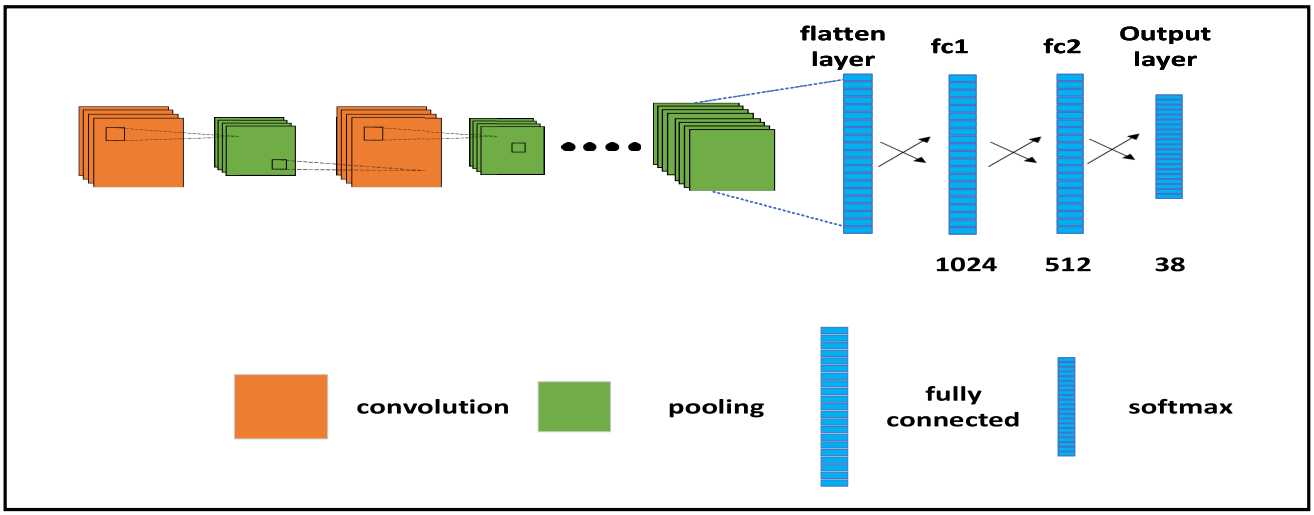


FIGURE 2. Block diagram of MR50 (Modified ResNet50) CNN.

TABLE 1. Classification performance of MR50 CNN on Training, Validation and Test set.

Model	Average training accuracy	Average validation accuracy	Average test accuracy
MR50	98.91	98.31	98.41

of the ResNet50 CNN, where W-width, H-Height, and C-Channel. During the process of training, categorical cross entropy is used as the loss function and Adam is used as an optimizer and Softmax is used as an activation function in the output layer. A 3-fold cross-validation is used to test the classification accuracy of the trained models. In each fold, 66.66% of images are used as training and validation data, and 33.33% are used as test data. The average 3-fold cross-validation classification performance of the MR50 CNN on train, validation, and test set is shown in Table. 1.

B. CREATION OF THE DATABASE

The images from the PatternNet dataset are preprocessed before being fed to the modified ResNet architecture MR50. The preprocessing step includes resizing the images of the PatternNet dataset as per the requirements of the ResNet architecture. After this pre-processing step, the images are fed to MR50 CNN to extract the deep features from the pool and fully connected layers of the trained CNNs. The trained CNN MR50 models are used to extract the deep features from an image. In order to extract the features from the CNN, the layers to the right side of the 'layer of interest' are chopped so that the outputs from the 'layer of interest' can be extracted. Therefore, each input image is fed to the trained MR50 CNN models, and the features are extracted from the respective layers. The size of these feature vectors depends on the layer that is considered as the output layer, i.e., for the last pooling layer, the feature vector size is 2048, whereas for fully connected layers FC1 and FC2, the size is 1024 and 512, respectively. Finally, the database is indexed with the deep features extracted from the pool and fully connected layers of the trained CNN models are represented in the database as shown below:

$$\overrightarrow{PL(x)} = [PL_1, PL_2, PL_3 \ldots\ldots PL_{2048}] \quad (1)$$

$$\overrightarrow{FC1(x)} = [C_1, C_2, C_3 \ldots\ldots C_{1024}] \quad (2)$$

$$\overrightarrow{FC2(x)} = [D_1, D_2, D_3 \ldots\ldots D_{512}] \quad (3)$$

C. QUERY FORMATION AND RETREIVAL

1) EXTRACTION OF DEEP FEATURES OF QUERY IMAGE USING THE TRAINED CNN MODELS

The pool layer, FC1, and FC2 features can be extracted from the MR50 CNN; however, in the latter stage of the proposed method, only the FC2 features were employed due to their demonstrated superior performance among the three. As a result, the FC2 features for each query image were computed and indexed in the database.

2) IDENTIFYING AN EFFECTIVE LOW-DIMENSIONAL FEATURE VECTOR

To make the retrieval simple and effective, the deep feature vector FC2 as shown in Equation 3 of the query image is further sent to modules 'MRMR' and 'LDA' in order to identify an effective low-dimensional feature vector representation of the query image. The process of identifying the effective features using both 'MRMR' and 'LDA' is briefly discussed in Sections IV-D1 and IV-D2. The feature vectors produced using MRMR and LDA are given

as follows:

$$\overrightarrow{FV_{mrmr}(x)} = [MRMR_1, MRMR_2, MRMR_3 \ldots\ldots MRMR_{37}] \quad (4)$$

$$\overrightarrow{FV_{LDA}(x)} = [LDA_1, LDA_2, LDA_3 \ldots\ldots LDA_{298}] \quad (5)$$

3) RETRIEVAL AND RANKING OF THE IMAGES USING THE CITY BLOCK DISTANCE

In this work, the most popular known distance metric City-block or Manhattan distance is used to retrieve and rank the images. Let 'Q' be the query image and 'P' be any image in the database with deep feature vector $\overrightarrow{FC2(Q)}$ and $\overrightarrow{FC2(P)}$ respectively and the distance between Q and P is given by:

$$dist_{cb}(Q, P) = \sum_{j=1}^{K} |Q_j - P_j| \quad (6)$$

where 'K' represent the dimension of the deep feature vector.

III. EXPERIMENTAL SETUP

A. DATABASE USED

In the paper, the work is mainly focused on the satellite image dataset PatterNet, as this is the largest publicly available high-resolution remote sensing dataset (Zhou et al. [55]), which has a total number of 38 classes: parking lot, solar panel, beach, freeway, christmas tree farm, nursing home, bridge, baseball field, football field, oil gas field, ferry terminal, river, runway marking, airplane, railway, wastewater treatment, runway, basketball court, tennis court, parking space, mobile home park, overpass, swimming pool, harbor, forest, closed road, chaparral, coastal mansion, storage tank, cemetery, dense residential, sparse residential, intersection, transformer station, golf course, crosswalk, oil well and shipping yard. Besides, this dataset has 30,400 images with 800 images per class and each image in the dataset is of size 256 × 256 pixels. The images in this dataset are gathered from Google Earth imagery or the Google Maps API for US cities. Moreover, in these images, the class of interest covers most of the image with a small amount of background which is not the case in the other popularly known remote sensing datasets such as the UC Merced dataset, WHU-RS19, RSSCN7, and Aerial image dataset. Because of its large collection and the fact that most of the images contain the region of interest, PatternNet is regarded as a superior dataset, particularly for deep learning. All the technical details regarding the database is briefly explained in [55]. Sample images from the dataset are shown in Figure 3 and the summary of the dataset is shown in Table 2.

TABLE 2. PatternNet dataset information.

Sno	Class	No.of images	Size
1	parking lot	800	256 x 256
2	solar panel	800	256 x 256
3	beach	800	256 x 256
4	freeway	800	256 x 256
5	christmas tree farm	800	256 x 256
6	nursing home	800	256 x 256
7	bridge	800	256 x 256
8	baseball field	800	256 x 256
9	football field	800	256 x 256
10	oil gas field	800	256 x 256
11	ferry terminal	800	256 x 256
12	river	800	256 x 256
13	runway marking	800	256 x 256
14	airplane	800	256 x 256
15	railway	800	256 x 256
16	wastewater treatment	800	256 x 256
17	runway	800	256 x 256
18	basketball court	800	256 x 256
19	tennis court	800	256 x 256
20	parking space	800	256 x 256
21	mobile home park	800	256 x 256
22	overpass	800	256 x 256
23	swimming pool	800	256 x 256
24	harbor	800	256 x 256
25	forest	800	256 x 256
26	closed road	800	256 x 256
27	chaparral	800	256 x 256
28	coastal mansion	800	256 x 256
29	storage tank	800	256 x 256
30	cemetery	800	256 x 256
31	dense residential	800	256 x 256
32	sparse residential	800	256 x 256
33	intersection	800	256 x 256
34	transformer station	800	256 x 256
35	golf course	800	256 x 256
36	crosswalk	800	256 x 256
37	oil well	800	256 x 256
38	shipping yard	800	256 x 256

B. PERFORMANCE METRICS USED

The proposed method's retrieval performance is assessed and evaluated with that of other retrieval methods using the metrics like precision, recall, mAP (Mean average precision), ANMRR (Average Normalized Modified Rank Retrieval) and City-block or Manhattan distance.

- Precision: It is defined as the ratio of the number of relevant or similar images retrieved for a given query to the total number of images retrieved from the database. Let τ_R represent the similar/relevant images in the database, and τ_T represents the collection of 'n' retrieved images given a query image 'q'. Percentage precision is calculated as shown in Eq.7.

$$precision(query, n) = \frac{|\tau_T \cap \tau_R|}{\tau_T} \times 100 \quad (7)$$

- Recall: Recall is the ratio of the total number of relevant images retrieved to the total number of relevant images that exist in the database. Percentage recall is calculated as shown in Eq.8.

$$recall(query, n) = \frac{|\tau_T \cap \tau_R|}{\tau_R} \times 100 \quad (8)$$

- Mean average precision (mAP): During the query phase, all the images in the database are ranked based on the distance between the features of the query image and the samples in the database in ascending order. After obtaining this ranked list, the average precision (AP)

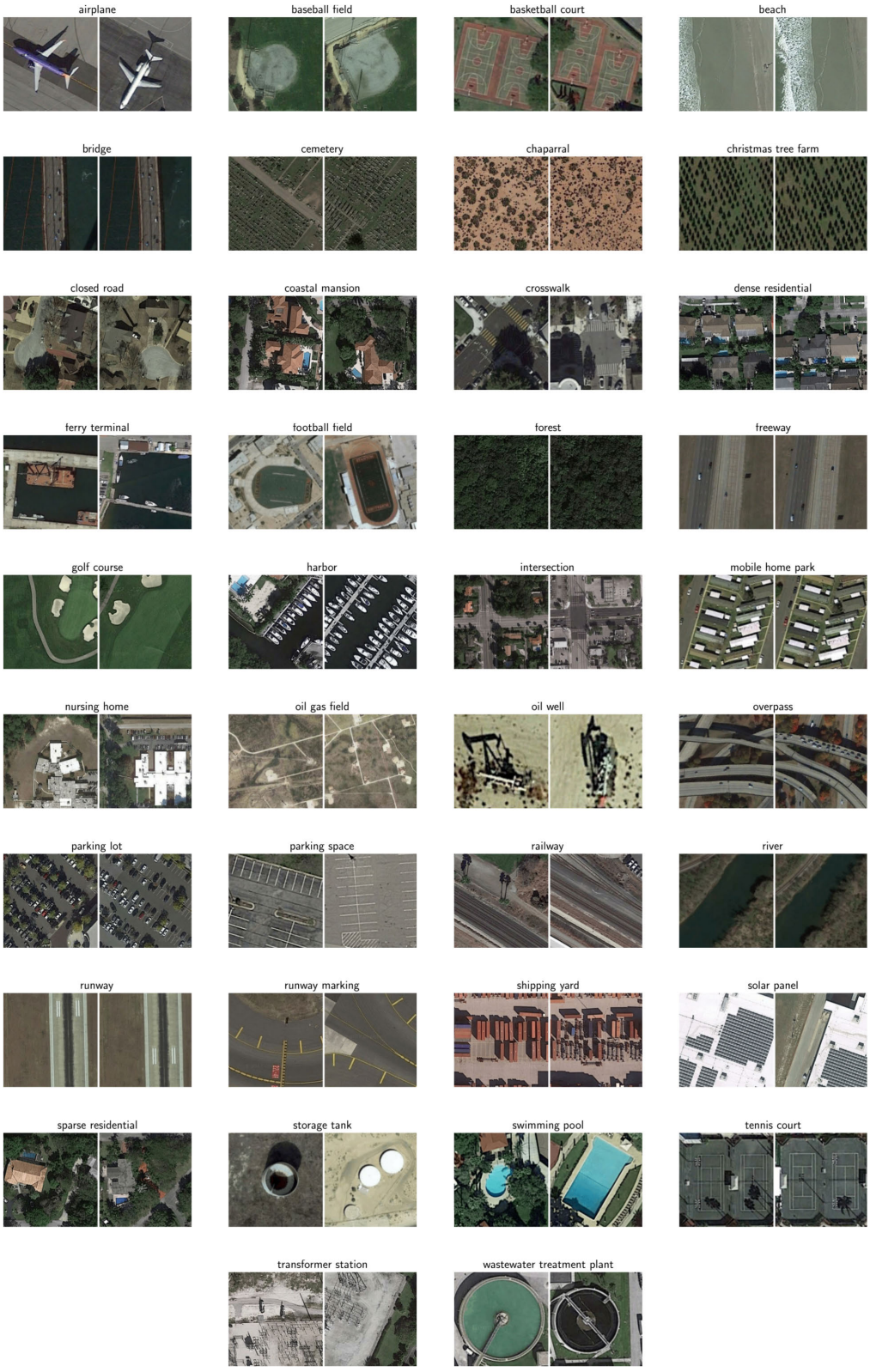


FIGURE 3. Sample images from each class of the PatternNet dataset.

for each query image is computed. Finally, averaging the AP of all query images mAP can be obtained. This mAP metric is used to evaluate the effectiveness of retrieval. The Mean average precision (mAP) is

given by:

$$mAP = \frac{1}{Q}\sum_{i=1}^{Q}\frac{1}{rel_i}\sum_{k=1}^{rel_i}P_{ik} \tag{9}$$

where 'Q' is the number of query images, rel_i is the total number of relevant images for the i^{th} query from the database. P_{ik} is precision of top k^{th} image retrieved **w.r.t** i^{th} query image.

- ANMRR: Another parameter used to assess the performance is 'Average Normalized Modified Rank Retrieval (ANMRR)'. A lower ANMRR indicates better performance, and a higher value indicates worse performance. The average rank $AVGR_{(q)}$ for any given query 'qy' is given by:

$$AVGR_{(qy)} = \sum_{k=1}^{NGT_{(qy)}}\frac{Rank(k)}{NGT_{(qy)}} \tag{10}$$

where $NGT_{(qy)}$ represents the total number of ground truth images exists for a given query 'qy' in the database and $Rank(k)$ is determined using the equation below:

$$K = min(X \times NGT_{(qy)}, 2 \times GMT) \tag{11}$$

where, $GMT = max\left\{NGT_{(qy)}\right\}$, for all qy's of a dataset (12)

The modified retreival rank of the query 'qy' is calculated as follows:

$$MRR_{(qy)} = AVGR_{(qy)} - 0.5 \times \left[1 + NGT_{(qy)}\right] \tag{13}$$

Finally, the normalized modified retrieval rank is computed as follows:

$$NMRR_{(qy)} = \frac{MRR_{(qy)}}{1.25 \times K - 0.5 \times \left[1 + NGT_{(qy)}\right]} \tag{14}$$

Finally, the average NMRR for all queries is calculated as follows:

$$ANMRR = \frac{1}{Q}\sum_{q=1}^{Q}NMRR_{(qy)} \tag{15}$$

- City-block distance: City-block or Manhattan distance is used to rank the retrievals. The total absolute difference between the two vectors is used to determine the city-block distance. The city-block distance between two points, E, and F, with K-dimensions is calculated as:

$$dist(E, F) = \sum_{j=1}^{K}\left|E_j - F_j\right| \tag{16}$$

IV. RESULTS AND DISCUSSION

The Nvidia DGX-1 Deep Learning System, which consists of a collection of dockers and the Ubuntu operating system, is used to conduct the experiments. This system includes 40,960 NVIDIA Cuda Cores and 8 Tesla V100 GPUs, each with 32GB of memory. This Nvidia DGX-1 deep learning system comes with a SATA 3.0 SSD and 480GB of storage with 6Gb/s.

Few methods reported in the literature that use hand-crafted features and deep features to index the images are used for comparative study. These are as follows:

1) Simple Statistics (SS): The method simple statistics [30], [55] i.e., uses the mean and standard deviation of a simple gray-scale image
2) Color Histogram (CH): In this method, color histogram [56] is used as a feature set, which is created by concatenating the three histograms and quantizing each channel of the RGB color space into 32 bins.
3) Gabor Texture (GT): This method uses a Gabor filter [26], [55] with five scales and eight orientations with a filter window size $32x32$.
4) GIST: This method uses gist (global image statistics) features as a feature vector [55], [57], which summarizes the gradient information. Convoluting different filters at different scales and orientations yields these features. Thus, it is possible to measure the high and low-frequency repeated gradient directions in an image using these features.
5) Local Binary Pattern (LBP): This method utilizes features derived from a Local Binary Pattern [9], [55] with an 8-pixel circular neighborhood radius of one. LBP is used to capture the local texture information in an image by dividing the image into small regions and computing a binary pattern for each region based on the intensity values of its pixels.
6) Pyramid Histogram of Oriented Gradients (PHOG): The method uses features computed using PHOG [10], [13], [55]. The feature vector is computed by building a quadtree of orientation histograms across the entire input image and then concatenating the histograms for each cell of the quadtree into a vector representation.
7) AlexNet_FC1 (AFC1): This method uses trained AlexNet [34], [55] to extract deep features from the first fully connected layer, FC1. AlexNet is a deep convolutional neural network (CNN) architecture that is considered a milestone in the development of deep learning. It was the first CNN architecture to win the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), a major computer vision benchmark, in 2012.
8) AlexNet_FC2 (AFC2): This method uses deep features [34], [55] extracted from the second fully connected layer FC2 of AlexNet instead of FC1.
9) VD16_FC1 (VDFC1): This method uses deep features [35], [55] extracted from the fully-connected layer number one FC1 of a VD16 CNN, which is a variant of VGG with 13 convolution layers and 3 fully-connected layers.
10) VD16_FC2 (VDFC2): This method uses deep features [35], [55] extracted from the fully-connected

layer number two of VD16. The architecture is the same as VD16, but the feature extraction is done from the fully-connected FC2 layer instead of FC1.

11) GoogleNet (GNet): This method uses the deep features of GoogleNet [36] extracted from the last pool layer of the CNN. This CNN uses an Inception module, which is a building block that combines multiple convolutional and pooling operations in a single module. This allows the network to learn more complex features compared to a traditional CNN.
12) ResNet50: This method uses deep features [37], [55] extracted from the last pool layer of ResNet50. ResNet50 is a 50-layer deep network that uses residual connections, which are shortcuts that allow information to bypass one or more layers in the network. This helps to alleviate the vanishing gradients problem, where the gradients of the network become too small to update the parameters effectively. The residual connections allow the network to train much deeper architectures without sacrificing performance.
13) ResNet101: This method uses deep features [37], [55] extracted from the last pool layer of ResNet101 CNN. The architecture is similar to ResNet50 and can be considered a deeper version of ResNet50 architecture having a deep network with 101 layers.
14) ResNet152: This method uses deep features [37], [55] extracted from the last pool layer of ResNet152 CNN. The architecture is similar to ResNet50 comprising a deeper network with 152 layers.
15) MR50_FC2 (MR50): This method uses the features extracted second fully connected layer of the proposed CNN 'MR50'. The length of the feature set is further reduced using the techniques discussed in IV-D1 and IV-D2.

A. PERFORMANCE OF HAND-CRAFTED FEATURES

This section discusses the performance of the low-level features, i.e., the hand-crafted features. Several performance metrics are computed and are shown in Table 3. The methods Simple statistics (SS), Color-histogram (CH), Gabor Texture (GT), GIST, Local Binary Pattern (LBP), and PHOG are shown under the category of hand-crafted features. The performance of the hand-crafted features at various operating points is measured in terms of ANMRR (Average Normalized Modified Retrieval Rank) and mAP (Mean Average Precision). From Table 3, it is evident that among the hand-crafted features, features extracted using Gabor texture (GT) perform the best. The features performance of methods SS and PHOG, which use Simple statistics and PHOG are poor when compared with the other hand-crafted features. For all the measures shown in Table 3, lower ANMRR indicates better performance, and for all other measures (mAP, P@5, P@10, P@50, P100, P@1000), higher values indicate better performance.

B. PERFORMANCE OF DEEP FEATURES

This section discusses the performance of the deep features, i.e., the features extracted from the fully connected layer or the pooling layers of the CNN. These are shown in Table 3. The performance of the deep features is far better than the hand-crafted features, which indicates that the CNNs are able to learn the discriminating features well compared to the hand-crafted features. From Table 3, it is clear that the deep features extracted from 'ResNet50' CNN give better performance than the other deep features extracted using various pre-existing CNNs (AFC1, AFC2, VDFC1, VDFC2, GNet, ResNet101, and ResNet152) cited under the category of deep features. Among these pre-existing CNNs that use deep features, the deep features extracted from the ResNet50 architecture are found to be better than any other CNN deep features reported in Table 3. The performance of the deep features extracted from the second fully connected layer FC2 of AFC2 and VDFC2 is found to be better than the deep features extracted from AFC1 and VDFC1, which extract features from the fully connected layer FC1. Among all of the methods reported in Table 3, the features extracted from the modified ResNet CNN MR50 are found to be the best i.e., the features extracted from the FC2 layer of the MR50 CNN have performed better than all other competing methods. Hence, it is observed that the features extracted from the fully connected layer FC2 perform better than FC1. Therefore, in the proposed method, the features extracted from the second fully connected layer of MR50 CNN are taken into consideration to improve the performance.

C. PERFORMANCE EVALUATION OF THE PROPOSED METHOD

This section discusses the performance of the proposed method. All the performance metrics reported are computed using the city block distance. The performance of the proposed CNN MR50 (modified) is compared with that of the competing methods reported in Table 3, which use hand-crafted features and deep features, and the following observations are made:

1) Table 3 provides a brief overview of the performance of different features extracted using different methods, a few of them make use of hand-crafted features, and others use deep features.
2) Among the methods that use hand-crafted features, Gabor Texture (GT) performs better than all other hand-crafted techniques reported in Table 3 with a mean average precision (mAP) of 27.69%.
3) The features extracted using the method GT filter have shown better mAP (Mean Average Precision) over other methods such as SS, CH, GIST, LBP, and PHOG by 21.07%, 2.59%, 7.68%, 1.86%, and 14.57%, respectively. Furthermore, GT outperforms all other methods that use hand-crafted features in terms of ANMRR (the lower the ANMRR, the greater the performance).

TABLE 3. Performance of hand-crafted and deep features with metrics ANMRR (Average normalized modified retreival rank), mAP (Mean average precision), and Precision *(P@k)*. Lower ANMRR indicates better performance, for mAP and *P@k*, a large value indicates better performance.

Features	Methods	ANMRR	mAP	P@5	P@10	P@50	P@100	P@1000
hand-crafted	Simple Statistics (SS)	0.8968	0.0662	0.0739	0.0741	0.0739	0.0738	0.0701
	Color Histogram (CH)	0.6697	0.251	0.7475	0.7032	0.5733	0.5062	0.2349
	Gabor Texture (GT)	0.6422	0.2769	0.8021	0.7631	0.6393	0.5674	0.2556
	GIST	0.7511	0.2001	0.6429	0.5957	0.4645	0.4013	0.1773
	LBP	0.647	0.2583	0.6358	0.6027	0.5115	0.4646	0.2505
	PHOG	0.8162	0.1312	0.4852	0.443	0.3376	0.2903	0.1295
deep features	AlexNet_Fc1 (AFC1)	0.3328	0.6003	0.9545	0.9438	0.8986	0.8617	0.4934
	AlexNet_Fc2 (AFC2)	0.326	0.6042	0.9448	0.9331	0.8872	0.8529	0.4985
	VD16_Fc1 (VDFC1)	0.3302	0.602	0.9388	0.9268	0.8806	0.8459	0.4959
	VD16_Fc2 (VDFC2)	0.3283	0.5986	0.9327	0.9204	0.874	0.8404	0.4972
	GoogLeNet (GNet)	0.2983	0.6311	0.9445	0.9331	0.8918	0.8603	0.5202
	ResNet50	0.2606	0.6788	0.9665	0.9594	0.9274	0.9006	0.5533
	ResNet101	0.2624	0.6765	0.9638	0.9551	0.9208	0.8933	0.5525
	ResNet152	0.2632	0.6757	0.9635	0.955	0.9208	0.8939	0.5511
	MR50_FC2 (MR50)	**0.0704**	**0.8990**	**0.9957**	**0.9943**	**0.9899**	**0.9862**	**0.7318**

4) Although the hand-crafted features are not as good as the deep features, GT has shown decent performance with an average precision of 80.21%, 76.31%, 63.93%, and 56.74% at lower operating points P@5, P@10, P@50 and P@100. However, the average precision at operating point P@1000 has shown a drastic downfall with 25.66% which indicates the performance of features is not all acceptable with higher operating points.
5) When deep features are taken into consideration, clearly, the features extracted using the proposed modified ResNet 'MR50' has shown better performance than any other reported in Table 3.
6) The performance of the deep features extracted from FC1, FC2 and pool layers of MR50 is calculated, which resulted in mAP of 88.70%, 89.90%, and 49.73%.
7) The features extracted from FC1, and FC2 using the CNN MR50 has shown significant performance improvement over the deep features extracted from ResNet50 CNN, which has better performance with the other methods reported in Table 3.
8) Among the Pool, FC1, and FC2 features, FC2 features have better performance using the MR50 CNN. If mAP is taken into consideration, FC1 has shown a performance improvement of 20.82%, and 22.02% using FC2 over the ResNet50 deep features.
9) Similarly, if the ANMRR (Average Normalized Modified Retrieval Rank) metric is taken into consideration, the features FC1, and FC2 extracted using the proposed CNN MR50 have an ANMRR of 0.0775 and 0.0704 respectively, which is a good improvement over the competing ResNet50 deep features, having an ANMRR of 0.2606.
10) Coming to the metric precision is concerned, a higher value indicates better performance and a lower value indicates low performance, but as far as the ANMRR is concerned, the lower the ANMRR, the better performance. So, the FC2 features extracted using the MR50 has shown a better performance of 0.0704 over the ResNet50 deep features with an ANMRR of 0.2606.
11) On the whole, the deep features extracted using MR50 has shown significant performance improvement over the nearest competing ResNet50 CNN deep features. Another notable performance improvement is the average precision of MR50 CNN deep features has better performance improvement of 2.92%, 3.49%,6.25%,8.56%, and 17.85% at operating points viz., P@5, P@10, P@50, P@100, and P@1000.

D. IMPROVISING THE PERFORMANCE OF THE PROPOSED METHOD

Firstly, the features from the pooling layer and the fully connected layers are extracted to check the performance of the modified CNN 'MR50'. In order to do this, the layers that are on the right-hand side of the 'layer of interest' are chopped. The average precision and average recall of all the features are plotted in Fig 4. In addition, the average precision and average recall of the deep features extracted from MR50 CNN are shown in Table 4 at various operating points. From Fig 4 and Table 4, it is observed that the performance of the deep features extracted from the fully connected layer FC2 are better than FC1 and Pool features. Furthermore, FC2 features are giving better performance than any other features that are reported in Table 3. As a result, FC2 features are taken into consideration for further evaluation to improve the retrieval performance. The following observations are made from Table 4:

1) FC2 features give better performance than any of the other features that are reported in Table 3.

TABLE 4. Average Precision and Average Recall of the deep features of MR50 CNN at various operating points.

Features	feature vector size	P@5	R@5	P@10	R@10	P@50	R@50	P@100	R@100	P@800	R@800	P@1000	R@100
POOL	2048	0.9645	0.0060	0.9468	0.0118	0.8895	0.0556	0.8471	0.1059	0.5266	0.5266	0.4550	0.5688
FC1	1024	0.9945	0.0062	0.9927	0.0124	0.9868	0.0617	0.9818	0.1227	0.8657	0.8657	0.7205	0.9007
FC2	**512**	**0.9957**	**0.0062**	**0.9943**	**0.0124**	**0.9899**	**0.0619**	**0.9862**	**0.1233**	**0.8952**	**0.8952**	**0.7318**	**0.9148**

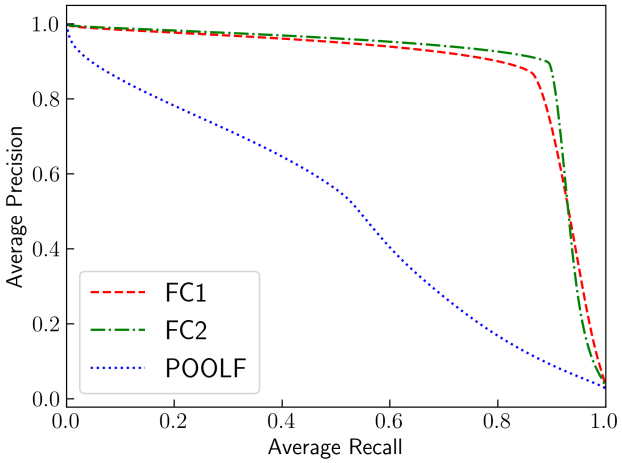


FIGURE 4. Average Precision Vs Average Recall of the deep features extracted from the CNN MR50.

2) The features FC1 and Pool have shown significant performance improvement over hand-crafted features, but they are not as good as the FC2 features.
3) The performance of the deep features FC1 and FC2 are nearly identical and do not differ significantly at lower operating points. However, with the increase in the operating point, FC2 shows improvement over FC1.
4) At lower operating points of 5, 10, 50, and 100, the performance between FC1 and FC2 doesn't vary much, but at operating points of 100 and 800, the precision of FC2 has increased by 1.41%, and 2.95% respectively as compared to FC1 features.
5) Pool features show good performance but are not on par with FC1 and FC2 at lower operating points. In addition, as the operating point increases, the Pool features show a drastic reduction in performance.

In brief, FC2 features have performed better than FC1 and Pool features across all the operating points. So, FC2 features are taken into consideration for improving the retrieval performance. The performance of any CBRSIR system depends on the type of features that are used and the size of the feature vector. These two factors play an important role in any retrieval system, the first, affects the performance of the system while the second affects the retrieval time. Therefore, to enhance retrieval performance, it is desirable to obtain an effective and low-dimensional representation from the features that are already available. A considerably smaller feature subset minimizes processing costs while maintaining the accuracy of the retrieval. In view of this, the study in the next sub-section aims to investigate the use of a feature selection strategy based on the mRMR criterion as well as the usage of LDA (Linear Discriminant Analysis) to learn low-dimensional features from high-level features. In order to minimize the feature vector without compromising the performance, the proposed method uses two techniques mRMR and LDA as discussed in section IV-D1 and section IV-D2.

1) FEATURE SELECTION USING MRMR

Maximum Relevance-Minimum Redundancy [58] feature selection is employed in order to select a subset of features that have a high relevance to the target variable and a low redundancy with each other. Most feature selection algorithms solely take into account how features relate to the target ignoring the interdependence among the features whereas the mRMR technique considers this too. Basically, this is a step process used to select the best feature subset. In the first step, according to the maximal statistical dependency criterion based on mutual information, the mRMR technique ranks features. The subsequent step is the gradual inclusion of top features, which creates the feature subsets until there is no further addition of the feature. As a result, the first subset contains only one top-ranked feature, the second feature subset contains the top two ranked features, and so on.

To calculate the optimal dimension of the feature set, all the training data is considered. All the training data is passed through the trained CNN MR50, and features are obtained from the FC2 layer output. The features that have maximum relevance with the target and minimum redundancy with other features are selected using the mRMR feature selection method. All the features obtained for the training data are then ranked using the mRMR algorithm [58]. To demonstrate the impact of feature dimension on retrieval performance, the average precision is computed for different feature-length/dimension for all the test data, and the same is plotted in Figure 5. From Figure 5, it is observed that further inclusion of any feature beyond a value doesn't impact the performance of the system much. Therefore, a reduced optimal feature subset with size 298 is selected empirically, which is close to the performance of the original feature set.

2) FEATURE SELECTION USING LDA

A prominent method for reducing the dimensions of feature vector is linear discriminant analysis (LDA). It focuses on

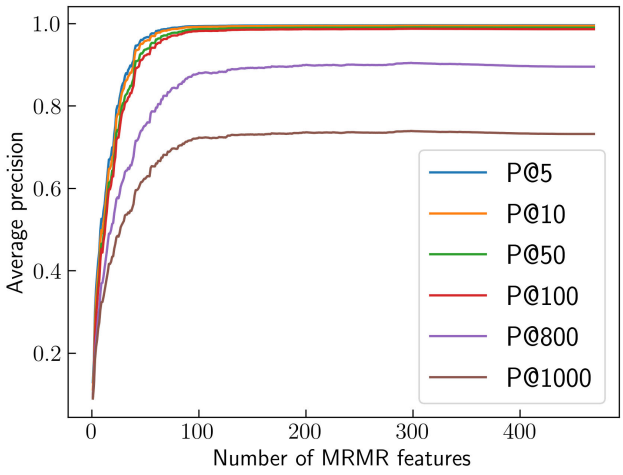


FIGURE 5. Average precision vs dimension of feature vector obtained using MRMR feature selection.

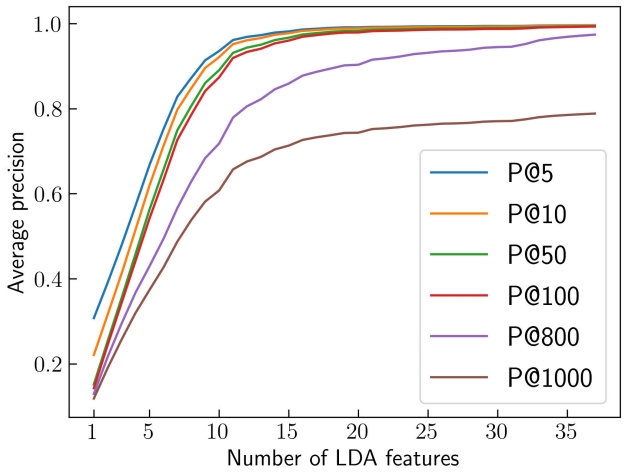


FIGURE 6. Average precision vs dimension of feature vector obtained using LDA.

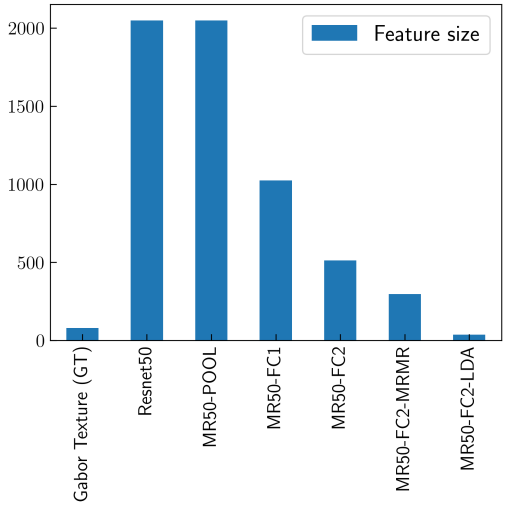
maximizing the separability among the existing categories (classes) in the target variable. This approach is considered as a supervised approach because it requires both features and class labels. The main goal of LDA is to maximize the separability among the different categories of data present in the feature space, i.e., to project the data onto new axes or feature space in such a way that it can maximize the class separability. LDA uses two criteria to project features onto a new axes, they are:

- maximize the difference between the two classes' means.
- Reduce variation within each class to a minimum.

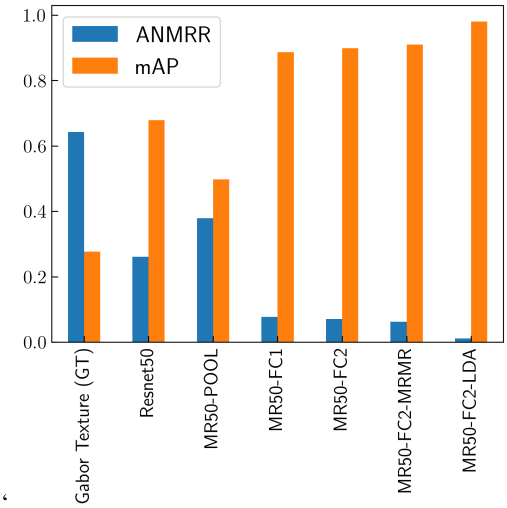
In addition to this, dimension reduction can also be done at the same time. LDA can reduce the dimensionality of the features to *C-1*, where *C* is the number of classes in the target variable. For instance, if there are 10 classes in the target variable, the new feature space can have at most 9 features. All the training data used for training the models was gathered. The features for all these training data are computed using the trained models. For each training instance, the feature vector is formed by taking the FC2 layer output. Linear discriminant analysis (LDA) is then applied to transform these features into new dimensions. For the database considered, LDA can have at most 37 features in the feature space, as the database has only 38 classes. Finally, to study the impact of the dimension of the new feature vector, the average precision at various operating points (viz., 5, 10, 50, 100, 800, and 1000) is plotted vs. the dimension of the selected feature vector. Figure 6 demonstrates the impact of the dimension of the new feature set on the retrieval performance. It is observed that for lower operating points, a feature dimension close to 20 performs closely with the original feature vector (FC2), which is of dimension 512. For higher operating points, it is observed that the optimal feature dimension is 37, which performs closely with the FC2 features. Therefore, the dimension of the feature vector can be reduced to 37 without compromising retrieval accuracy. This will also improve the retrieval speed.

The following observations were made from the experiments based on Section IV-D1 and IV-D2:

1) Table 5 and its corresponding Figure 7 shows the difference in the performance of the proposed method with other better methods which use Gabor Texture (M3) hand-crafted features and ResNet50 deep features.
2) From Figures 7a and 7b, although the dimension of the Gabor texture hand-crafted feature is computationally less compared to other deep features vectors (such as ResNet50, MR50-FC2, etc.) methods, it has a performance trade-off.
3) The size of the feature vector obtained from the FC1 and FC2 layer output of the modified MR50 CNN are 1024 and 512 respectively. The size of those feature vectors (MR50-FC1, MR50-FC2) are 50% and 75% less respectively as compared to the feature vectors obtained using ResNet50.
4) The dimension of feature vectors (MR50-FC2) is further reduced to 298 and 37 using mRMR and LDA respectively.
5) mRMR features have shown a drastic decrease in feature size compromising the performance but when the FC2 features are used with LDA, the feature size is further reduced by 98.19% when compared with the ResNet50 feature vector size.
6) Figure 7 gives the visual impression of the performance dominance of the MR50 deep features over other competing methods listed in Table 3.
7) The Average precision(%) Vs Average recall(%) of the proposed MR50 CNN is shown in Figure 8 with features extracted from the pool and fully-connected layers. From the Figure 8, it is clear that the FC2 features using LDA (FC2 - LDA) has the highest area



(a) Dimension of feature vectors for all the competing methods



(b) Retrieval performance (ANMRR and mAP) of all the competing methods

FIGURE 7. Performance comparison of all competing methods.

TABLE 5. Performance metrics of the proposed CNN MR50 features with other competing features, Gabor Texture(using hand-crafted features) and ResNet50 (using deep features).

Method	Feature size	ANMRR	mAP
Gabor Texture (GT)	80	0.6422	0.2769
ResNet50 [37, 55]	2048	0.2606	0.6788
Proposed method			
POOL	2048	0.3785	0.4973
FC1	1024	0.0775	0.8870
FC2	512	0.0704	0.8990
FC2+MRMR	298	0.0623	0.9102
FC2+LDA	**37**	**0.0115**	**0.9802**

under the curve and is the clear winner over the features FC1, FC2, Pool, and FC2 - MRMR.

8) The class-level retrieval performances of the proposed CNN MR50 deep features FC1, FC2, Pool, FC2-MRMR, and FC2-LDA are shown in Figure 10 using the metric 'ANMRR'. A lower value of ANMRR indicates better performance. From Figure 10, it is observed that features computed using the proposed method (MR50 - FC2+LDA) have good performance over all other features across all the classes.

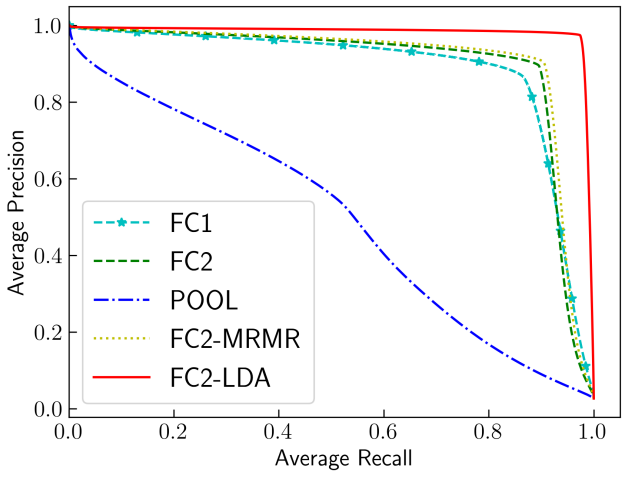


FIGURE 8. Average precision Vs Average recall of all features of MR50.

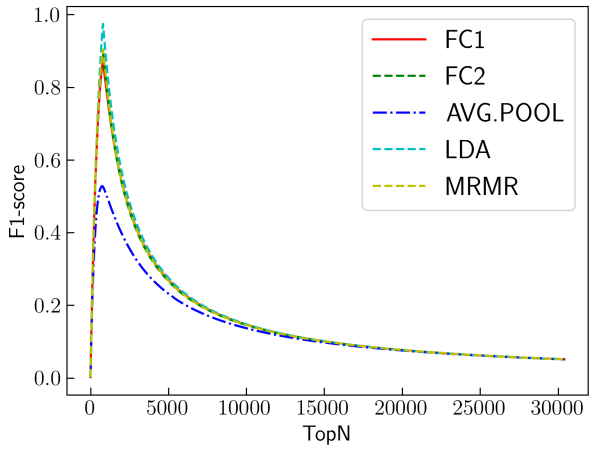


FIGURE 9. F1 score of all features of MR50.

Apart from precision and recall, F1 score is also considered as a comprehensive metric in machine learning and statistics. Generally, precision measures the accuracy of the retrieved results i.e., whether the results retrieved are relevant. On the other hand, recall measures the ability of the system to find all similar instances of a class among all the instances that actually belong to that class. Moreover, increase in precision reduces the recall and vice versa. Therefore, F1 score is used to assess the performance of a model which combines both the precision and recall to a single value. This F1 score is expressed as harmonic mean of precision and recall scores of the system. Higher F1 score indicates that the system is good at retrieving the relevant images while minimizing the irrelevant images. Therefore, F1 score of all the MR50 features are plotted in Fig 9 for all the operating points. From

TABLE 6. Feature vector size, feature extraction time and retreival time of a query image.

Method	Feature size	Feature extraction time(msec)	Retreival time(sec)	Total time (sec)	Computational time complexity
ResNet50	2048	268	3.19	3.458	O(N logN)
MR50-FC1	1024	239	3.21	3.449	O(N logN)
MR50-FC2	512	249	3.21	3.459	O(N logN)
MR50-POOL	2048	238	3.68	3.918	O(N logN)
MR50-MRMR-FC2	**298**	268	2.78	**3.048**	O(N logN)
MR50-LDA-FC2	**37**	305	2.41	**2.715**	O(N logN)

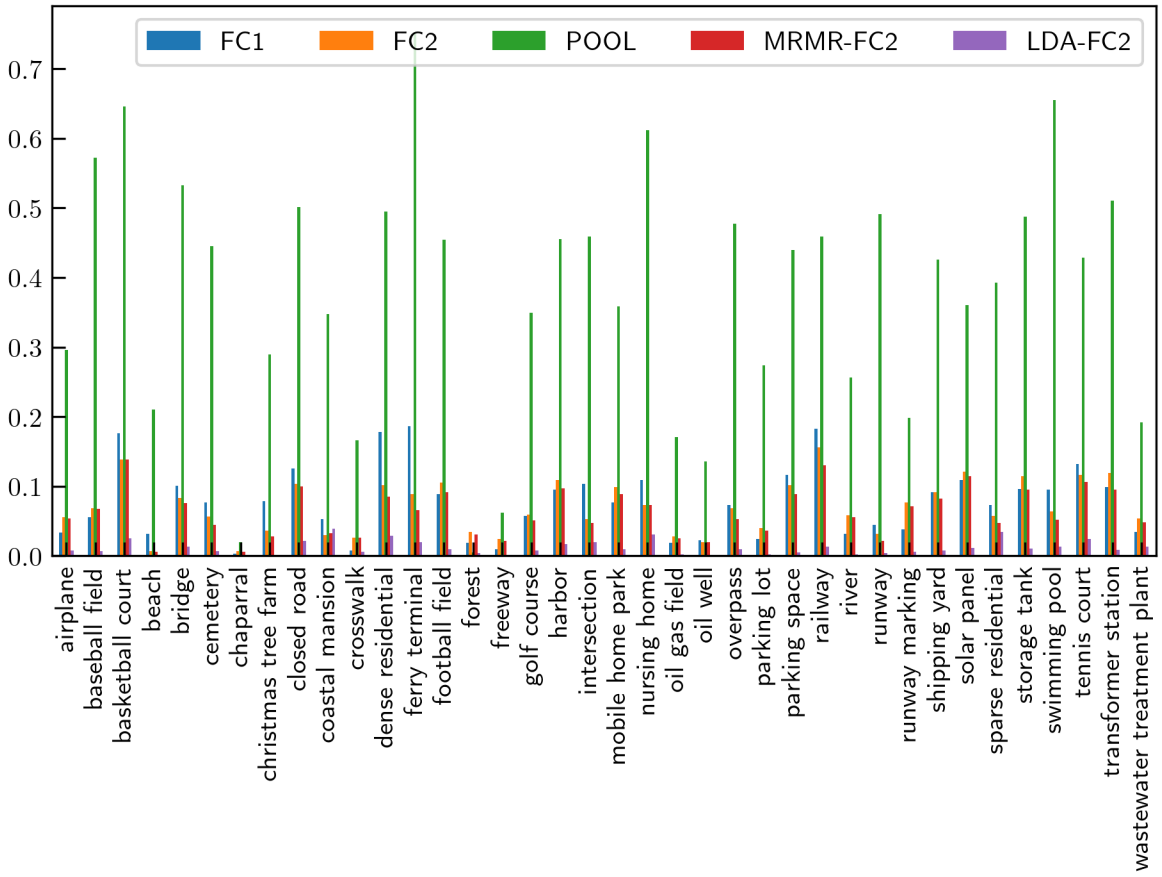


FIGURE 10. ANMRR of all the features of the MR50.

Fig 9, it is evident that the FC2-LDA feature set outperforms all other competing feature sets considered in this study in terms of F1 score.

E. EVALUATION OF RETRIEVAL TIME OF THE PROPOSED METHOD

Table 6, gives an overview of the feature vector length, the time taken by the CPU to search the relevant images along with the computation complexity of the proposed modified ResNet MR50 and the competing method which uses ResNet50 deep features. It can be observed from Table 6 that the feature extraction time of the MR50 and the ResNet50 CNN doesn't vary much. Any traditional distance-based retrieval approach requires a minimum of O(NlogN) comparisons utilizing 'Quicksort' for retrieving similar images against a query image for a database with 'N' number of images and '*C*' number of classes, resulting in a time complexity of O(NlogN). Although, the theoretical time complexity of all the features has O(NlogN), there is a difference in CPU time for searching relevant images for different methods because of the variation in size of the feature vectors. From Table 6, it is observed that the total CPU time of the proposed method using mRMR and LDA features is observed as less compared to others. LDA features

are computationally less expensive as compared to mRMR features because the feature size of LDA is 37 whereas the feature size of mRMR is 298. Although the feature extraction time using MR50 CNN with LDA is higher than others, it is compensated by the retrieval time which is low compared to other features.

V. CONCLUSION

The work presented in this paper is an effort to improve the retrieval performance of the system using the deep features extracted from modified ResNet50 CNN MR50. This modified ResNet50 serves as a powerful deep feature extractor, capturing deep semantic features that encode rich and meaningful information from remote sensing images. The specific modifications applied to the architecture enhance its discriminative power and generalization ability, resulting in improved feature representations. In addition, the integration of Maximum Relevance and Minimum Redundancy (MRMR) and Linear Discriminant Analysis (LDA) for feature reduction further enhanced the retrieval efficiency, preserving the performance of the system intact. The use of deep semantic features in CBRSIR is essential as they capture high-level semantics, enabling a more sophisticated understanding and analysis of remote sensing imagery. These features encode meaningful information related to objects, scenes, and other semantic aspects, improving retrieval performance and facilitating accurate retrieval from large image databases. Experimental evaluations on a remote sensing image dataset 'PatternNet' validate the effectiveness of the proposed approach, demonstrating significant improvements in retrieval efficiency while maintaining retrieval accuracy.

ACKNOWLEDGMENT

The authors thank AI & ML Laboratory, Department of Computer Science and Engineering, SRM University-AP, Andhra Pradesh, for providing the DGX-1 facility to carry out their research work.

REFERENCES

[1] J. Yang, J. Liu, and Q. Dai, "An improved bag-of-words framework for remote sensing image retrieval in large-scale image databases," *Int. J. Digit. Earth*, vol. 8, no. 4, pp. 273–292, Feb. 2014, doi: 10.1080/17538947.2014.882420.

[2] C. Ma, F. Chen, J. Yang, J. Liu, W. Xia, and X. Li, "A remote-sensing image-retrieval model based on an ensemble neural networks," *Big Earth Data*, vol. 2, no. 4, pp. 351–367, Oct. 2018, doi: 10.1080/20964471.2019.1570815.

[3] M. N. Vharkate and V. B. Musande, "Fusion based feature extraction and optimal feature selection in remote sensing image retrieval," *Multimedia Tools Appl.*, vol. 81, no. 22, pp. 31787–31814, Apr. 2022, doi: 10.1007/s11042-022-11997-y.

[4] N. B. Devi, "Satellite image retrieval of random forest (RF-PNN) based probablistic neural network," *Earth Sci. Informat.*, vol. 15, no. 2, pp. 941–949, Feb. 2022, doi: 10.1007/s12145-021-00759-3.

[5] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 8, pp. 837–842, Aug. 1996, doi: 10.1109/34.531803.

[6] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, Jul. 1989, doi: 10.1109/34.192463.

[7] N. Kingsbury, "Image processing with complex wavelets," *Phil. Trans. Roy. Soc. London A, Math., Phys. Eng. Sci.*, vol. 357, no. 1760, pp. 2543–2560, Sep. 1999, doi: 10.1098/rsta.1999.0447.

[8] M. Pietikäinen, T. Ojala, and Z. Xu, "Rotation-invariant texture classification using feature distributions," *Pattern Recognit.*, vol. 33, no. 1, pp. 43–52, Jan. 2000.

[9] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002, doi: 10.1109/TPAMI.2002.1017623.

[10] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2005, pp. 886–893, doi: 10.1109/CVPR.2005.177.

[11] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Computer Vision—ECCV 2006*. Berlin, Germany: Springer, 2006, pp. 404–417.

[12] M. Kokare, P. K. Biswas, and B. N. Chatterji, "Rotation-invariant texture image retrieval using rotated complex wavelet filters," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 36, no. 6, pp. 1273–1282, Dec. 2006, doi: 10.1109/tsmcb.2006.874692.

[13] A. Bosch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," in *Proc. 6th ACM Int. Conf. Image video Retr.*, Jul. 2007, pp. 401–408, doi: 10.1145/1282280.1282340.

[14] S. Murala, R. P. Maheshwari, and R. Balasubramanian, "Local tetra patterns: A new feature descriptor for content-based image retrieval," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2874–2886, May 2012, doi: 10.1109/tip.2012.2188809.

[15] M. Sarker, "Content-based image retrieval using Haar wavelet transform and color moment," *Smart Comput. Rev.*, vol. 3, no. 3, Jun. 2013, pp. 155–165, doi: 10.6029/smartcr.2013.03.002.

[16] Y. Zhao, W. Jia, R.-X. Hu, and H. Min, "Completed robust local binary pattern for texture classification," *Neurocomputing*, vol. 106, pp. 68–76, Apr. 2013.

[17] S. R. Dubey, S. K. Singh, and R. K. Singh, "Rotation and illumination invariant interleaved intensity order-based local descriptor," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5323–5333, Dec. 2014, doi: 10.1109/tip.2014.2358879.

[18] I. J. Jacob, K. G. Srinivasagan, and K. Jayapriya, "Local oppugnant color texture pattern for image retrieval system," *Pattern Recognit. Lett.*, vol. 42, pp. 72–78, Jun. 2014, doi: 10.1016/j.patrec.2014.01.017.

[19] P. Bosilj, E. Aptoula, S. Lefèvre, and E. Kijak, "Retrieval of remote sensing images with pattern spectra descriptors," *ISPRS Int. J. Geo-Inf.*, vol. 5, no. 12, p. 228, Dec. 2016, doi: 10.3390/ijgi5120228.

[20] M. Schröder, M. Walessa, H. Rehrauer, K. Seidel, and M. Datcu, "Gibbs random field models: A toolbox for spatial information extraction," *Comput. Geosci.*, vol. 26, pp. 423–432, May 2000, doi: 10.1016/s0098-3004(99)00122-3.

[21] H. Yao, B. Li, and W. Cao, "Remote sensing imagery retrieval based-on Gabor texture feature classification," in *Proc. 7th Int. Conf. Signal Process. (ICSP)*, 2004, pp. 733–736, doi: 10.1109/icosp.2004.1452767.

[22] S. Newsam, L. Wang, S. Bhagavathy, and B. S. Manjunath, "Using texture to analyze and manage large collections of remote sensed image and video data," *Appl. Opt.*, vol. 43, no. 2, pp. 210–217, Jan. 2004, doi: 10.1364/ao.43.000210.

[23] S. Bouteldja and A. Kourgli, "Multiscale texture features for the retrieval of high resolution satellite images," in *Proc. Int. Conf. Syst., Signals Image Process. (IWSSIP)*, 2015, pp. 170–173.

[24] P. J. Costianes and J. B. Plock, "Gray-level co-occurrence matrices as features in edge enhanced images," in *Proc. IEEE 39th Appl. Imag. Pattern Recognit. Workshop (AIPR)*, Oct. 2010, pp. 1–6, doi: 10.1109/aipr.2010.5759705.

[25] J. G. Daugman, "Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 7, pp. 1169–1179, Jun. 1988, doi: 10.1109/29.1644.

[26] Z. Shao, W. Zhou, L. Zhang, and J. Hou, "Improved color texture descriptors for remote sensing image retrieval," *J. Appl. Remote Sens.*, vol. 8, no. 1, Jul. 2014, Art. no. 083584.

[27] Z. Shao, W. Zhou, Q. Cheng, C. Diao, and L. Zhang, "An effective hyperspectral image retrieval method using integrated spectral and textural features," *Sensor Rev.*, vol. 35, no. 3, pp. 274–281, Jun. 2015, doi: 10.1108/sr-10-2014-0716.

[28] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004, doi: 10.1023/b:visi.0000029664.99615.94.

[29] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, Oct. 2003, pp. 1470–1477.
[30] Y. Yang and S. Newsam, "Geographic image retrieval using local invariant features," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 2, pp. 818–832, Feb. 2013, doi: 10.1109/tgrs.2012.2205158.
[31] A. Ma and I. K. Sethi, "Local shape association based retrieval of infrared satellite images," in *Proc. 7th IEEE Int. Symp. Multimedia (ISM)*, Dec. 2005, p. 7, doi: 10.1109/ism.2005.75.
[32] F. Dell'Acqua and P. Gamba, "Query-by-shape in meteorological image archives using the point diffusion technique," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 9, pp. 1834–1843, Sep. 2001, doi: 10.1109/36.951074.
[33] G. J. Scott, M. N. Klaric, C. H. Davis, and C.-R. Shyu, "Entropy-balanced bitmap tree for shape-based object retrieval from large-scale satellite imagery databases," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 5, pp. 1603–1616, May 2011.
[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Stateline, NV, USA, vol. 25, Dec. 2012, pp. 1097–1105.
[35] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," Sep. 2014, *arXiv:1409.1556*.
[36] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.
[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Dec. 2015, *arXiv:1512.03385*.
[38] S. Agrawal, A. Chowdhary, S. Agarwala, V. Mayya, and S. S. Kamath, "Content-based medical image retrieval system for lung diseases using deep CNNs," *Int. J. Inf. Technol.*, vol. 14, no. 7, pp. 3619–3627, Dec. 2022, doi: 10.1007/s41870-022-01007-7.
[39] M. A. Mohammed, Z. A. Oraibi, and M. A. Hussain, "Content based image retrieval using fine-tuned deep features with transfer learning," in *Proc. 2nd Int. Conf. Comput. Syst., Inf. Technol., Electr. Eng. (COSITE)*, Aug. 2023, pp. 108–113, doi: 10.1109/COSITE60233.2023.10249430.
[40] D. Pathak and U. S. N. Raju, "Content-based image retrieval for super-resolutioned images using feature fusion: Deep learning and hand crafted," *Concurrency Comput., Pract. Exper.*, vol. 34, no. 22, Jan. 2022, doi: 10.1002/cpe.6851.
[41] D. Pathak and U. S. N. Raju, "Content-based image retrieval using feature-fusion of GroupNormalized-inception-darknet-53 features and handcraft features," *Optik*, vol. 246, Nov. 2021, Art. no. 167754, doi: 10.1016/j.ijleo.2021.167754.
[42] Y. Liu, Y. Peng, D. Hu, D. Li, K.-P. Lim, and N. Ling, "Image retrieval using CNN and low-level feature fusion for crime scene investigation image database," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, Nov. 2018, pp. 1208–1214, doi: 10.23919/apsipa.2018.8659471.
[43] F. Ye, W. Luo, M. Dong, D. Li, and W. Min, "Content-based remote sensing image retrieval based on fuzzy rules and a fuzzy distance," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022, doi: 10.1109/LGRS.2020.3030858.
[44] R. Yelchuri, J. K. Dash, P. Singh, A. Mahapatro, and S. Panigrahi, "Exploiting deep and hand-crafted features for texture image retrieval using class membership," *Pattern Recognit. Lett.*, vol. 160, pp. 163–171, Aug. 2022, doi: 10.1016/j.patrec.2022.06.017.
[45] H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han, "Large-scale image retrieval with attentive deep local features," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3476–3485.
[46] R. Cao, Q. Zhang, J. Zhu, Q. Li, Q. Li, B. Liu, and G. Qiu, "Enhancing remote sensing image retrieval using a triplet deep metric learning network," *Int. J. Remote Sens.*, vol. 41, no. 2, pp. 740–751, Jan. 2020, doi: 10.1080/2150704X.2019.1647368.
[47] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "End-to-end learning of deep visual representations for image retrieval," *Int. J. Comput. Vis.*, vol. 124, no. 2, pp. 237–254, Sep. 2017, doi: 10.1007/s11263-017-1016-8.
[48] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," Mar. 2017, *arXiv:1703.07737*.
[49] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," 2015, *arXiv:1503.03832*.
[50] U. Chaudhuri, B. Banerjee, A. Bhattacharya, and M. Datcu, "Attention-driven graph convolution network for remote sensing image retrieval," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022, doi: 10.1109/LGRS.2021.3105448.
[51] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261–2269.
[52] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," 2016, *arXiv:1610.02357*.
[53] W. Zhou, H. Guan, Z. Li, Z. Shao, and M. R. Delavar, "Remote sensing image retrieval in the past decade: Achievements, challenges, and future directions," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 1447–1473, 2023, doi: 10.1109/jstars.2023.3236662.
[54] S. H. S. Basha, S. R. Dubey, V. Pulabaigari, and S. Mukherjee, "Impact of fully connected layers on performance of convolutional neural networks for image classification," *Neurocomputing*, vol. 378, pp. 112–119, Feb. 2020, doi: 10.1016/j.neucom.2019.10.008.
[55] W. Zhou, S. Newsam, C. Li, and Z. Shao, "PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval," *ISPRS J. Photogramm. Remote Sens.*, vol. 145, pp. 197–209, Nov. 2018, doi: 10.1016/j.isprsjprs.2018.01.004.
[56] M. J. Swain and D. H. Ballard, "Color indexing," *Int. J. Comput. Vis.*, vol. 7, no. 1, pp. 11–32, Nov. 1991, doi: 10.1007/bf00130487.
[57] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001, doi: 10.1023/a:1011139631724.
[58] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005, doi: 10.1109/TPAMI.2005.159.

RAJESH YELCHURI received the bachelor's degree in computer science and engineering from JNTU University, India, in 2005, and the master's degree in computer science and engineering from ANU, Guntur, India, in 2013. He is currently pursuing the Ph.D. degree with SRM University-AP, Amaravati, Andhra Pradesh, India. His current research interests include pattern recognition, content-based image retrieval, image processing, and texture analysis.

ALAA O. KHADIDOS received the B.S. degree in computer science from King Abdulaziz University, Jeddah, Saudi Arabia, in 2006, the M.Sc. degree in computer science from the University of Birmingham, Birmingham, U.K., in 2011, and the Ph.D. degree in computer science from the University of Warwick, Coventry, U.K., in 2017. He is currently an Associate Professor with the Faculty of Computing and Information Technology, King Abdulaziz University. His main research interests include computer vision, machine learning, optimization, and medical image analysis.

ADIL O. KHADIDOS received the B.Sc. degree in computer science from King Abdulaziz University, Jeddah, Saudi Arabia, in 2006, the M.Sc. degree in internet software systems from the University of Birmingham, Birmingham, U.K., in 2011, and the Ph.D. degree in computer science from the University of Southampton, Southampton, U.K., in 2017. He is currently an Associate Professor with the Faculty of Computing and Information Technology, King Abdulaziz University. His main research interests include computer swarm robotics, entomology behavior, machine learning, self-distributed systems, and embedded systems.

ABDULRHMAN M. ALSHAREEF received the Ph.D. degree in computer science from the University of Ottawa. He is currently an Associate Professor with the Information System Department, King Abdulaziz University, Jeddah, Saudi Arabia. His research interests include recommender systems, social media mining, data sciences, information assurance artificial intelligence, and e-business technologies.

GANDHARBA SWAIN received the M.C.A. degree from the University College of Engineering, Burla, in 1999, the M.Tech. degree in CSE from the National Institute of Technology, Rourkela, India, in 2004, and the Ph.D. degree in CSE from Siksha 'O' Anusandhan University, Bhubaneswar, India, in 2014. He is currently a Professor with the Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India. He has more than 20 years of teaching experience and authored two books and more than 90 research articles. Many of his articles are published in journals of reputed publishers like Elsevier, Springer, Hindawi, Wiley, and Inderscience. His research interests include security, image tamper detection, and block chain technology.

JATINDRA KUMAR DASH received the bachelor's degree in electronics and communication engineering from the Institution of Engineers, India, in 1999, the Master of Engineering degree in computer science and engineering from the Government College of Engineering, Tirunelveli, India, in 2001, and the Ph.D. degree from the Department of Electronics and Electrical Communication Engineering, IIT Kharagpur, India. He was a Visiting Researcher with the University of California at Berkeley, Berkeley, CA, USA. He was a Research Consultant with the Sponsored Research and Industrial Consultancy (SRIC), IIT Kharagpur. Before this, he was an Assistance Professor and the Head of the Department of Computer Science and Engineering, School of Engineering, Centurion University of Technology and Management, Parlakhemundi. He is currently an Associate Professor with the Department of Computer Science and Engineering, SRM University-AP, India. He has more than 15 years of teaching and three years of research experience. His research interests include image processing, pattern recognition, texture analysis, and medical imaging.