International Journal of Applied Mathematics
Volume 38 No. 4s, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

AN IMPROVISED CONTENT BASED IMAGE RETRIEVAL SYSTEM FOR REMOTE SENSING IMAGES

Nisha Gupta[1], Ajay Mittal[2], and Satvir Singh[3]

[1]Department of Computational Sciences, MRSPTU Bathinda, India, nishasbs2019@gmail.com

(Corresponding author)

[2]Department of Applied Sciences, Aryabhatta Group of Institutes, Barnala, India,

ajay11mittal@gmail.com

[3]Department of Electronics and Communication, IKGPTU Kapurthala, India,
drsatvir.in@gmail.com

Abstract

Image classification is the correctly identifying the objects of an image. Remote sensing image classification is the efficient execution of image categorization of high resolution spatial images for large remote sensing archives. High performance of image categorization known as classification is directly based upon efficient image feature extraction. As information technology and digital devices have advanced billions of individuals are now utilizing the Internet. The volume of digital data in the form of pictures, videos and text grows tremendously every day. One of the biggest concerns is keeping up with the huge online information repositories. Our daily applications are significantly dependent on multimedia content archives on the internet. The digital format is used to store images in web repositories. It can be challenging to find relevant images in the vast multimedia archive. The Remote Sensing Image Retrieval System (RSIR) searches digital images from various remote sensing archives. The user obtains the most relevant and related images from the database based on the attributes of the query image. RSIR has advanced with the development of novel techniques including transfer learning employing pre-trained models and deep learning breakthroughs. Two processes are involved in the Content-Based Image Retrieval System: feature extraction and feature matching. The goal of feature extraction is to reduce the amount of information that describes the total image content using an appropriate distance measure. Feature matching compares the features that were extracted from the database with those that were extracted from the query image. Every database-indexed image is ranked based on how far away it is from the query image. This research goes beyond remote sensing alone. Content Based Image Retrieval System Content Based Image Retrieval (CBIR) could be very useful in forestry, agriculture and geosciences where satellites can acquire images to determine the earth's geological characteristics at a certain location and time. The fields of oceanography, geology, archaeology and astrology are other application areas for the content based image retrieval system.

1. Introduction

With the advancements of computer technology and digital devices billions of people are browsing web. There is an exponential growth of digital data in the form of texts, images and videos every day. Taking care of the huge information archives on web is a significant issue. The web repositories for multimedia data are massively used by our daily life applications. The data in these web repositories is stored in the form of digital images. Storing and retrieving images from the vast multimedia databases is an immense challenge. The searching for digital images in large databases is known as image retrieval. Image Retrieval (IR) is the mechanism by which the images related to query image are searched from the images database and user retrieves the most similar images from the database.

Received: July 20, 2025

There are main three steps in content-based image retrieval system: image representation, image organization, and image similarity measurement. CBIR is to retrieve information from the very big visual database. Thus organizing the large scale database efficiently and identifying the relevant results of a given query within acceptable time limits and storage demands is the main goal of image retrieval system. CBIR system employs various feature extraction techniques to capture the features of image forming a feature vector. Image is commonly processed by its color, shape and texture features known as local descriptors of image. There are three main steps for retrieving images in CBIR system; local feature extraction, embedding and aggregation. Local features extraction is the first step e.g. Scale Invariant Feature Transform (SIFT) is used to extract features of an image. In the second embedding step, the extracted features are mapped into a high dimensional feature vector comprising of each extracted local feature of image. In aggregation step the high dimensional features are mapped into a single vector e.g. Vector of Locally Aggregated Descriptor (VLAD), finds a single vector to represent each local vector identifying an image. The mapping of high dimensional vector into a single vector is single vector is known as pooling. The aggregation step helps in reducing the storage requirement for storing a huge data of images in the image database. The aggregated vector is directly used to retrieve similar images from the database. Thus the aggregated representation decreases the storage space requirement as well as it allows the distance calculation between query image and database sample images. Big data needs very compact representation for storage and very fast searching. Binary hashing is used to generate compact binary codes for images.(Do and Cheung [2017]). Binary hashing may be categorized into data independent and data dependent hashing. Data dependent hashing uses training data available for learning hash functions. No training data is available in case of data independent hashing to learn hash functions .

2. Literature Survey on Salient Feature Extraction Techniques

Feature extraction is the transformation of human perception into a numerical representation in form of a feature vector manipulated by machines. This is the prime most and most crucial stage in choosing representative features of a remote sensing image retrieval system design. The dimensionality reduction procedure can effectively represent just significant image features as a comparative lower level feature vector. Researchers use a variety of feature descriptors to characterize an image's visual content as a low dimensional feature vector (Zhuo, B. Cheng, and Jing Zhang [2014]). Major findings using vaious feature descriptors are shown in table1.

Table 1: Recognition Results for Feature Extraction Techniques

Global Feature Descriptors						
Sr. No.	**Authors**	**Technique**	**Dataset**	**Evaluation Metrics**	**Score**	**Limitations**
1	(Pavithra and Sree Sharmila [2019])	DCD	Wang	Accuracy	73%	Do not bridge semantic gap
			Oxford	Accuracy	41%	
			Oxford flowers	Accuracy	31%	
2	(Rana, Dey, and Siarry [2019])	Color Moment + Ranklet + Invariant Moment	Simplicity	Accuracy	67%	High computational cost due to high feature vector dimension
			Corel 5k	Accuracy	67%	
			Corel 10k	Accuracy	67.96%	
			Caltech101	Accuracy	64.50%	
			MSR	Accuracy	64.80%	

Received: July 20, 2025

3	(Bella and Vasuki [2019])	Color Moment + GLCM + Geometric Shape Feature	Corel 1k	Accuracy	83.30%	Retrieval time can be reduced with a suitable optimization
			Corel 5k	Accuracy	66.90%	
			Corel 10k	Accuracy	56.40%	
4	(Tadi Bani and Fekri-Ershad [2019])	Quantized Color Histogram + Gabor Filter + GLCM	Simplicity	Accuracy	82%	High run time
5	(Alsmadi [2020])	GLCM + DWT + Canny Edge Histogram	Corel	Average Precision	90.15%	Criticality of cooling process. Increased calculation time
				Average Recall	18.03%	
				Accuracy	90%	
6	(Ashraf et al. [2020])	Color Moment + DWT/Gabor Filter/CEDD	Corel1k	Accuracy	87.50%	High computational cost due to high feature vector dimension
			Corel1.5k	Accuracy	86.33%	
			Corel5k	Accuracy	79.83%	
			GHIM20k	Accuracy	76.50%	
Local Feature Descriptors						
Sr. No.	**Authors**	**Technique**	**Dataset**	**Evaluation Metrics**	**Score**	**Limitations**
7	(Sharif et al. [2019])	SIFT, BRISK	Corel 1k	Accuracy	84.39%	Not tested against large-scale unlabeled datasets
			Corel 1.5k	Accuracy	78.14%	
			Corel 5k	Accuracy	57.37%	
			Caltech 256	Accuracy	47.52%	
8	(Agarwal, Singhal,and Lall [2019])	MCLTP	Corel 1k	Accuracy	83.30%	Feature vector dimension is 3072
			Corel 10k	Accuracy	0.569	
			CMU-PIE	Accuracy	0.863	
			STex	Accuracy	0.981	
			MIT VisTex	Accuracy	0.409	
9	(Sarwar et al.[2019])	LBPV, LIOP	Wang 1k	Accuracy	89%	Not tested on large-scale datasets; not suitable for multispectral images
			Wang 1.5k	Accuracy	76%	
			Holidays	Accuracy	69%	
10	(Baig et al. [2020])	CoHOG, SURF	Corel 1k	Accuracy	86.40%	Not tested on large-scale datasets
			Corel 5k	Accuracy	77.20%	

Received: July 20, 2025

			Sense 15	Accuracy	81.20%	
			Caltech 256	Accuracy	68.30%	

Hashing Based Feature Extraction

Sr. No.	Authors	Technique	Dataset	Evaluation Metrics	Score	Limitations
11	(Do, Le, et al.[2019])	Relaxed Binary Auto-encoder (RBA)	CIFAR 10	MAP	53.17%	Optimization at each step adds overhead
			MNIST	MAP	75.48%	
			NUS-WIDE	MAP	64.01%	
		Simultaneous Aggregation and Hashing (SAH)	CIFAR 10	MAP	77.22%	
			MNIST	MAP	63.31%	
			NUS-WIDE	MAP	45.05%	
12	(Liu et al. [2020])	SBS-CNN (Similarity-based Siamese CNN)	UC-Merced	ANMR	21.85%	ReLU limits training; gradient descent adds error overhead
13	(J. Wang et al.[2020])	OSA-HSR + CNN	Ohio State aerial imagery	Average computation time	501.66 ns	Stochastic gradient evaluation increases runtime

Graph Based Feature Descriptors

Sr. No.	Authors	Technique	Dataset	Evaluation Metrics	Score	Limitations
14	(B. Chaudhuri,Demir , Bruzzone,et al. [2016])	ARG (Attributed Relational Graph)	50 randomly selected images per category (k=232)	Precision	59.76%	High computational time in graph construction (81.75s per match)
				Average Precision	70%	
				Average ANMR	0.57%	
15	(B. Chaudhuri,Demir ,S. Chaudhuri, et al.[2017])	Multilabel Image Retrieval Method (MLIRM)	UC-Merced	Accuracy	74.29%	Accuracy sensitive to segmentation and region features; high computation time
				Precision	85.68%	
				Recall	80.25%	

Convolution Based Feature Extraction

Sr. No.	Authors	Technique	Dataset	Evaluation Metrics	Score	Limitations
16	(Alzu’bi,	CNN	Oxford 5k	Accuracy	0.957	Accuracy

Received: July 20, 2025

	Amira,and Ramzan [2017])		Oxford 105k	Accuracy	0.886	decreases with larger datasets
17	(Tzelepi and Tefas [2018])	CNN / Unsupervised CNN / Supervised	Paris 6k	Accuracy	0.9859	More retrieval time
			UK Bench	Accuracy	0.8347	
18	(Zheng et al. [2019])	CNN	Caltech 101	Accuracy	0.885	Needs faster training/testing; gravitational field database takes time
			Holiday	Accuracy	0.941	
			Oxford	Accuracy	0.962	
19	(Sezavar, Farsi, and Mohamadzadeh [2019])	CNN	Corel	Accuracy	0.9559	More retrieval time without sparse representation
			ALOI	Accuracy	0.9706	
			MPEG7	Accuracy	0.7749	

3. Major Research Issues in Remote Sensing Image Retrieval System

- Appropriate feature extraction is the main research issue in CBIR. Extracting most significant key-points of query image as well as each sample of dataset image is the concerning research issue. Retrieval performance is directly affected by the quality of features extracted.
- Dimensionality reduction of feature vectors of images is the concerning issue as after reduction visual features representing images may not lose their context. Low level feature vector should not incorporate noisy or inappropriate features.
- Selection of appropriate retrieval scheme , retrieval or classification methods directly affect the retrieval accuracy in context of storage demands and computational time. Managing storage demands and computational time is again the research issue.
- Managing retrieval time within acceptable limits is again the main issue. As time consumed in matching query image features with dataset image directly affects the classification and retrieval performance of RSIR system. Managing retrieval time is the main issue.

4. Candidate Datasets for Remote Sensing Image Classification and Retrieval suitable for Deep Learning Implementations

Performance evaluation and the use of Remote Sensing Image Retrieval (RSIR) techniques are based on benchmark datasets. Experiments may only be carried out if the right dataset is chosen (G. Cheng, Han, and Lu [2017]). Four remote sensing datasets—UC-Merced (Yang and Newsam [2010]), AID (G.-S. Xia et al. [2017]), NWPU-RESISC45 (G. Cheng, Han, and Lu [2017]), and PatterNet (W. Zhou et al. [2018])(W. Zhou et al. [2017])—with unisource retrieval may be deemed highly appropriate for remote sensing image classification and retrieval systems following a thorough examination of their characteristics. Table 2 provides an overview of these datasets' attributes.

Table 2: Proposed Remote Sensing Datasets

Dataset Selected	Classes in Dataset	Total Images in Dataset	Image Size (Pixels)	Spatial Resolution (m)
UC-Merced (Yang and Newsam [2010])	21	2100	256×256	0.3
AID (G.-S. Xia et al. [2017])	30	10000	600×600	8-0.5
NWPU-RESISC45 (G. Cheng,Han, and Lu [2017])	12	31500	256×256	30-0.2
PatterNet (W. Zhou et al.[2018])	38	30400	256×256	4.69-0.06

5. Pre-trained models employing Transfer Learning

Neural network trained on large dataset gains knowledge from this data and this acquired knowledge termed as weights of the network (Krizhevsky, Sutskever, and Hinton [2012]). Only the learned features in the form of weights can be extracted and then transferred to any other neural network instead of training that neural network from the initial stage. Instead of building a model from scratch pre-trained models are trained on large dataset. Pre-trained models act as a feature extractor by removing the output layer (Zhen et al. [2020]). Finally entire network is converted into fixed feature extractor by freezing the initial layers weights while retraining only higher layers for new problem specific dataset (Y. Wang et al. [2021]).

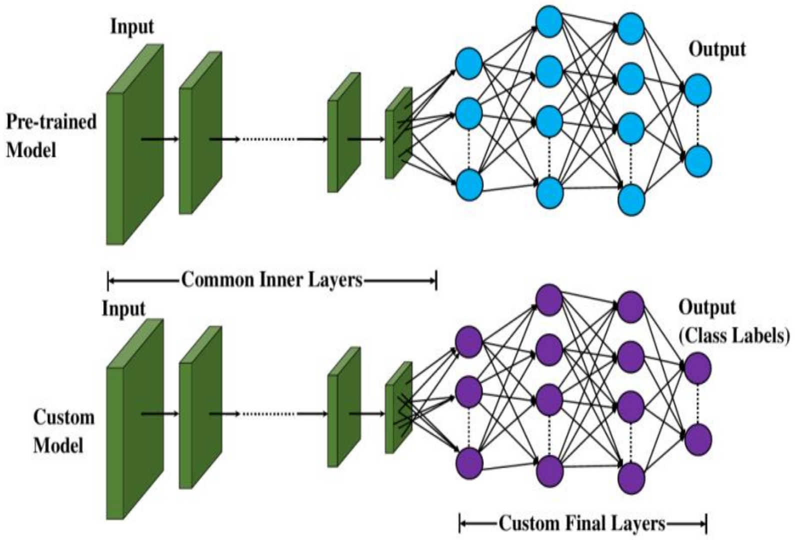

Figure 1: Transfer Learning

6. Pre-Trained Models

Pre-Trained CNN models are Neural Networks trained on a very larger dataset like ImageNet for general image recognition and are availed to be used for other applications. These models are learned to recognize a wider range of features as well patters in images (Krishna and Kalluri [2019]). Pre-Trained models employ Transfer Learning that is pre-Trained models are used as a starting point for the applications even if the application area is different from original training performed on Pre-Trained model (Muhammad et al. [2018]). Thus Pre-Trained models save the time as well as computational resources that could be otherwise required to train the Neural Network from the starting point (Risojevi´c and Stojni´c [2021]). Fine tuning the Pre-Trained models for a particular application requires comparatively lesser data for training known as Transfer Learning.

7. Proposed Pre-Trained Models

Proposed pre-trained models are VGG-19,Inception-v3 and variants of DenseNet.

7.1. VGG-19

RGB input image of the size 224 × 224 is fed to VGG-19 in the form of input matrix of shape 224, 224, 3. The image is pre-processed by rescaling the image (rescale=1.0/255, zoom range=0.2, width shift range=0.2, height shift range=0.2).This pre-processing step is repeated for whole training set images. VGG-19 uses 3 × 3 filters with stride 1. These filters convolute over the whole image for all training sample images. To preserve the spatial resolution of images padding is used. Max pooling uses 2 × 2 pixel windows with stride 2. Maxpooling is followed by activation function ReLu for classification at convolution output layer (Krishnapriya and Karuna [2023]).

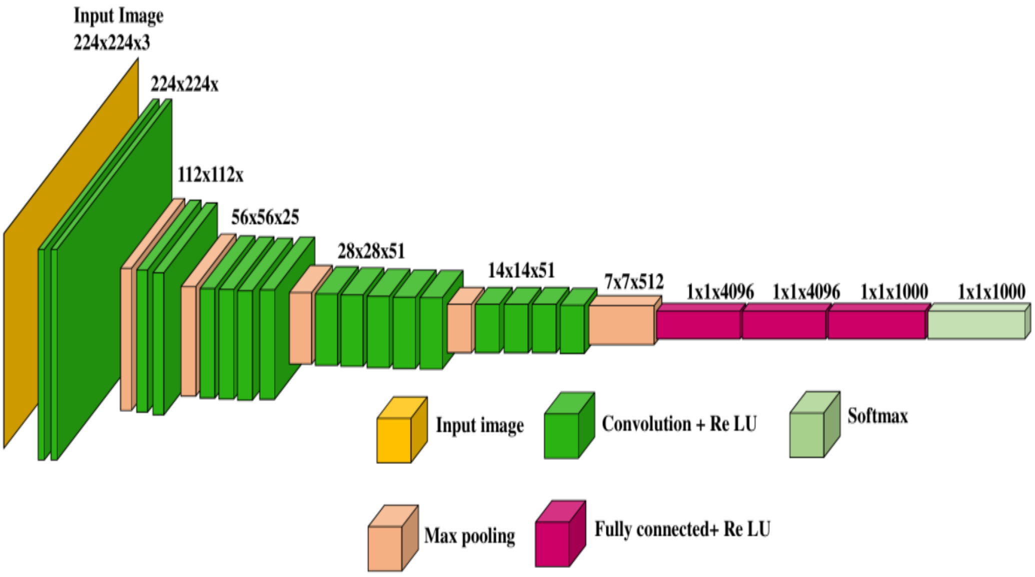

Figure 2: Architecture of VGG-19

First two convolution layers use 3 × 3 filters. Next two layers have 64 filters those results in 224 × 224 × 64 volume. The filters are always 3 × 3 with stride of 1. Pooling layer is used with max-pool of 2 × 2 window size, stride 2. Pooling reduces height and width from 224 × 224 × 64 to 112 × 112 × 64. This is followed by two more convolution layers having 128 filters. This results in the reduced dimension of 112 × 112 × 128. Pooling layer reduces the dimensions to 56 × 56 × 128. Two more convolution layers are added with 256 filters each followed by down sampling layer that reduces the size to 28 × 228 × 56 (Bagaskara and Suryanegara [2021]). Two more stack each with 3 convolution layer is separated by a max-pool layer. 7 × 7 × 512 dimension is flattened into Fully

Connected (FC) after pooling layer. Thus at the last is the resultant reduced feature vector of extracted features. In the end, there are three fully connected layers responsible for classification. First two are of size 4096 with 1000 channels for classification and the third final layer has a softmax function. Softmax can classify 1000 object categories as VGG-19 is pre-trained on ImageNet large scale dataset having 1000 classes of various categories of images as presented in 3 (Marmanis et al. [2015]).

7.2. Inception-v3

Inception-v3 is made up of both symmetric and asymmetric building blocks. It includes convolutions, average pooling, max pooling, concatenations, dropouts and fully connected layers. Batch normalization is applied to activation inputs to make the network stable. Loss is computed using Softmax at the last layer. After performing factorization into Smaller convolutions, Spatial factorization into asymmetric convolutions, utilizing auxiliary classifiers, efficient grid size reduction , the final Inception-v3 model is represented in figure 3.

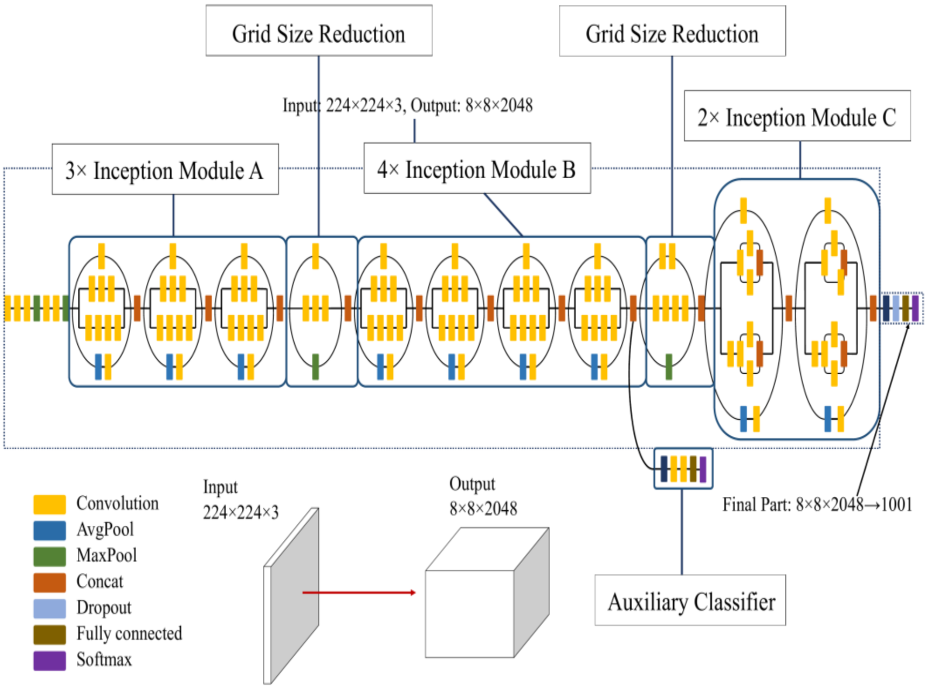

Figure 3: Inception-v3 Main Architecture

7.3. DenseNet

In DenseNet architecture, each layer is directly connected with every other layer thus named as Densely Connected Network. For L number of layers, DenseNet is having L(L+1)/2 direct and shorter connections between layers close to input and output proves to be much efficient to train and achieves much more accuracy. DenseNet is a feed forward network in which each layer is directly connected to the front layers. Input of i^{th} layer can be the output of $(i - 1)t^{h}$ layer and $(i - 2)^{th}$ and $(i - n)^{th}$ layer. Direct connection of each layer in the network uses Batch Normalization to normalize the input of each layer that reduces the absolute difference between data. Connection between all the layers ensures the maximum information flow. Additional inputs from all preceding layers passes on its features maps to all subsequent layers as presented in figure4.

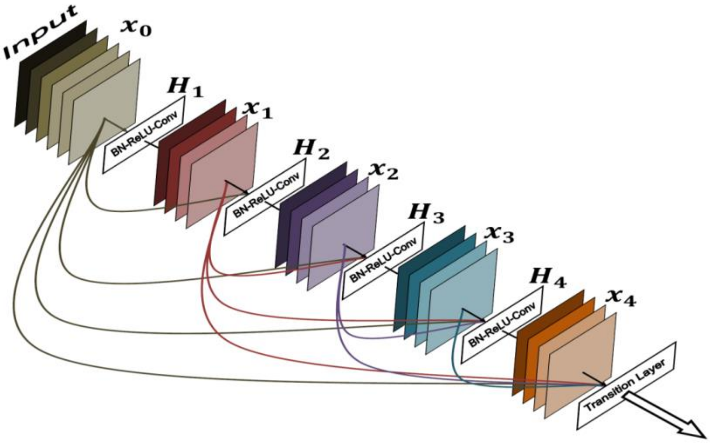

Figure 4: DenseNet Block Diagram

A. Image Classification

Image classification is the correctly identifying the objects of an image. Remote sensing image classification is the efficient execution of image categorization of high resolution spatial images for large remote sensing archives. High performance of image categorization known as classification is directly based upon efficient image feature extraction.

8. Experimental Environment

Open source library used: Tensor Flow and Keras , Input image: 224 × 224 (After pre-processing), Processing platform: Google Colab utilizing Nvidia Tesla T4 graphics card, Bach size: 32, Epochs: 100, steps per epoch: 40, validation steps: 10, Validation steps: 20 or 10 during each epoch, Learning rate: 0.1, Output activation function: Softmax, Loss function: Categorical cross entropy, Data split ratio : 70%(training), 20%(validation), 10%(testing), RAM: 12GB

9. General Flow of Transfer Learning Algorithm for Remote Sensing Image Classification

.General flow of transfer learning is presented in figure 6 and details of training testing stage is depicted in figure 7 .

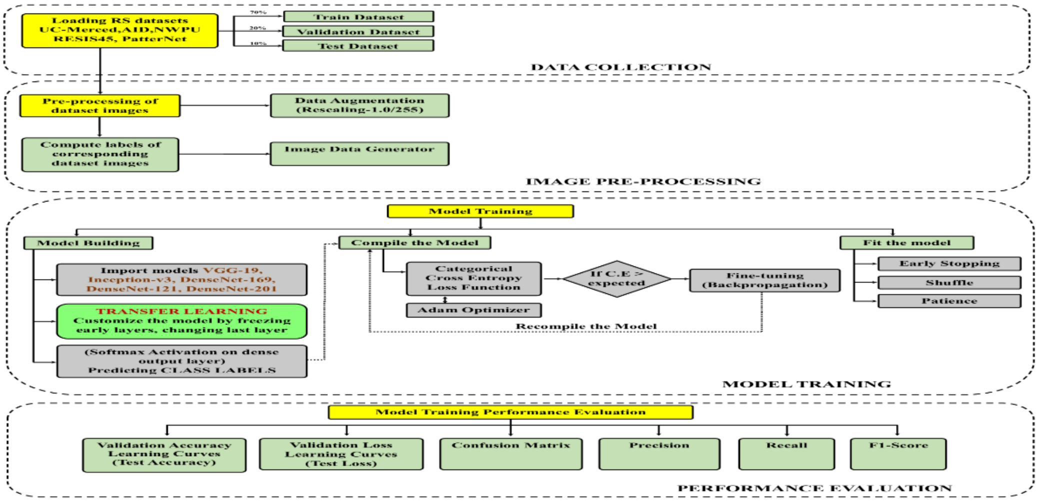
Loading RS datasets UC-Merced,AID,NWPU RESIS45, PatterNet
70%
20%
10%
Train Dataset
Validation Dataset
Test Dataset
DATA COLLECTION
Pre-processing of dataset images
Data Augmentation (Rescaling-1.0/255)
Compute labels of corresponding dataset images
Image Data Generator
IMAGE PRE-PROCESSING
Model Training
Model Building
Import models VGG-19, Inception-v3, DenseNet-169, DenseNet-121, DenseNet-201
TRANSFER LEARNING Customize the model by freezing early layers, changing last layer
(Softmax Activation on dense output layer) Predicting CLASS LABELS
Compile the Model
Categorical Cross Entropy Loss Function
Adam Optimizer
If C.E > expected
Fine-tuning (Backpropagation)
Recompile the Model
Fit the model
Early Stopping
Shuffle
Patience
MODEL TRAINING
Model Training Performance Evaluation
Validation Accuracy Learning Curves (Test Accuracy)
Validation Loss Learning Curves (Test Loss)
Confusion Matrix
Precision
Recall
F1-Score
PERFORMANCE EVALUATION

Figure 5: Remote Sensing Image Classification based on Transfer Learning

Received: July 20, 2025

10. Evaluation Metrics

Evaluating the algorithms is an essential part of any research. Well known evaluation metrics used by researchers are: Accuracy, Precision, Recall and F1-Score.

10.1. Accuracy

Accuracy is the ratio of number of predictions by total number of predictions. Accuracy is the fraction of predictions our model got right. Accuracy indicates number of actual positive and negative outcomes divided by total number of samples.

$$\textbf{Accuracy} = \frac{\textbf{True Positives+True Negatives}}{\textbf{Total Samples}}$$

10.2. Precision

Precision indicates the ratio of actual positive results by the number of positive results as predicted by the classifier.

$$\textbf{Precision} = \frac{\textbf{True Positives}}{\textbf{Total Positives+False Positives}}$$

10.3. Recall

Recall indicates the ratio of correct positive results by the number of all actual samples.

$$\textbf{Recall} = \frac{\textbf{True Positives}}{\textbf{True Positives+False Negatives}}$$

10.4. F1-Score

F1-Score is the Harmonic Mean between precision and recall.

$$\textbf{F1 score} = \frac{\textbf{2P}\times\textbf{R}}{\textbf{P+R}}$$

The range for F1-Score is [0, 1].It indicates how precise the used classifier is inferring that how many instances classifier classifies correctly and it does not miss a significant number of instances.F1 Score is the balance between precision and recall. Lower recall but high precision gives an extremely accurate results but with the limitation that it misses a large number of instances that are difficult to classify. The greater values of the accuracy, precision, Recall and F1 Score the better is the performance of the adopted model.

11. Summary of experimental results performed for Image Classification

Table 3: Investigations on Test Accuracy and Test Loss on Proposed Datasets

UC-Merced		
Pre-Trained Models	**Test Accuracy**	**Test Loss**
VGG-19	0.8952	0.3518
Inception-v3	0.9095	1.9066
DenseNet-169	0.9048	1.2614
DenseNet-121	0.9381	0.4508
DenseNet-201	0.9429	1.3238

Received: July 20, 2025

AID Dataset		
Pre-Trained Models	**Test Accuracy**	**Test Loss**
VGG-19	0.8213	0.5739
Inception-v3	0.8004	3.458
DenseNet-169	0.8769	2.0396
DenseNet-121	0.8838	1.6608
DenseNet-201	0.8669	2.7096
NWPU-RESISC45 Dataset		
Pre-Trained Models	**Test Accuracy**	**Test Loss**
VGG-19	0.9228	0.7884
Inception-v3	0.91	1.254
DenseNet-169	0.9325	0.8443
DenseNet-121	0.9228	0.7884
DenseNet-201	0.9275	0.8956
PatterNet Dataset		
Pre-Trained Models	**Test Accuracy**	**Test Loss**
VGG-19	0.8461	0.5788
Inception-v3	0.9095	2.1474
DenseNet-169	0.9632	1.0238
DenseNet-121	0.9727	0.6076
DenseNet-201	0.976	0.6402

Table 4: Investigations on Precision, Recall and F1-Score on Proposed Datasets

UC-Merced			
Pre-Trained Models	**Precision**	**Recall**	**F1-Score**
VGG-19	0.9008	0.9008	0.8905
Inception-v3	0.9204	0.9103	0.9103
DenseNet-169	0.9104	0.9002	0.9005
DenseNet-121	0.9501	0.9405	0.9405
DenseNet-201	0.9504	0.9402	0.9402
AID			
Pre-Trained Models	**Precision**	**Recall**	**F1-Score**

VGG-19	0.8303	0.8202	0.8202
Inception-v3	0.8305	0.8002	0.8002
DenseNet-169	0.8903	0.8802	0.8801
DenseNet-121	0.9003	0.8801	0.8802
DenseNet-201	0.8901	0.8704	0.8704
NWPU-RESISC45			
Pre-Trained Models	**Precision**	**Recall**	**F1-Score**
VGG-19	0.8604	0.8604	0.8604
Inception-v3	0.9203	0.9102	0.9101
DenseNet-169	0.9301	0.9304	0.9304
DenseNet-121	0.9303	0.9201	0.9201
DenseNet-201	0.9302	0.9303	0.9302
PatterNet			
Pre-Trained Models	**Precision**	**Recall**	**F1-Score**
VGG-19	0.8904	0.8501	0.8501
Inception-v3	0.9202	0.9105	0.9105
DenseNet-169	0.9704	0.9604	0.9604
DenseNet-121	0.9705	0.9705	0.9705
DenseNet-201	0.9801	0.9801	0.9802

Table 3 and table 4 demonstrates the following points:

- PatterNet dataset achieved the best accuracy score of 0.9760 with the DenseNet-201 Pre-trained model.
- The PatterNet dataset achieved the highest precision of 0.9801, recall of 0.9801 and F1-score of 0.9802 on DenseNet-201 Pre-trained model.

B. Image Retrieval

Content based image retrieval system encompasses two steps: Feature extraction and Feature Matching. Extraction of features aims to decrease amount of data describing the overall image content. Feature matching involves the comparing of extracted features from the query image to the extracted features from the database using a suitable distance metric. Each indexed image from the database is ranked according to its computed distance from the query image.

12. Environmental Setup for Image Retrieval on Google Colab with T4 GPU

Software Environment is TensorFlow version: 2.15.0, Keras version: 2.15.0, scikit-learn version: 1.2.2, numpy version is: 1.23.5, cv2 : 4.8.0, matplotlib : 3.7.1

12.1 Euclidean Distance

Euclidean distance represents the line segment length between two points having Cartesian coordinates. If p and q to be two points present on the real line, then the Euclidean distance between two points is: $d(p,q) = |p - q|$. Generalized form for higher dimensions:

$$\boldsymbol{d(p,q) = \sqrt{(p-q)^2}} \qquad (1)$$

Squaring and taking the square root replaces any negative number by its absolute value.

$$\boldsymbol{d(p,q) = \sqrt{(p_1-q_1)^2 + (p_2-q_2)^2 + \cdots + (p_i-q_i)^2 + \cdots + (p_n-q_n)^2}} \qquad (2)$$

The distance can be computed using the points given by polar coordinates. If the polar coordinates of point P are (r, θ) and Q are(s, ψ), then the distance between these points is given by the cosine law:

$$\boldsymbol{d(p,q) = \sqrt{r^2 + s^2 - 2rs\ \cos(\theta - \psi)}} \qquad (3)$$

Where, p and q are complex numbers.

Euclidean distance interprets the distance between query image and retrieved images. Smaller distance indicates the greater similarity between query image and retrieved images and vice a versa.

12.2 Cosine Similarity

Cosine similarity is the measurement metric to quantify the similarity between two or more vectors. It measures the cosine value of angle between two non-zero vectors projected in multi-dimensional space. Cosine similarity is the mathematical representation of division between the dot product of vectors and the product of the Euclidean norms or magnitude of each vector.

$$\boldsymbol{Similarity = cos(\theta) = \frac{A.B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}},} \qquad (4)$$

$A = point\ P1$ and $B = point\ P2$ considering two axis X, Y . Value of cosine similarity is bounded in the range of 0,1. Cosine Distance between two different points is represented in equation 17.

$$Cosine\ Distance\ = 1\ -\ Cosine\ Similarity$$

Case 1: If angle between points $P1, P2$ is 45°

Then $Cosine\ Similarity =\ Cos\ 45\ =\ 0.525, Cosine\ Distance\ =\ 1 - 0.525 =\ 0.475$

Case 2: If the two data points $P1, P2$ are aparted from each other and angle between points is 90°.

Then $e\ Similarity =\ Cos\ 90\ =\ 0\ , Cosine\ Distance\ =\ 1 - 0 = 1$

Case 3: If two points $P1, P2$ are very near and lies on same axis to each other .The angle between points is 0°

Received: July 20, 2025

Then $Cosine\ Similarity = \ Cos\ 0\ = \ 1, Cosine\ Distance\ = \ 1 - 1 = 0$

Case 4: If points $P1, P2$ lies opposite two each other and angle between points is 1800

Then $Cosine\ Similarity = \ Cos\ 180\ = \ -1, Cosine\ Distance\ = \ 1 - (-1)\ = 2$

Case 5: If angle between points $P1, P2$ is270°.

Then Cosine $Similarity = \ Cos\ 270\ = \ 0, Cosine\ Distance\ = \ 1 - 0 = 1$

Case 6: If angle between points $P1, P2$ is 360°.

Then Cosine $Similarity = \ Cos\ 360\ = \ 1, Cosine\ Distance\ = \ 1 - 1 = 0$

12.3 Structured Similarity Index Metric (SSIM)

The $SSIM$ is a metric used to evaluate similarity between a test image X with a reference image Y. $SSIM$ evaluates the similarity between two images by computing a local spatial index. X and Y are the images to be compared. X and Y images are in form of matrices of pixels that is x={$xi|i = 1,2, \dots, N$} and $y = \{yi|i = 1,2, \dots, N\}$. x and y are located at the same spatial position in both images. SSIM is defined in terms of the average pixel values, μx and μy. Pixel values have the standard deviations (SD) σx and σy at x and y and covariance σxy of x and y through the following indexes:

$$\boldsymbol{I(x,y) = (2\mu_x\mu_y + C_1)/(\mu_x^2 + \mu_y^2 + C_1)}, \quad (5)$$

$$\boldsymbol{I(x,y) = (2\sigma_x\sigma_y + C_2)/(\sigma_x^2 + \sigma_y^2 + C_2)}, \quad (6)$$

$$\boldsymbol{I(x,y) = (2\sigma_{xy} + C_3)/(\sigma_x\sigma_y + C_3)}, \quad (7)$$

$$\boldsymbol{SSIM(x,y)} = \frac{(2\mu_x\mu_y+C_1)(2\sigma_{xy}+C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)} \quad (8)$$

In equation 9.13 $C1$, $C2$, and $C3$ are constants used to avoid the instabilities if$(\mu 2x + \mu 2y)$, $(\sigma 2x + \sigma 2y)$, or $\sigma x\sigma y$ gets closer to zero.

- The $l(x,y)$ index is the luminance differences,
- $c(x,y)$ with contrast differences
- $r(x,y)$ with structure variations between x and y.

The general form of the $SSIM$ index is represented in equation

$$\boldsymbol{SSIM(x,y) = [I(x,y)]^{\alpha}.[C(x,y)]^{\beta}.[r(x,y)]^{\gamma}}, \quad (9)$$

where α, β, and γ are parameters that define the relatively importance of each component. SSIM (x,y) ranges from 0 that is completely different or dissimilar to 1 that is completely similar. Finally, a mean $SSIM$ index is computed to evaluate the global image similarity.$SSIM$ puts everything in a scale of$-1\ to\ 1$. SSIM score 1 means the two images are very similar .Score of -1 means that the two images are dissimilar. $SSIM$ measures the similarity between two images by considering luminance, contrast and structure information. The $SSIM$ index produces a score between -1 and 1, where 1 indicates perfect similarity. It is particularly useful when assessing the impact of compression, noise, or other distortions on the perceived quality of an image.The $SSIM$ algorithm compares local patterns of pixel intensities rather than relying solely on global statistics.

Received: July 20, 2025

13. Proposed Method

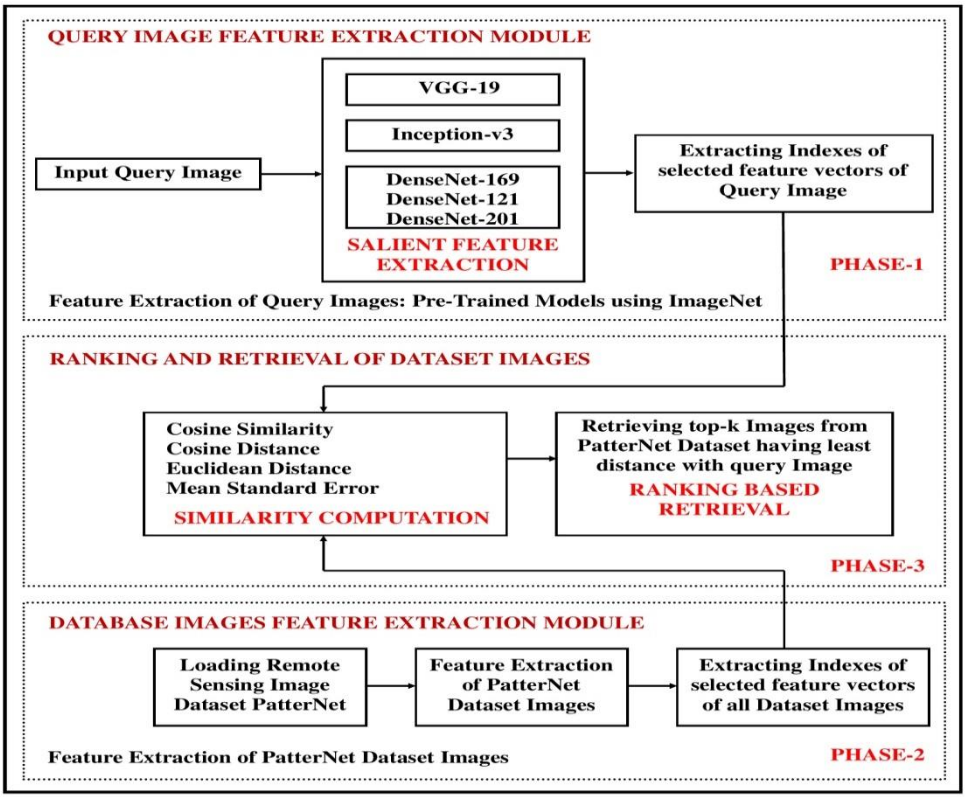
QUERY IMAGE FEATURE EXTRACTION MODULE
Input Query Image
VGG-19
Inception-v3
DenseNet-169
DenseNet-121
DenseNet-201
SALIENT FEATURE EXTRACTION
Extracting Indexes of selected feature vectors of Query Image
PHASE-1
Feature Extraction of Query Images: Pre-Trained Models using ImageNet
RANKING AND RETRIEVAL OF DATASET IMAGES
Cosine Similarity
Cosine Distance
Euclidean Distance
Mean Standard Error
SIMILARITY COMPUTATION
Retrieving top-k Images from PatterNet Dataset having least distance with query Image
RANKING BASED RETRIEVAL
PHASE-3
DATABASE IMAGES FEATURE EXTRACTION MODULE
Loading Remote Sensing Image Dataset PatterNet
Feature Extraction of PatterNet Dataset Images
Extracting Indexes of selected feature vectors of all Dataset Images
Feature Extraction of PatterNet Dataset Images
PHASE-2

Figure 6: Proposed Image Retrieval System

13.1 Experimental Results for Image Retrieval using Proposed Pre-Trained Models

Using Deep Learning with PatterNet dataset using pre-trained models VGG-19, Inception-v3, DenseNet-169, DenseNet-121 and DenseNet-201 this work offers an image retrieval system that utilizes the structured similarity index, euclidean distance and cosine distance. The findings demonstrate that for remote sensing image retrieval systems, pre-trained models outperformed Convolution Neural Networks and traditional machine learning approaches. Test accuracy scores for these pre-trained models show that DenseNet-169 scored 87.69% on AID dataset, DenseNet-201 scored 94.29% on UC-Merced dataset, DenseNet-169 scored 93.25% on NWPU-RESISC45 dataset, DenseNet-201 scored 97.60% on PatterNet dataset for image classification. Thus, PatterNet dataset scored higher than all datasets so retrieval is performed on PatterNet dataset using pre-trained models. In conclusion the suggested image retrieval system searches and sorts the PatterNet dataset which is a very big remote sensing dataset into related images (top-k, k=10). The image retrieval system with the highest Structural Similarity Index value for the highest ranking images is DenseNet-201.

Table 5: Experimental Results for Image Retrieval: VGG-19

VGG-19: COSINE DISTANCE , EUCLIDEAN DISTANCE				
Retrieved Image Classes	**Cosine Similarity Score**	**Cosine Distance= 1-Cosine Similarity**	**Retrieved Image Classes**	**Euclidean Distance**
Airplane	0.6525	0.3475	Railway	60.8280

Received: July 20, 2025

Railway	0.6419	0.3581	Runway_marking	61.3320
Airplane	0.6375	0.3625	Runway_marking	61.3687
Runway_marking	0.6360	0.364	Runway_marking	61.4322
Runway_marking	0.6336	0.3664	Railway	61.7268
Runway_marking	0.6334	0.3666	Runway_marking	61.9572
Railway	0.6323	0.3677	Runway_marking	61.9726
Runway	0.6322	0.3678	Runway_marking	61.9730
Runway_marking	0.6279	0.3721	Railway	62.0001
Runway_marking	0.6272	0.3728	Railway	62.0615
Structural Similarity Index Metric with top ranked image (SSIM)				**0.2997**

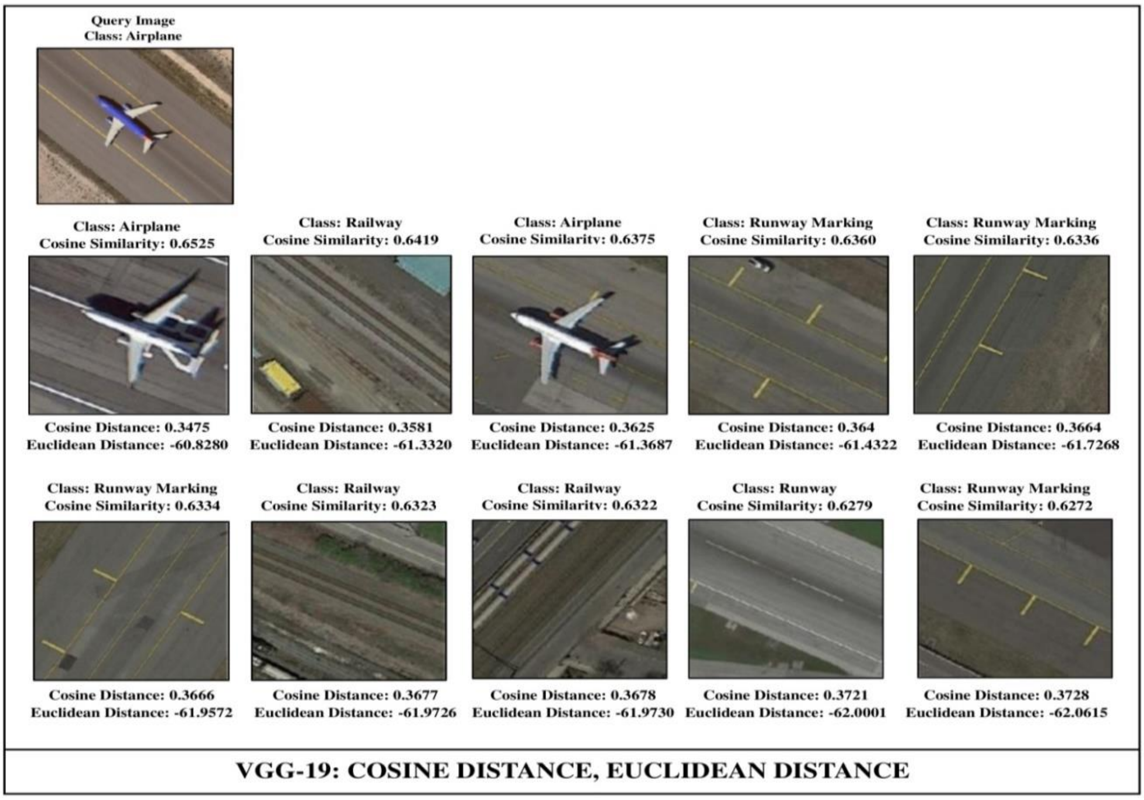
Query Image
Class: Airplane
Class: Airplane
Cosine Similarity: 0.6525
Cosine Distance: 0.3475
Euclidean Distance: -60.8280
Class: Railway
Cosine Similarity: 0.6419
Cosine Distance: 0.3581
Euclidean Distance: -61.3320
Class: Airplane
Cosine Similaritv: 0.6375
Cosine Distance: 0.3625
Euclidean Distance: -61.3687
Class: Runway Marking
Cosine Similarity: 0.6360
Cosine Distance: 0.364
Euclidean Distance: -61.4322
Class: Runway Marking
Cosine Similarity: 0.6336
Cosine Distance: 0.3664
Euclidean Distance: -61.7268
Class: Runway Marking
Cosine Similarity: 0.6334
Cosine Distance: 0.3666
Euclidean Distance: -61.9572
Class: Railway
Cosine Similarity: 0.6323
Cosine Distance: 0.3677
Euclidean Distance: -61.9726
Class: Railway
Cosine Similaritv: 0.6322
Cosine Distance: 0.3678
Euclidean Distance: -61.9730
Class: Runway
Cosine Similarity: 0.6279
Cosine Distance: 0.3721
Euclidean Distance: -62.0001
Class: Runway Marking
Cosine Similarity: 0.6272
Cosine Distance: 0.3728
Euclidean Distance: -62.0615
VGG-19: COSINE DISTANCE, EUCLIDEAN DISTANCE

Figure 7: Retrieving top-10 Remote Sensing Images: VGG-19

Table 6: Experimental Results for Image Retrieval: INCEPTION-v3

INCEPTION-V3 : COSINE DISTANCE , EUCLIDEAN DISTANCE				
Retrieved Image Classes	**Cosine Similarity Score**	**Cosine Distance= 1-Cosine Similarity**	**Retrieved Image Classes**	**Euclidean Distance**
Airplane	0.8366	0.1634	Airplane	11.1610
Airplane	0.8265	0.1735	Airplane	11.3709
Airplane	0.8240	0.176	Airplane	11.6555
Airplane	0.8237	0.176	Airplane	11.7180
Airplane	0.8192	0.1808	Airplane	12.0374
Airplane	0.8053	0.1947	Airplane	12.7593
Airplane	0.7809	0.2191	Airplane	12.7863
Airplane	0.7791	0.2209	Airplane	12.8599
Airplane	0.7766	0.2234	Airplane	12.8609
Airplane	0.7751	0.2249	Airplane	12.9831
Structural Similarity Index Metric with top ranked image (SSIM)				**0.3630**

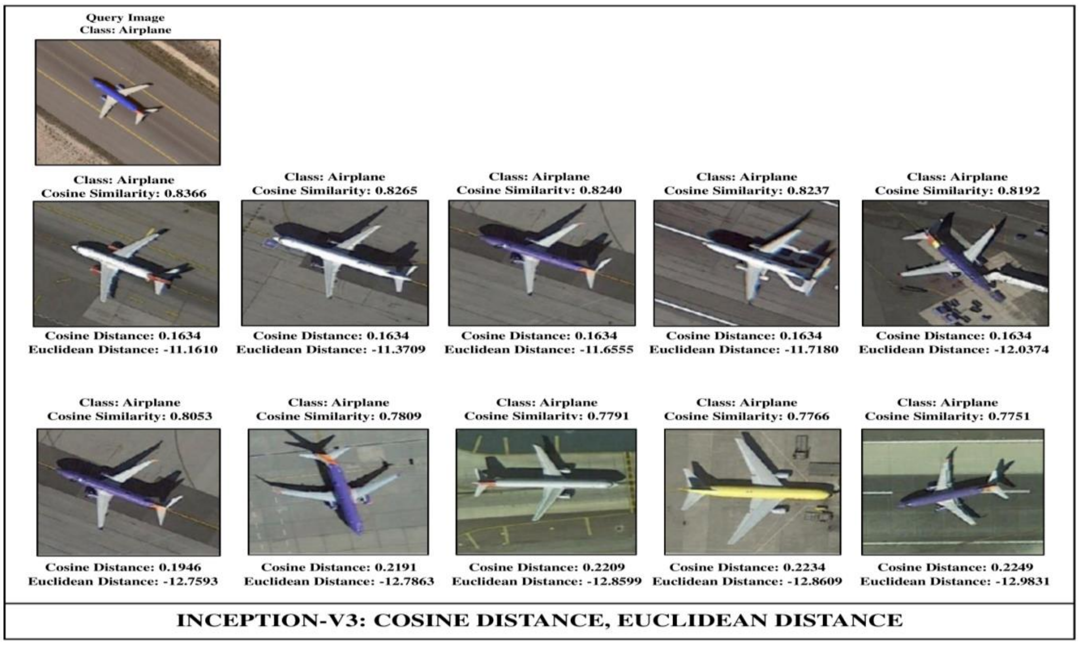
Query Image
Class: Airplane
Class: Airplane
Cosine Similarity: 0.8366
Cosine Distance: 0.1634
Euclidean Distance: -11.1610
Class: Airplane
Cosine Similarity: 0.8265
Cosine Distance: 0.1634
Euclidean Distance: -11.3709
Class: Airplane
Cosine Similaritv: 0.8240
Cosine Distance: 0.1634
Euclidean Distance: -11.6555
Class: Airplane
Cosine Similarity: 0.8237
Cosine Distance: 0.1634
Euclidean Distance: -11.7180
Class: Airplane
Cosine Similarity: 0.8192
Cosine Distance: 0.1634
Euclidean Distance: -12.0374
Class: Airplane
Cosine Similarity: 0.8053
Cosine Distance: 0.1946
Euclidean Distance: -12.7593
Class: Airplane
Cosine Similarity: 0.7809
Cosine Distance: 0.2191
Euclidean Distance: -12.7863
Class: Airplane
Cosine Similaritv: 0.7791
Cosine Distance: 0.2209
Euclidean Distance: -12.8599
Class: Airplane
Cosine Similarity: 0.7766
Cosine Distance: 0.2234
Euclidean Distance: -12.8609
Class: Airplane
Cosine Similarity: 0.7751
Cosine Distance: 0.2249
Euclidean Distance: -12.9831
INCEPTION-V3: COSINE DISTANCE, EUCLIDEAN DISTANCE

Figure 8: Retrieving top 10 Remote Sensing Images: Inception-v3

Table 7: Experimental Results for Image Retrieval: DENSENET-169

DENSENET-169 : COSINE DISTANCE , EUCLIDEAN DISTANCE				
Retrieved Image Classes	**Cosine Similarity Score**	**Cosine Distance= 1-Cosine Similarity**	**Retrieved Image Classes**	**Euclidean Distance**
Airplane	0.7892	0.2108	Airplane	20.8729
Airplane	0.7469	0.2531	Airplane	21.5150
Airplane	0.7185	0.2815	Airplane	22.8922
Airplane	0.7149	0.2851	Airplane	23.3005
Airplane	0.7093	0.2907	Airplane	23.4499
Airplane	0.7016	0.2984	Airplane	23.5187
Airplane	0.6987	0.3013	Airplane	23.5881
Airplane	0.6977	0.3023	Airplane	23.6836
Airplane	0.6953	0.3047	Airplane	23.7170
Airplane	0.6952	0.3048	Airplane	23.8126
Structural Similarity Index Metric with top ranked image (SSIM)				**0.3630**

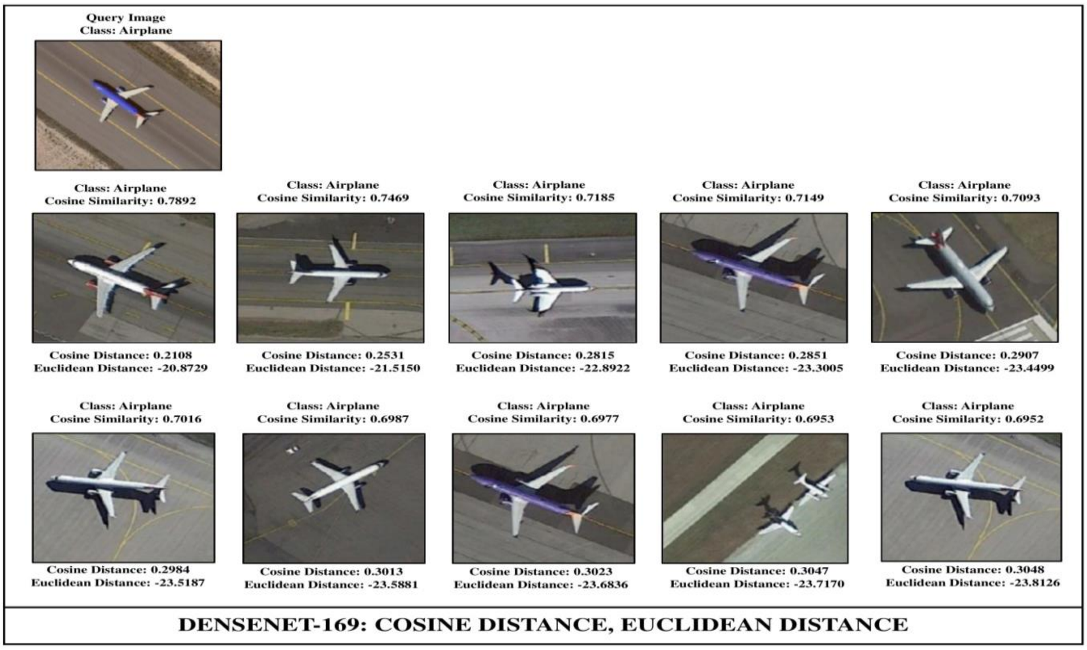
Query Image
Class: Airplane
Class: Airplane
Cosine Similarity: 0.7892
Cosine Distance: 0.2108
Euclidean Distance: -20.8729
Class: Airplane
Cosine Similarity: 0.7469
Cosine Distance: 0.2531
Euclidean Distance: -21.5150
Class: Airplane
Cosine Similarity: 0.7185
Cosine Distance: 0.2815
Euclidean Distance: -22.8922
Class: Airplane
Cosine Similarity: 0.7149
Cosine Distance: 0.2851
Euclidean Distance: -23.3005
Class: Airplane
Cosine Similarity: 0.7093
Cosine Distance: 0.2907
Euclidean Distance: -23.4499
Class: Airplane
Cosine Similarity: 0.7016
Cosine Distance: 0.2984
Euclidean Distance: -23.5187
Class: Airplane
Cosine Similarity: 0.6987
Cosine Distance: 0.3013
Euclidean Distance: -23.5881
Class: Airplane
Cosine Similarity: 0.6977
Cosine Distance: 0.3023
Euclidean Distance: -23.6836
Class: Airplane
Cosine Similarity: 0.6953
Cosine Distance: 0.3047
Euclidean Distance: -23.7170
Class: Airplane
Cosine Similarity: 0.6952
Cosine Distance: 0.3048
Euclidean Distance: -23.8126
DENSENET-169: COSINE DISTANCE, EUCLIDEAN DISTANCE

Figure 9: Retrieving top 10 Remote Sensing Images: DENSENET-169

Received: July 20, 2025

Table 8: Experimental Results for Image Retrieval: DENSENET-121

DENSENET-121: COSINE DISTANCE , EUCLIDEAN DISTANCE				
Retrieved Image Classes	**Cosine Similarity Score**	**Cosine Distance= 1-Cosine similarity**	**Retrieved Image Classes**	**Euclidean Distance**
Airplane	0.5964	0.4036	Airplane	-290.6993
Airplane	0.5760	0.424	Airplane	- 292.8274
Airplane	0.5742	0.4258	Airplane	- 293.7777
Airplane	0.5702	0.4298	Airplane	- 293.7981
Airplane	0.5702	0.4298	Airplane	- 294.3795
Airplane	0.5664	0.4336	Airplane	- 295.0906
Airplane	0.5630	0.437	Airplane	- 295.5878
Airplane	0.5616	0.4384	Airplane	- 296.4362
Airplane	0.5612	0.4388	Airplane	- 297.9952
Airplane	0.5605	0.4395	Airplane	- 298.4723
Structural Similarity Index Metric with top ranked image (SSIM)				**0.3630**

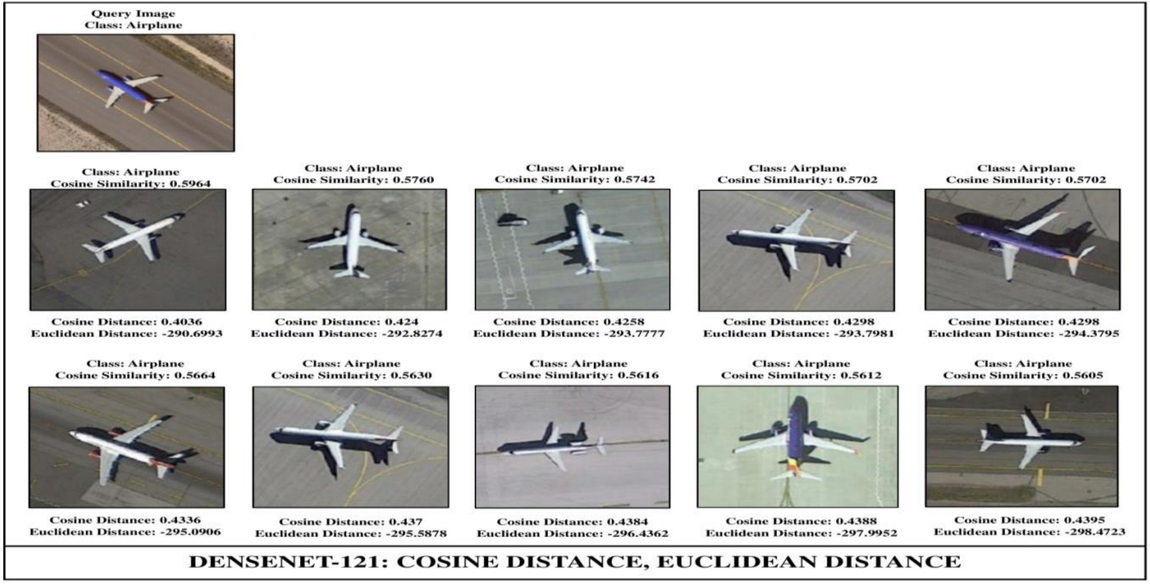
Query Image
Class: Airplane
Class: Airplane
Cosine Similarity: 0.5964
Cosine Distance: 0.4036
Euclidean Distance: -290.6993
Class: Airplane
Cosine Similarity: 0.5760
Cosine Distance: 0.424
Euclidean Distance: -292.8274
Class: Airplane
Cosine Similarity: 0.5742
Cosine Distance: 0.4258
Euclidean Distance: -293.7777
Class: Airplane
Cosine Similarity: 0.5702
Cosine Distance: 0.4298
Euclidean Distance: -293.7981
Class: Airplane
Cosine Similarity: 0.5702
Cosine Distance: 0.4298
Euclidean Distance: -294.3795
Class: Airplane
Cosine Similarity: 0.5664
Cosine Distance: 0.4336
Euclidean Distance: -295.0906
Class: Airplane
Cosine Similarity: 0.5630
Cosine Distance: 0.437
Euclidean Distance: -295.5878
Class: Airplane
Cosine Similarity: 0.5616
Cosine Distance: 0.4384
Euclidean Distance: -296.4362
Class: Airplane
Cosine Similarity: 0.5612
Cosine Distance: 0.4388
Euclidean Distance: -297.9952
Class: Airplane
Cosine Similarity: 0.5605
Cosine Distance: 0.4395
Euclidean Distance: -298.4723
DENSENET-121: COSINE DISTANCE, EUCLIDEAN DISTANCE

Figure 10: Retrieving top 10 Remote Sensing Images: DENSENET-121

Table 9: Experimental Results for Image Retrieval: DENSENET-201

DENSENET-201: COSINE DISTANCE , EUCLIDEAN DISTANCE				
Retrieved Image Classes	**Cosine Similarity Score**	**Cosine Distance= 1-Cosine similarity**	**Retrieved Image Classes**	**Euclidean Distance**

Airplane	0.7504	0.2496	Airplane	21.4609
Airplane	0.7263	0.2737	Airplane	22.7139
Airplane	0.7252	0.2748	Airplane	23.5719
Airplane	0.7057	0.2943	Airplane	23.6351
Airplane	0.7025	0.2975	Airplane	23.6660
Airplane	0.6954	0.3046	Airplane	23.7794
Airplane	0.6939	0.3061	Airplane	23.7797
Airplane	0.6908	0.3092	Airplane	24.1893
Airplane	0.6845	0.3155	Airplane	24.1949
Airplane	0.6840	0.316	Airplane	24.2173
Structural Similarity Index Metric (SSIM) with top ranked image				**0.3630**

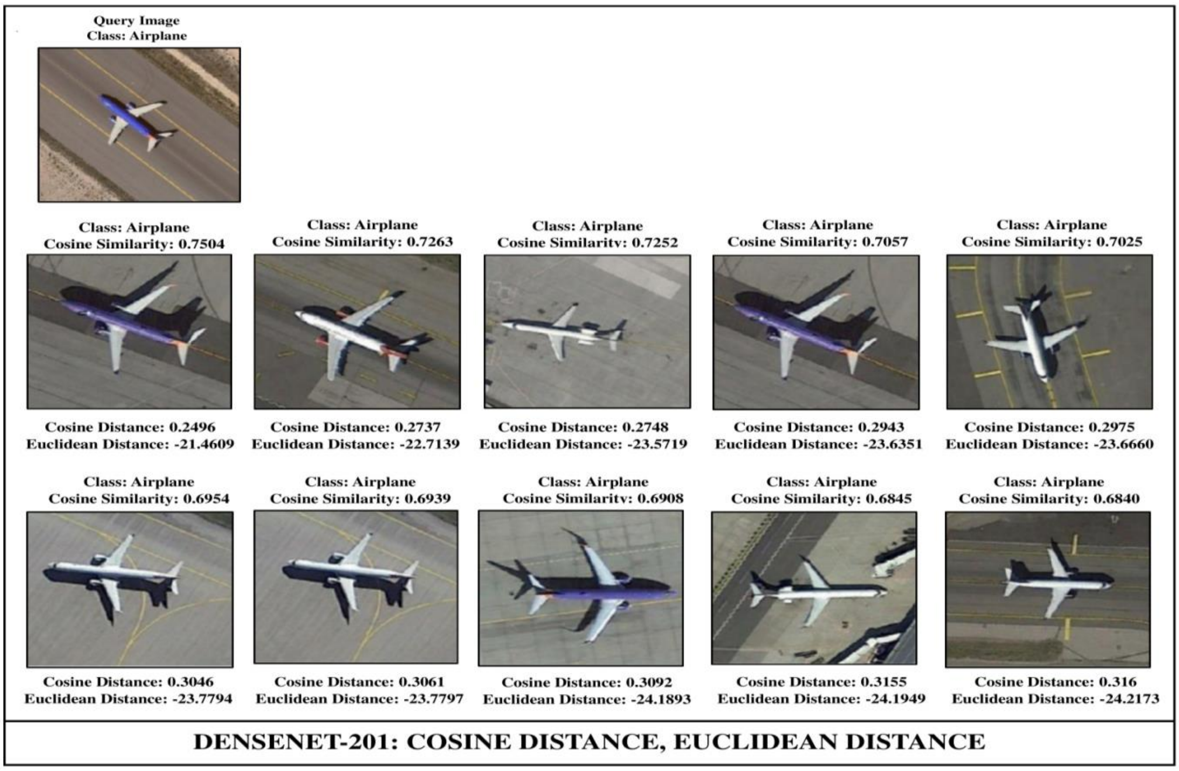
Query Image
Class: Airplane
Class: Airplane
Cosine Similarity: 0.7504
Cosine Distance: 0.2496
Euclidean Distance: -21.4609
Class: Airplane
Cosine Similarity: 0.7263
Cosine Distance: 0.2737
Euclidean Distance: -22.7139
Class: Airplane
Cosine Similarity: 0.7252
Cosine Distance: 0.2748
Euclidean Distance: -23.5719
Class: Airplane
Cosine Similarity: 0.7057
Cosine Distance: 0.2943
Euclidean Distance: -23.6351
Class: Airplane
Cosine Similarity: 0.7025
Cosine Distance: 0.2975
Euclidean Distance: -23.6660
Class: Airplane
Cosine Similarity: 0.6954
Cosine Distance: 0.3046
Euclidean Distance: -23.7794
Class: Airplane
Cosine Similarity: 0.6939
Cosine Distance: 0.3061
Euclidean Distance: -23.7797
Class: Airplane
Cosine Similarity: 0.6908
Cosine Distance: 0.3092
Euclidean Distance: -24.1893
Class: Airplane
Cosine Similarity: 0.6845
Cosine Distance: 0.3155
Euclidean Distance: -24.1949
Class: Airplane
Cosine Similarity: 0.6840
Cosine Distance: 0.316
Euclidean Distance: -24.2173
DENSENET-201: COSINE DISTANCE, EUCLIDEAN DISTANCE

Figure 11: Retrieving top 10 Remote Sensing Images: DenseNet-201

14. Conclusion

In this paper, deep learning pre-trained models VGG-19, Inception- v3, and DenseNet including its variants DenseNet-169, DenseNet-121 and DenseNet-201 are used for remote sensing image retrieval. Experiments are conducted on remote sensing dataset UC-Merced, AID, NWPU-RESIS45 and PatterNet. This paper goes through two steps image classification and image retrieval. According to results, DenseNet-201 performs most effectively for image classification on PatterNet dataset. Further for image retrieval cosine similarity is used to calculate the similarity index. The euclidean distance indicates how close the top-k images are to the query image. The Structured Similarity Index (SSI) of the image with the highest ranking is determined. Finally proposed image retrieval system looks for related images (top-k, k=10) by searching and sorting very large remote sensing dataset PatterNet. The image retrieval system with the highest structural similarity index value for the highest-ranking images is DenseNet-201 on PatterNet dataset.

The work demonstrated in this study can be extended in many ways. For improved feature extraction and retrieval of remote sensing images it might be important to test several pre-trained models in the future in order to improve the existing systems's performance.

References

1. Agarwal, Megha, Amit Singhal, and Brejesh Lall (2019). "Multi-channel local ternary pattern for content-based image retrieval". In: Pattern Analysis and Applications 22, pp. 1585–1596.

2. Alem, Abebaw and Shailender Kumar (2022). "Deep Learning Models Performance Evaluations for Remote Sensed Image Classification". In: IEEE Access 10, pp. 111784–111793.

3. Alkhawlani, Mohammed, Mohammed Elmogy, and Hazem El Bakry (2015). "Text-based, content-based, and semantic-based image retrievals: a survey". In: Int. J. Comput. Inf. Technol 4.01, pp. 58–66.

4. Alsmadi, Mutasem K (2020). "Content-based image retrieval using color, shape and texture descriptors and features". In: Arabian Journal for Science and Engineering 45.4, pp. 3317–3330.
5. Alzu'bi, Ahmad, Abbes Amira, and Naeem Ramzan (2017). "Content-based image retrieval with compact deep convolutional features". In: Neurocomputing 249, pp. 95–105.

6. Al-Anzi, Fawaz S and Dia AbuZeina (2017). "Toward an enhanced Arabic text classification using cosine similarity and Latent Semantic Indexing". In: Journal of King Saud University-Computer and Information Sciences 29.2, pp. 189–195.

7. Ashraf, Rehan et al. (2020). "MDCBIR-MF: multimedia data for content-based image retrieval by using multiple features". In: Multimedia tools and applications 79.13, pp. 8553–8579.

8. Bagaskara, Abitya and Muhammad Suryanegara (2021). "Evaluation of VGG-16 and VGG-19 deep learning architecture for classifying dementia people". In: 2021 4th International Conference of Computer and Informatics Engineering (IC2IE).IEEE, pp. 1–4.

9. Baig, Fahad et al. (2020). "Boosting the performance of the BoVW model using SURF–CoHOG-based sparse features with relevance feedback for CBIR". In: Iranian Journal of Science and Technology, Transactions of Electrical Engineering 44.1, pp. 99–118.

10. Bakurov, Illya et al. (2022). "Structural similarity index (SSIM) revisited: A data-driven approach". In: Expert Systems with Applications 189, p. 116087.Bella, Mary I Thusnavis and A Vasuki (2019). "An efficient image retrieval framework using fused information feature". In:Computers & Electrical Engineering 75, pp. 46–60.

11. Brunet, Dominique, Edward R Vrscay, and Zhou Wang (2011). "On the mathematical properties of the structural similarity index". In: IEEE Transactions on Image Processing 21.4, pp. 1488–1499.

Received: July 20, 2025

12. Chaudhuri, Bindita, Beg¨um Demir, Lorenzo Bruzzone, et al. (2016). "Region-based retrieval of remote sensing images using an unsupervised graph-theoretic approach". In: IEEE Geoscience and Remote Sensing Letters 13.7, pp. 987–991.

13. Chaudhuri, Bindita, Beg¨um Demir, Subhasis Chaudhuri, et al. (2017). "Multilabel remote sensing image retrieval using a semisupervised graph-theoretic method". In: IEEE Transactions on Geoscience and Remote Sensing 56.2, pp. 1144–1158.

14. Cheng, Gong, Junwei Han, and Xiaoqiang Lu (2017). "Remote sensing image scene classification: Benchmark and state of the art". In: Proceedings of the IEEE 105.10, pp. 1865–1883.

15. Dalal, Navneet and Bill Triggs (2005). "Histograms of oriented gradients for human detection". In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). Vol. 1. Ieee, pp. 886–893.

16. Desai, Chitra (2021). "Image classification using transfer learning and deep learning". In: International Journal of Engineering and Computer Science 10.9, pp. 25394–25398.

17. Do, Thanh-Toan and Ngai-Man Cheung (2017). "Embedding based on function approximation for large scale image search". In: IEEE transactions on pattern analysis and machine intelligence 40.3, pp. 626–638.

18. Do, Thanh-Toan, Khoa Le, et al. (2019). "Simultaneous feature aggregating and hashing for compact binary code learning". In: IEEE Transactions on Image Processing 28.10, pp. 4954–4969.

19. Gower, John Clifford (1985). "Properties of Euclidean and non-Euclidean distance matrices". In: Linear algebra and its applications 67, pp. 81–97.29

20. He, Kaiming et al. (2016). "Deep residual learning for image recognition". In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

21. Huang, Gao et al. (2017). "Densely connected convolutional networks". In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4700–4708.

22. Husain, Syed Sameed and Miroslaw Bober (2016). "Improving large-scale image retrieval through robust aggregation of local descriptors". In: IEEE transactions on pattern analysis and machine intelligence 39.9, pp. 1783–1796.

23. Imbriaco, Raffaele et al. (2021). "Toward multilabel image retrieval for remote sensing". In: IEEE Transactions on Geoscience and Remote Sensing 60, pp. 1–14.

24. Jabeen, Safia et al. (2018). "An effective content-based image retrieval technique for image visuals representation based on the bag-of-visual-words model". In: PloS one 13.4, e0194526.

25. Kokare, Manesh, BN Chatterji, and PK Biswas (2002). "A survey on current content based image retrieval methods". In: IETE Journal of Research 48.3-4, pp. 261–271.

26. Krishna, Sajja Tulasi and Hemantha Kumar Kalluri (2019). “Deep learning and transfer learning approaches for image classification”. In: International Journal of Recent Technology and Engineering (IJRTE) 7.5S4, pp. 427–432.

27. Krishnapriya, Srigiri and Yepuganti Karuna (2023). “Pre-trained deep learning models for brain MRI image classification”. In: Frontiers in Human Neuroscience 17, p. 1150120.

28. Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2012). “Imagenet classification with deep convolutional neural networks”. In: Advances in neural information processing systems 25.

29. Li, Guoqing et al. (2021). “Efficient densely connected convolutional neural networks”. In: Pattern Recognition 109, p. 107610.

30. Li, Yansheng, Jiayi Ma, and Yongjun Zhang (2021). “Image retrieval from remote sensing big data: A survey”. In: Information Fusion 67, pp. 94–115.

31. Liu, Chao et al. (2020). “Deep hash learning for remote sensing image retrieval”. In: IEEE Transactions on Geoscience and Remote Sensing 59.4, pp. 3420–3443.

32. Marmanis, Dimitrios et al. (2015). “Deep learning earth observation classification using ImageNet pretrained networks”. In: IEEE Geoscience and Remote Sensing Letters 13.1, pp. 105–109.

33. Muhammad, Usman et al. (2018). “Pre-trained VGGNet architecture for remote-sensing image scene classification”. In: 2018 24th International Conference on Pattern Recognition (ICPR). IEEE, pp. 1622–1627.

34. Ojala, Timo, Matti Pietikainen, and Topi Maenpaa (2002). “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns”. In: IEEE Transactions on pattern analysis and machine intelligence 24.7, pp. 971–987.

35. ¨Ozkan, Sava¸s et al. (2014). “Performance analysis of state-of-the-art representation methods for geographical image retrieval and categorization”. In: IEEE Geoscience and Remote Sensing Letters 11.11, pp. 1996–2000.

36. Park, Kwangil, June Seok Hong, and Wooju Kim (2020). “A methodology combining cosine similarity with classifier for text classification”. In: Applied Artificial Intelligence 34.5, pp. 396–411.

37. Pathak, Debanjan and USN Raju (2021). “Content-based image retrieval using feature-fusion of GroupNormalized-Inception-Darknet-53 features and handcraft features”. In: Optik 246, p. 167754.

38. Pavithra, LK and T Sree Sharmila (2019). “An efficient seed points selection approach in dominant color descriptors (DCD)”.In: Cluster Computing 22.4, pp. 1225–1240.

39. Ponomarev, Alexey et al. (2016). “Content-based image retrieval using color, texture and shape features”. In: Key Engineering Materials 685, pp. 872–876.

40. Puls, Erick da Silva, Matheus V Todescato, and Joel L Carbonera (2023). “An evaluation of pre-trained models for feature extraction in image classification”. In: arXiv preprint arXiv:2310.02037.

41. Rana, Soumya Prakash, Maitreyee Dey, and Patrick Siarry (2019). "Boosting content based image retrieval performance through integration of parametric & nonparametric approaches". In: Journal of Visual Communication and Image Representation 58, pp. 205–219.

42. Risojevi´c, Vladimir and Vladan Stojni´c (2021). "Do we still need ImageNet pre-training in remote sensing scene classification?"In: arXiv preprint arXiv:2111.03690.

43. Sarwar, Amna et al. (2019). "A novel method for content-based image retrieval to improve the effectiveness of the bag-of-words model using a support vector machine". In: Journal of Information Science 45.1, pp. 117–135.

44. Sezavar, Amir, Hassan Farsi, and Sajad Mohamadzadeh (2019). "Content-based image retrieval by combining convolutional neural networks and sparse representation". In: Multimedia Tools and Applications 78, pp. 20895–20912.

45. Sharif, Uzma et al. (2019). "Scene analysis and search using local features and support vecto machine for effective content-based image retrieval". In: Artificial Intelligence Review 52, pp. 901–925.

46. Shrivastava, Nishant and Vipin Tyagi (2015). "An efficient technique for retrieval of color images in large databases". In:Computers & Electrical Engineering 46, pp. 314–327.
47. Smeulders, Arnold WM et al. (2000). "Content-based image retrieval at the end of the early years". In: IEEE Transactions on pattern analysis and machine intelligence 22.12, pp. 1349–1380.

48. Sumbul, Gencer and Beg¨um Demir (2022). "Plasticity-stability preserving multi-task learning for remote sensing image retrieval". In: IEEE Transactions on Geoscience and Remote Sensing 60, pp. 1–16.

49. Tadi Bani, Neda and Shervan Fekri-Ershad (2019). "Content-based image retrieval based on combination of texture and colour information extracted in spatial and frequency domains". In: The electronic library 37.4, pp. 650–666.

50. Thirumaladevi, S, K Veera Swamy, and M Sailaja (2023). "Remote sensing image scene classification by transfer learning to augment the accuracy". In: Measurement: Sensors 25, p. 100645.30

51. Tzelepi, Maria and Anastasios Tefas (2018). "Deep convolutional learning for content based image retrieval". In: Neurocomputing275, pp. 2467–2478.

52. Wang, Cheng et al. (2019). "Pulmonary image classification based on inception-v3 transfer learning model". In: IEEE Access7, pp. 146533–146541.

53. Wang, Jie et al. (2020). "Object-scale adaptive convolutional neural networks for high-spatial resolution remote sensing image classification". In: IEEE Journal of Selected Topics in Applied
54. Earth Observations and Remote Sensing 14, pp. 283–299.

55. Wang, Liwei, Yan Zhang, and Jufu Feng (2005). "On the Euclidean distance of images". In: IEEE transactions on pattern analysis and machine intelligence 27.8, pp. 1334–1339.

56. Wang, Yuze et al. (2021). “Cross-sensor remote-sensing images scene understanding based on transfer learning between heterogeneous networks”. In: IEEE Geoscience and Remote Sensing Letters 19, pp. 1–5.

57. Xia, Gui-Song et al. (2017). “AID: A benchmark data set for performance evaluation of aerial scene classification”. In: IEEE Transactions on Geoscience and Remote Sensing 55.7, pp. 3965–3981.

58. Xia, Peipei, Li Zhang, and Fanzhang Li (2015). “Learning similarity with cosine similarity ensemble”. In: Information sciences
59. 307, pp. 39–52.

60. Yang, Yi and Shawn Newsam (2010). “Bag-of-visual-words and spatial extensions for land-use classification”. In: Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems, pp. 270–279.

61. Zhang, Jianming et al. (2019). “A full convolutional network based on DenseNet for remote sensing scene classification”. In: Mathematical Biosciences and Engineering 16.5, pp. 3345–3367.

62. Zhang, Zhiqi et al. (2022). “A discriminative feature learning approach with distinguishable distance metrics for remote sensing image classification and retrieval”. In: IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 16, pp. 889–901.

63. Zhen, Liangli et al. (2020). “Deep multimodal transfer learning for cross-modal retrieval”. In: IEEE Transactions on Neural Networks and Learning Systems 33.2, pp. 798–810.
64. Zheng, Qinghe et al. (2019). “Differential Learning: A Powerful Tool for Interactive Content-Based Image Retrieval.” In:Engineering Letters 27.1.

65. Zhou, Shuren et al. (2018). “Improved VGG model for road traffic sign recognition”. In: Computers, Materials & Continua 57.1, pp. 11–24.

66. Zhou, Weixun et al. (2017). “Learning low dimensional convolutional neural networks for high-resolution remote sensing image retrieval”. In: Remote Sensing 9.5, p. 489.– (2018). “PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval”. In: ISPRS journal of photogrammetry and remote sensing 145, pp. 197–209.

67. Zhuo, Li, Bo Cheng, and Jing Zhang (2014). “A comparative study of dimensionality reduction methods for large-scale image retrieval”. In: Neurocomputing 141, pp. 202–210.