IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 16, 2021, accepted February 28, 2021, date of publication March 2, 2021, date of current version March 22, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3063545

Maximum Response Deep Learning Using Markov, Retinal & Primitive Patch Binding With GoogLeNet & VGG-19 for Large Image Retrieval

KHAWAJA TEHSEEN AHMED[1], SAROOSH JAFFAR[2], MALIK GHULAM HUSSAIN[2], SHAHID FAREED[1], ARIF MEHMOOD[3], AND GYU SANG CHOI[4]

[1]Department of Information Technology, Bahauddin Zakariya University, Multan 60800, Pakistan
[2]Department of Computer Science, Bahauddin Zakariya University, Multan 60800, Pakistan
[3]Department of CS and IT, The Islamia University of Bahawalpur, Bahawalpur 63100, Pakistan
[4]Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, South Korea

Corresponding author: Gyu Sang Choi (castchoi@ynu.ac.kr)

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2019R1A2C1006159) and MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2016-0-00313) supervised by the IITP (Institute for Information & communications Technology Promotion).

ABSTRACT Smart and productive image retrieval from flexible image datasets is an unavoidable necessity of the current period. Crude picture marks are imperative to mirror the visual ascribes for content-based image retrieval (CBIR). Algorithmically enlightened and recognized visual substance structure image marks to accurately file and recover comparative outcomes. Consequently, highlighted vectors ought to contain adequate image data with color, shape, objects, spatial data viewpoints to recognize image class as a qualifying applicant. This article presents the maximum response of visual features of an image over profound convolutional neural networks in blend with an innovative content-based image retrieval plan to recover phenomenally precise outcomes. For this determination, a serial fusion of GoogLeNet and VGG-19 based generated signatures are formulated with visual features including texture, color and shape. Initially, the maximum response is calculated for texture pattern by using Markov Random Field (MRF) classifier. Thereafter, cascaded samples are passed through a human retinal system like descriptor named Fast Retina Keypoint (FREAK) for corresponding fundamental points through the image. GoogLeNet and VGG-19 are applied to extract deep features of an image; hence color components are obtained using a correlogram. Finally, all the image signatures are combined and passed through the BoW scheme. The proposed method is applied experimentally to challenging datasets, including Caltech-256, ALOT (250), Corel 1000, Cifar-100, and Cifar 10. Remarkable precision,Recall and F-score results obtained.The texture dataset ALOT (250) with the uppermost precision rate 0.99 for a maximum of its categories, whereas Caltech-256 gives 0.66 precision, and Corel 1000 0.99 for VGG-19 and 0.95 for GoogLeNet. Recall, F-score, ARR and ARP rates shows the significant rates in most of the image categories.

INDEX TERMS Bag of words, cascade sampling, content based image retrieval, color components, maximum response for texture pattern, combination of features.

I. INTRODUCTION

CBIR systems extract the features from the images stored in database and the query image and when the user inputs a query image the CBIR system matches their features such as texture, color, edges, etc. and return the relevant images to the user. Prior to feature extraction is feature detection. Feature detection means to choose the points of interest. Feature detection is the primary step for any image retrieval system and its effectiveness

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

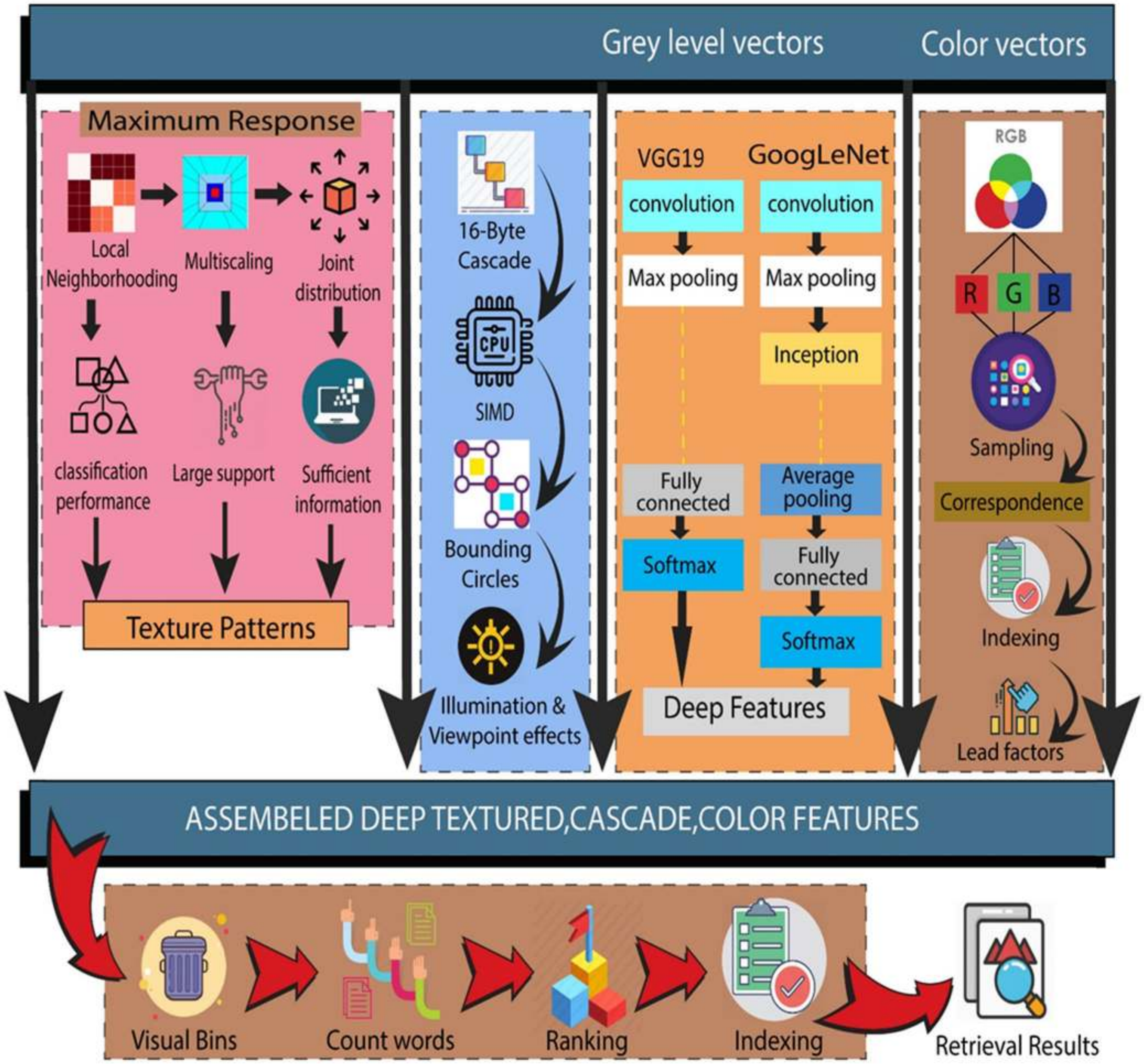

FIGURE 1. The proposed method demonstrating step by step process of image retrieval.

truly depends on the methodology chosen to extract features.

Sometimes learning or representing the features takes more time and computational cost is relatively high. Extraction of the visual features of images, e.g. shape, color, and texture, helps the CBIR systems to automatically annotate images and only this feature extraction strategy makes them better than the TBIR systems. Thus, it is vital to learn better feature extraction techniques and fast similarity measures to achieve better performances [1]. Feature descriptors are used to differentiate one image feature from another by encoding information of interest into a series of numbers. They are also a mean of comparing features of different images and feature extraction can only be performed after applying descriptor algorithm on them. Features play fundamental role in the retrieval of precise images. Image features are classified into two classes as global features and local features. Global features cover or focus on the overall image attributes [3]. They represent the semantic similarity at the abstract level between the images. They do not play any role in overcoming the semantic gap. For example color, shape, texture etc. are some global features. Local features work on the very deep level as

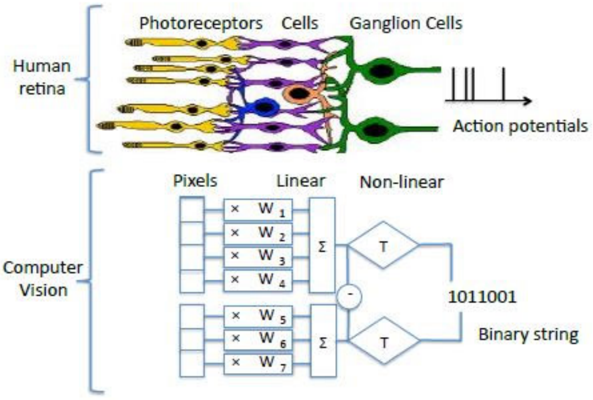

FIGURE 2. Human retina pattern followed by computer vision [1].

they are capable of extracting deep image characteristics such as edges, corners etc. During extraction process attributes of each pixel are computed considering its neighborhoods. Local features are helpful in overcoming the semantic gap. However, both global features and local features must be combined to get the most useful and maximum image features. Color features can be extracted using color-histograms and color-coherence vector etc. The information related to color distribution in image is provided by color histogram [4]. Color histogram descriptors do not take into account the spatial location and only focus on the position of all the colors. The color coherence vector (CCV) measures the spatial coherence of image pixels and thus incorporates spatial information. Red color will have high coherence if the red pixels of the image are members of large red regions but have low coherence if the red pixels are scattered.

In this research work, the CBIR technique is employed to calculate the maximum response for texture with color coordinates and cascade samples for GoogLeNet and VGG19 for efficient image retrieval. Strong experimental conventions using public datasets Corel 1000, ALOT (250), 256_Object Categories have demonstrated that the proposed techniques analyze well against best in class archive image retrieval, color, object detection, and texture identifying approaches.

II. RELATED WORK

TABLE 1. Value of precision represented in tabular form for categories falling in bird type in Caltech-250 dataset.

No.	Category	Precision	Recall	Fscore
1	bat	0.1	1	0.18
2	cormorant	0.4	0.25	0.31
3	duck	0.2	0.5	0.29
4	goose	0.25	0.4	0.31
5	hummingbird	0.35	0.29	0.31
6	ibis-101	0.55	0.18	0.27
7	ostrich	0.35	0.29	0.31
8	owl	0.15	0.67	0.24
9	penguin	0.25	0.4	0.31
10	swan	0.35	0.29	0.31

TABLE 2. Value of precision represented in tabular form for categories falling in air vehicles type in Caltech-250 dataset.

No.	Category	Precision
1	airplanes-101	0.9
2	blimp	0.1
3	fighter-jet	0.1
4	helicopter	0.2
5	hot-air-balloon	0.15

TABLE 3. Value of precision represented in tabular form for categories falling in animal type in Caltech-250 dataset.

No.	Category	Precision	No.	Category	Precision
1.	bear	0.45	2.	horse	0.45
3.	chimp	0.3	4.	kangroo-101	0.5
5.	dog	0.1	6.	leopards-101	0.7
7.	el	0.15	8.	llama-101	0.3
9.	elephant-101	0.5	10.	porcupine	0.2
11.	elk	0.6	12.	raccoon	1
13.	giraffe	0.25	14.	skunk	0.1
15.	goat	0.2	16.	teddy-bear	0.1
17.	gorilla	0.65	18.	triceratops	0.1
19.	greyhound	0.55			

A growing number of digital images are seen in the current era that creates more difficulty in retrieving the query image from an extensive image database. These circumstances introduced a highly effective technique for the retrieval of images. The well-known and accurate CBIR method uses the internal attributes of images to perform the task efficiently [5]. In the Content-based image retrieval system, there are two main performance indicators for image retrieval: the visual feature presentation and the similarity measures of an image. Another technique [6] presents an involuntary model Deep convolutional neural network (DCNN) using deep learning and computer vision to solve the problem. A Bi-layer Content-Based Image Retrieval (BiCBIR) system concept was given [7], which comprises two modules. Module number one works on the principle of feature extraction of shared dataset images based on the color, shape, and texture. Module number two owns two layers. At first, at layer one, comparison of all images occurs with the query image based on shape, spatial feature, texture, and directories of L most resembled images to query image are retrieved. Then, at layer two, L images recovered from the first layer and compared from the requested image for feature space, color, shape, and output is returned as F images resembled query image.

FIGURE 3. Solitary sample image is selected from respective Caltech 256 categories falling in bird type.

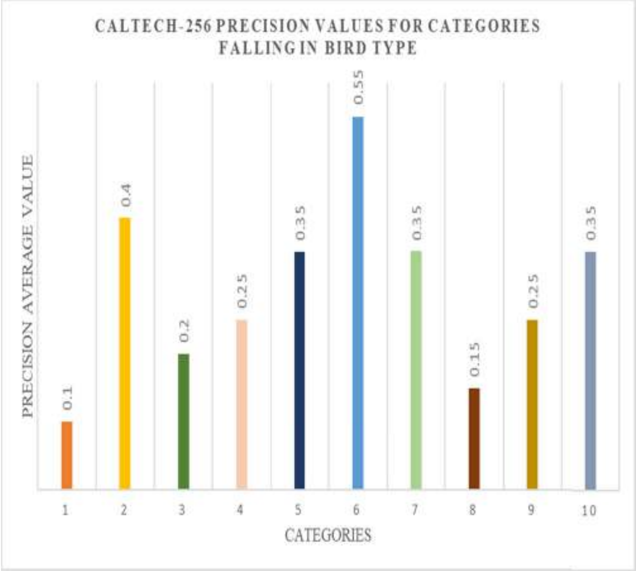

FIGURE 4. Precision rate for categories falling in air vehicles type available in Caltech 256 dataset.

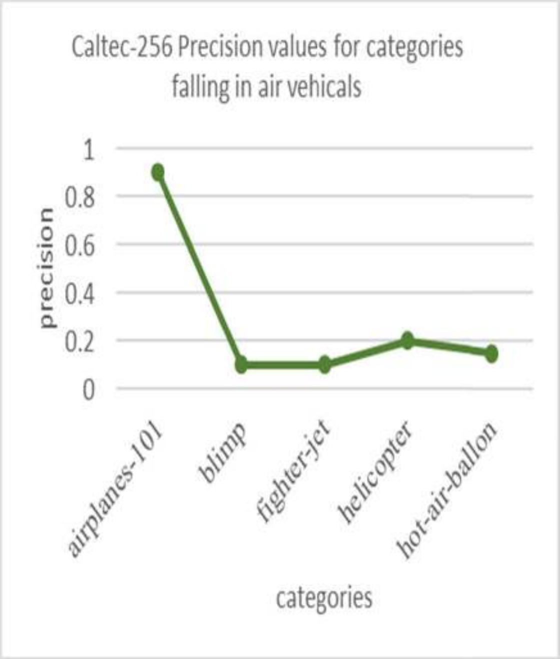

FIGURE 5. Precision rate for categories falling in air vehicles type available in Caltech 256 dataset.

The texture, shape, and color are considered low-level image attributes. Feature merging is applied in CBIR to the proliferation of the performance. The fusion of texture and color features had proposed [8] to excerpt local features as their feature vector. Furthermore, the proposed research has three main stages: feature extraction, equal for similarities, and the 3rd is an appraisal of performance. For the extraction of color features, Color Moments (CM) had used, and Gabor Wavelet and Discrete Wavelet transform to extract the texture features. Additionally, a descriptor to boost the influence of feature vector presentation, known as Color and Edge Directivity Descriptor (CEDD), was a part of the feature vector. These are selected as combinedly, as these features are described as instinctive, dense, and powerful for image representation. In another endeavor [12], a one of a kind crossover capable and beneficial transformative recovery strategy (CBIR-GAF) depended upon the blend of 4 global descriptors had recommended. Every descriptor yields a rundown of recovered related images to the client question image, and if these rundowns are consolidated properly by late combination, the outcomes would be greatly improved and solid. To achieve further comprehend feature description, feature maps of fixed size are computed by combining local variation degree (LVD) of intensity and Gabor features [9]. That includes fusion improved the importance between sub-locales and the legitimacy in the inconsistency of complex surfaces. On these element maps, nearby histograms are made sense of over fixed-size windows to characterize the neighborhood structures formed by include esteems. Furthermore, energy functional is done via Non-Negative Matrix Factorization (NMF), which works to encourage every pixel to drip inside the sub-region that had the most boundless examination zone in its area.

CNN-focused methodologies remove image credits at the final layer, utilizing a solitary CNN design generally through fewer quantization techniques, which limits the utilization of center convolutional layers for recognizing image

FIGURE 6. Sample image is selected from respective Caltech 256 categories falling in animal type.

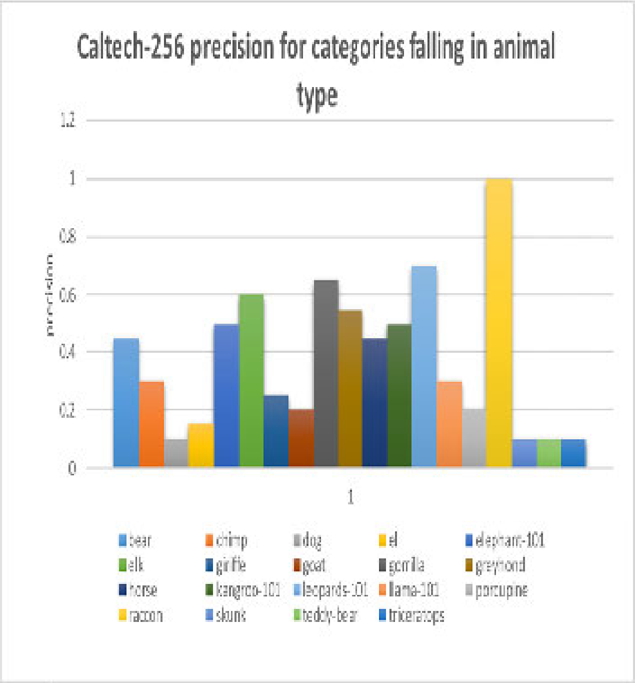

FIGURE 7. Precision rate for categories falling in animal type available in Caltech 256 dataset.

neighborhood arrangements. The proposed network architecture was primed by deep CNNs adequately pre-trained on a big basic image dataset then modified for the CBIR task. Furthermore, to lessen image features, a proficient bilinear root pooling is suggested and functional on the low-dimensional pooling layer to dense but great discriminative image descriptors. Moreover [10], a technique for image interpretation was suggested, which was established by Machine Learning, and the deep learning algorithms model had generated using a mixture of an upgraded AlexNet Convolutional Neural Network (CNN), Histogram of Oriented Gradients (HOG), and Local Binary Pattern (LBP) descriptors. Additionally, the Principle Component Analysis (PCA) algorithm had been utilized for dimension lessening. Range equal to millions of samples is there in Natural image datasets, therefore, responsive to deep-learning techniques. Transfer learning was a technique proposed in [11] numerous arenas of science, remote sensing involved, are able to utilize the achievement of natural image organization by convolutional neural network models via that method. A spatial division network, to identify bounding boxes only with fragile observation was applied. The proposed model [12] consists of two state-of-the-art differentiable modules that perform the spatial division, named determination network and parameterized partition, in feature guides of grouping networks. The opted factors of the spatial division obtained after training would relate to a fixed size of forecast bounding box coordinates. Another attempt [13] to improve image retrieval accuracy image hashing based on deep convolutional neural networks, deep hashing, had stepped forward. Mostly the prevailing deep hashing approaches extracted the feature vector-only commencing the output of the second last entirely-associated layer, aiming predominantly on semantic information while overlooking comprehensive structure statistics. To seal this hole, a unique image hashing process, MLSH (Multi-Level Supervised Hashing) was introduced. A multiple-hash-table method is used to assimilate multilevel elements take out from a distinct deep CNN. As a substitute for easy series, numerous hash tables are trained separately using diverse stages of features from diverse layers that are then combined for effectual image retrieval. Using CNN based VGGNET architecture for the classification of the desired images a Diabetic retinopathy classification system achieved a correspondingly improved solution from side to side the fusion of a Gaussian mixture model (GMM), visual geometry group network (VGGNet), singular value decomposition (SVD), principal component analysis (PCA), and softmax, for region segmentation, high dimensional feature extraction, feature selection and fundus image organization, respectively. The trials are performed using a conventional KAGGLE dataset comprising more than 30,000 images.

FIGURE 8. Caltech-256 dataset showing different sample images of categories falling in computer hardware type.

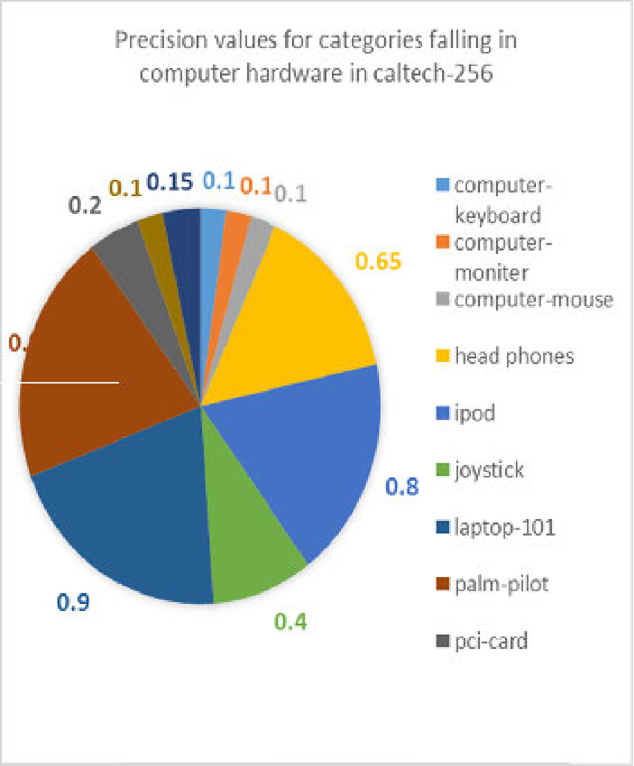

FIGURE 9. The precision rate for categories falling in computer hardware type available in Caltech 256 dataset.

The proposed VGG-19 DNN based DR model [14] outclassed the AlexNet and spatial invariant feature transform (SIFT) in terms of sorting precision and computational time.

TABLE 4. Value of precision represented in tabular form for categories falling in computer hardware type in Calch-256 dataset.

No.	Category name	Precision
1	computer-keyboard	0.1
2	computer-monitor	0.1
3	computer-mouse	0.1
4	headphones	0.65
5	iPod	0.8
6	joystick	0.4
7	laptop-101	0.9
8	palm-pilot	0.9
9	pci-card	0.2
10	vcr	0.1
11	video-projector	0.15

In [15] an End-to-End BoWs (E^2BoWs) model is presented which mainly based on Deep Convolutional Neural Network (DCNN). The mentioned model uses an image as input, at that point distinguishes and parts semantic objects of an image, and in conclusion, restores the visual words with incredible semantic discriminative force as yield. Precisely, that model initially produces Semantic Feature Maps (SFMs) compatible with altered object classes via convolutional layers, then presented Bag-of-Words Layers (BoWL) to create visual words from each specific feature map. The model had evaluated on numerous image exploration datasets, together with MNIST, CIFAR-10, CIFAR-100, SVHN, NUS-WIDE, as well as MIRFLICKR-25K.

III. METHODOLOGY

A. TEXTURE PATTERN ANALYSIS

A set of small units combined to make an image texture, images of the same category usually have a similar texture. Texture is categorized using the JD (joint distribution) of concentration values over tremendously condensed neighborhoods (initial from as small as three-into-three pixels square), also it outstrips classification spending filter banks with massive support. We applied the MRF (Markov Random Field) model classifier, which does not use large scale filter banks while using local pixel neighborhoods directly, more extraordinary synthesis results could be achieved [16]. Outstanding classification performance had earned from small neighborhoods of size 3 × 3, 5 × 5, and 7 × 7. In MRF, filter banks have replaced; for a particular point on an image, underdone pixel concentrations of an $M \times M$ equal four-sided neighborhood nearby that particular point has been occupied, where M^2 dimensional feature space to form a vector row reordered.

Formally MRF classification performance for texture realization can be measured as [17]

$$p\left(\frac{I\left(a_x\right)}{I\left(a\right)}, \forall a \neq a_x\right) = p(I(a_x)|I\left(a\right), a\epsilon M\left(a_x\right) \qquad (1)$$

TABLE 5. **Precision, recall and fscore for Caltech 256 (Categories 1-100).**

No.	Category	Precision	Recall	Fscore	No.	Category	Precision	Recall	Fscore
1.	airplanes-101	0.9	0.11	0.2	**51.**	cormorant	0.4	0.25	0.31
2.	ak47	0.15	0.67	0.24	**52.**	covered-wagon	0.25	0.4	0.31
3.	american-flag	0.15	0.67	0.24	**53.**	cowboy-hat	0.3	0.33	0.32
4.	backpack	0.15	0.67	0.24	**54.**	crab-101	0.1	1	0.18
5.	baseball-bat	0.2	0.5	0.29	**55.**	desk-globe	0.55	0.18	0.27
6.	baseball-glove	0.4	0.25	0.31	**56.**	diamond-ring	0.75	0.13	0.23
7.	basketball-hoop	0.15	0.67	0.24	**57.**	dice	0.1	1	0.18
8.	bat	0.1	1	0.18	**58.**	dog	0.1	1	0.18
9.	bathtub	0.2	0.5	0.29	**59.**	dolphin-101	0.45	0.22	0.3
10.	bear	0.45	0.22	0.3	**60.**	doorknob	0.1	1	0.18
11.	beer-mug	0.15	0.67	0.24	**61.**	drinking-straw	0.1	1	0.18
12.	billiards	0.5	0.2	0.29	**62.**	duck	0.2	0.5	0.29
13.	binoculars	0.25	0.4	0.31	**63.**	dumb-bell	0.25	0.4	0.31
14.	birdbath	0.1	1	0.18	**64.**	eiffel-tower	0.15	0.67	0.24
15.	blimp	0.1	1	0.18	**65.**	el	0.15	0.67	0.24
16.	bonsai-101	0.15	0.67	0.24	**66.**	electric-guitar-101	0.3	0.33	0.32
17.	boom-box	0.15	0.67	0.24	**67.**	elephant-101	0.5	0.2	0.29
18.	bowling-ball	0.15	0.67	0.24	**68.**	elk	0.6	0.17	0.26
19.	bowling-pin	0.1	1	0.18	**69.**	ewer-101	0.1	1	0.18
20.	boxing-glove	0.3	0.33	0.32	**70.**	eyeglasses	0.8	0.13	0.22
21.	brain-101	0.35	0.29	0.31	**71.**	faces-easy-101	0.9	0.11	0.2
22.	breadmaker	0.2	0.5	0.29	**72.**	fern	0.3	0.33	0.32
23.	buddha-101	0.1	1	0.18	**73.**	fighter-jet	0.1	1	0.18
24.	bulldozer	0.15	0.67	0.24	**74.**	fire-extinguisher	0.15	0.67	0.24
25.	butterfly	0.1	1	0.18	**75.**	fire-hydrant	0.1	1	0.18
26.	cactus	0.15	0.67	0.24	**76.**	fire-truck	0.8	0.13	0.22
27.	cake	0.1	1	0.18	**77.**	fireworks	0.15	0.67	0.24
28.	calculator	0.1	1	0.18	**78.**	flashlight	0.15	0.67	0.24
29.	cannon	0.15	0.67	0.24	**79.**	floppy-disk	0.2	0.5	0.29
30.	canoe	0.1	1	0.18	**80.**	football-helmet	0.3	0.33	0.32
31.	car-side-101	0.6	0.17	0.26	**81.**	french-horn	0.8	0.13	0.22
32.	car-tire	0.1	1	0.18	**82.**	fried-egg	0.25	0.4	0.31
33.	cartman	0.1	1	0.18	**83.**	frisbee	0.15	0.67	0.24
34.	cd	0.1	1	0.18	**84.**	frog	0.25	0.4	0.31
35.	centipede	0.15	0.67	0.24	**85.**	frying-pan	1	0.1	0.18
36.	cereal-box	0.1	1	0.18	**86.**	galaxy	0.15	0.67	0.24
37.	chandelier-101	0.15	0.67	0.24	**87.**	gas-pump	0.15	0.67	0.24
38.	chess-board	0.1	1	0.18	**88.**	giraffe	0.25	0.4	0.31
39.	chimp	0.3	0.33	0.32	**89.**	goat	0.2	0.5	0.29
40.	chopsticks	0.1	1	0.18	**90.**	golden-gate-bridge	0.15	0.67	0.24
41.	clutter	0.6	0.17	0.26	**91.**	goldfish	0.1	1	0.18
42.	cockroach	0.15	0.67	0.24	**92.**	golf-ball	0.15	0.67	0.24
43.	coffee-mug	0.05	2	0.1	**93.**	goose	0.25	0.4	0.31
44.	coffin	0.1	1	0.18	**94.**	gorilla	0.65	0.15	0.25
45.	coin	0.15	0.67	0.24	**95.**	grand-piano-101	1	0.1	0.18
46.	comet	0.2	0.5	0.29	**96.**	grapes	0.5	0.2	0.29
47.	computer-keyboard	0.1	1	0.18	**97.**	grasshopper	0.45	0.22	0.3
48.	computer-monitor	0.1	1	0.18	**98.**	greyhound	0.55	0.18	0.27
49.	computer-mouse	0.1	1	0.18	**99.**	guitar-pick	0.5	0.2	0.29
50.	conch	0.1	1	0.18	**100.**	hamburger	0.4	0.25	0.31

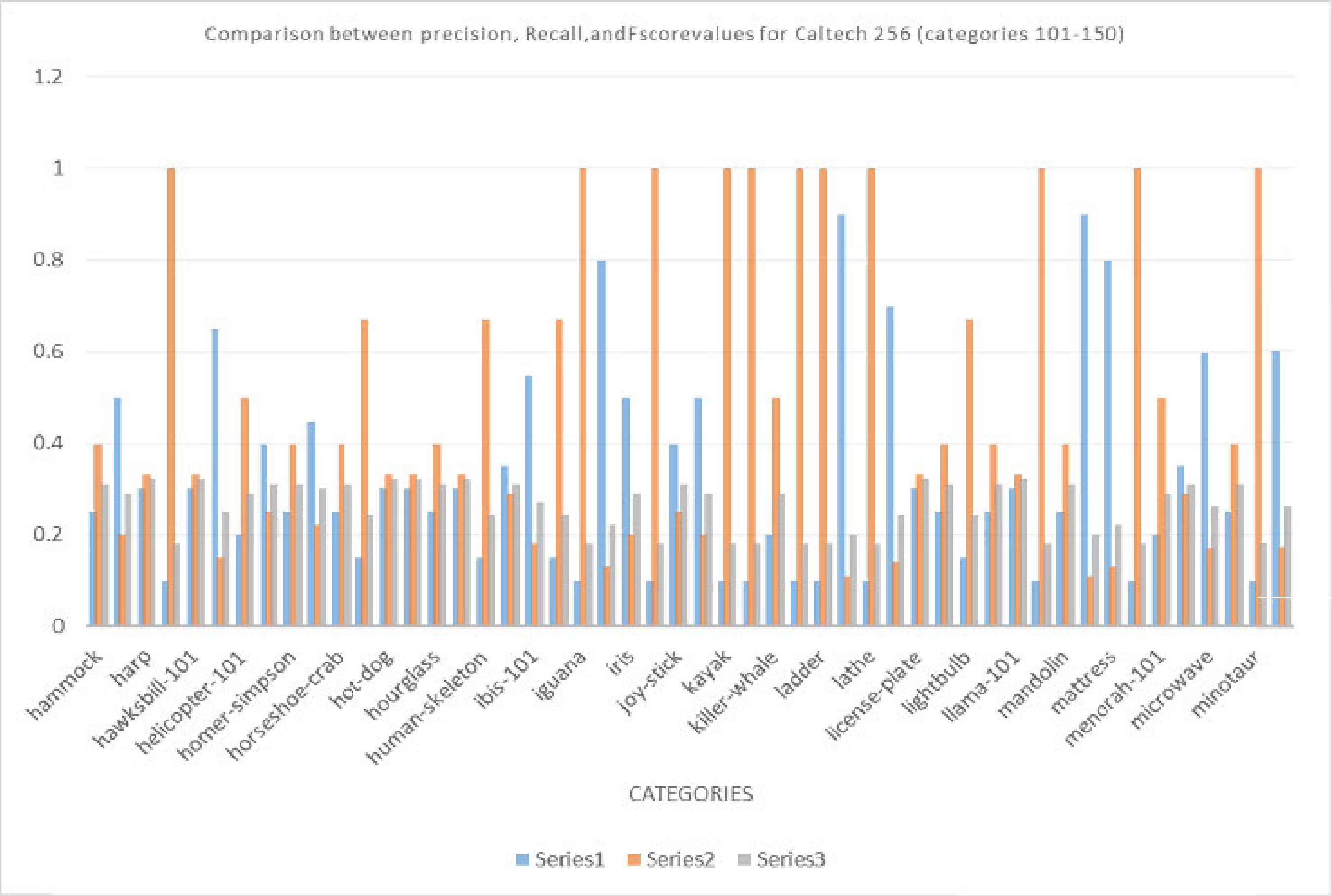

FIGURE 10. Comparison between precision, recall, and F-score values for Caltech 256 (categories 101-150).

In the 2-dimensional integer grid on which the image I have been defined, a_x is a spot on it and $M(a_x)$ is the neighborhood of that spot. Excluding the central pixel, M has defined as the $M \times M$ square neighborhood. Neighbors of a central pixel trained the distribution of it, although its value itself is weighty. In the neighborhood classifier, the central pixel has left out, i.e., in a 3×3 neighborhood, only eight neighbors are concerned with each central pixel to create feature vectors and textons. When we put the value of $\mathrm{M} = 5$, far better results of classification have been obtained as compared to $\mathrm{M} = 3$ and $\mathrm{M} = 7$. This result satisfied that for classification, the joint distribution of the neighbors is a decent selection.

When the MRF model explicitly is seen then instead of overlooking the central pixel value, do $p(I(a_x)|I\ (a)\ , a\epsilon M\ (a_x)$, i.e., neighbor values affect the distribution of central pixels. Rather than using texton, we will focus on neighborhooding. On applying texton as a neighborhood, i.e., in an $\mathrm{M}^2 - 1$-dimensional space, each and every pixel value except the central are castoff to an arrangement of feature trajectories, categorized by the phrasebook of textons. For each of the n textons sequentially, a 1-D distribution of the vital pixels' intensity is learned and denoted as 'm' bin histogram. The Joint Probability Density (JPD) function has now been represented as an n $\times$ m matrix. Clusters have formed across the image due to the co-occurrence of neighbors; because of texture homogeneous repeated behavior with small statistical dissimilarities. It is concluded that a Cluster-based illustration of the Probability Density Function via a few textons should be sufficient even in excessive dimensional places; meanwhile, the maximum of the area is blank and does not need to be displayed. By comparing the maximum response of the Neighborhood classifier, the VZ classifier consuming the MR8 filter bank to that of the Joint classifier (JC) and MRF classifier as the dimensions of the neighborhood is different [17]. The MR8 filter bank has scrambled dejected the support of the most extensive scale filters is similar to the neighborhood size.

The MR8 classifier performs not as good as the Joint classifier (JC) and so do the neighborhood classifier for any specified size of the neighborhood. A superior classifier could be defined as which is capable of learning from absolute pixel values. The MRF classifier attains 97.8 percent using a seven-into-seven neighborhood with 2440 textons and 90 bins. The number of models compacts by GA (Greedy Algorithm) to

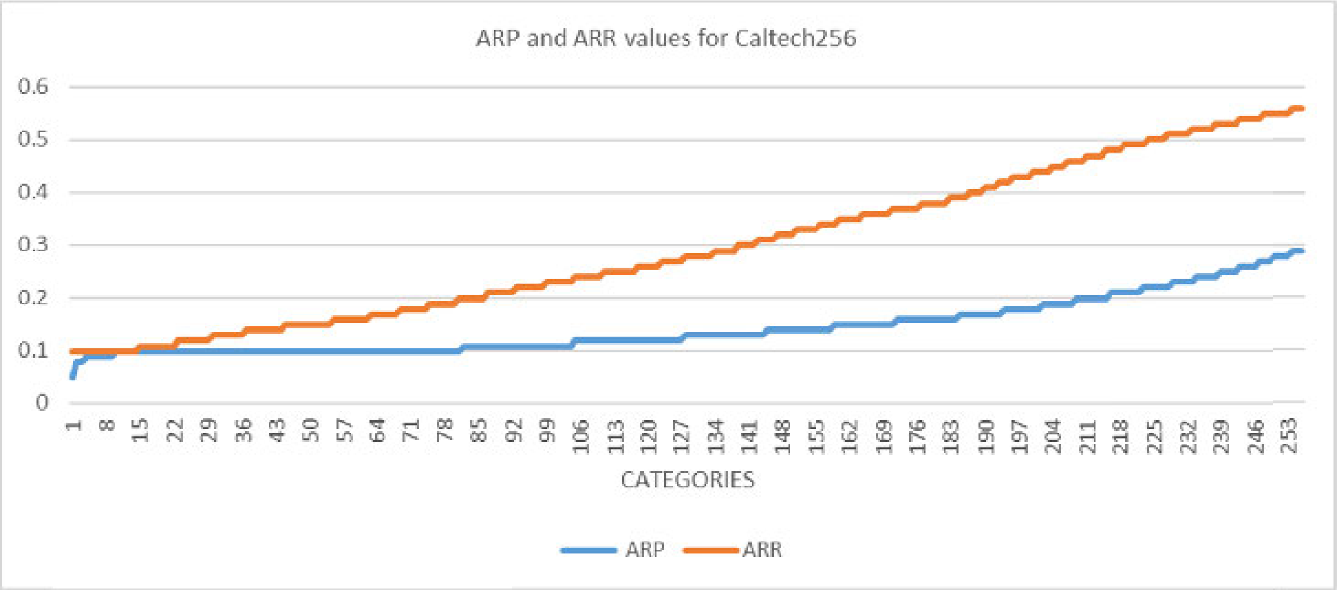

FIGURE 11. Comparison between ARP and ARR for caltech 256 (categories 101-150).

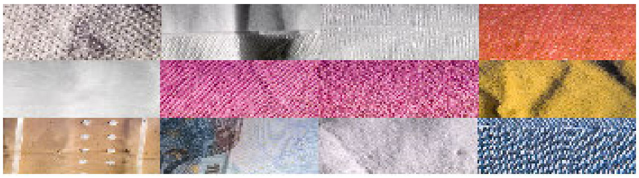

FIGURE 12. ALOT (250) dataset showing different sample images of categories falling in fabric texture.

7.27 on average. For synthesis and classification of a texture, filter, banks are not necessary.

B. DESCRIBING KEY POINTS AND MATCHING THEM FOR OBJECT RECOGNITION

Inspired to the humanoid pictorial system and, more accurately, the retina, FREAK (Fast Retina Key Point) is used for interest point description for object recognition in images [18]. A cascade binary string of 16 bytes is figured out by proficiently relating image intensities following a retinal specimen pattern. BRISK is a twofold descriptor invariant to scale and rotation, used in a specific sampling pattern to limit crucial points. Computer vision followed the human retina pattern as follows fig 2.

Sampling grids are probable to match and describe by comparing several pairs of pixels of simple intensity. A circular bounding pattern is used by BRISK [19], in which points have equally spaced on a circular central point. The removal of noise had done by smoothing each point. Trying to be similar to the retina model for every sample point, we will use different kernel sizes. The standard deviation of the Gaussian kernels had functional to each specimen point. As we lead towards better performance, so concerning the log-polar retinal pattern changed, the size of the Gaussian kernels is preferred. Furthermore, adding redundancy and overlapping the receptive fields increases not only the performance but also discriminative power.

TABLE 6. Value of precision represented in tabular form for categories falling in fabric type in ALOT dataset.

No.	folder number	Precision	No.	folder number	Precision
1	17	0.97	12	111	0.99
2	18	0.86	13	137	0.95
3	38	0.78	14	144	1
4	43	0.88	15	180	1
5	58	0.96	16	181	0.9
6	60	0.98	17	182	1
7	61	0.67	18	185	1
8	67	0.96	19	207	1
9	95	0.99	20	209	0.92
10	96	0.98	21	215	0.98
11	106	0.99			

Let's ponder the approachable fields *P, Q, R* with the intensities I_i, in a way that[17];

$$\mathrm{I_P} > \mathrm{I_Q},\quad \mathrm{I_Q} > \mathrm{I_R},\quad \text{and } \mathrm{I_P} > \mathrm{I_R} \tag{2}$$

On the off chance that the fields don't have overlap, then the last test $I_P > I_R$ isn't including any discriminant data.

Overlain fields cause redundancy; which is correspondingly part of the entire receptive fields of the retina. Compute the dissimilarity between critical points for a couple of

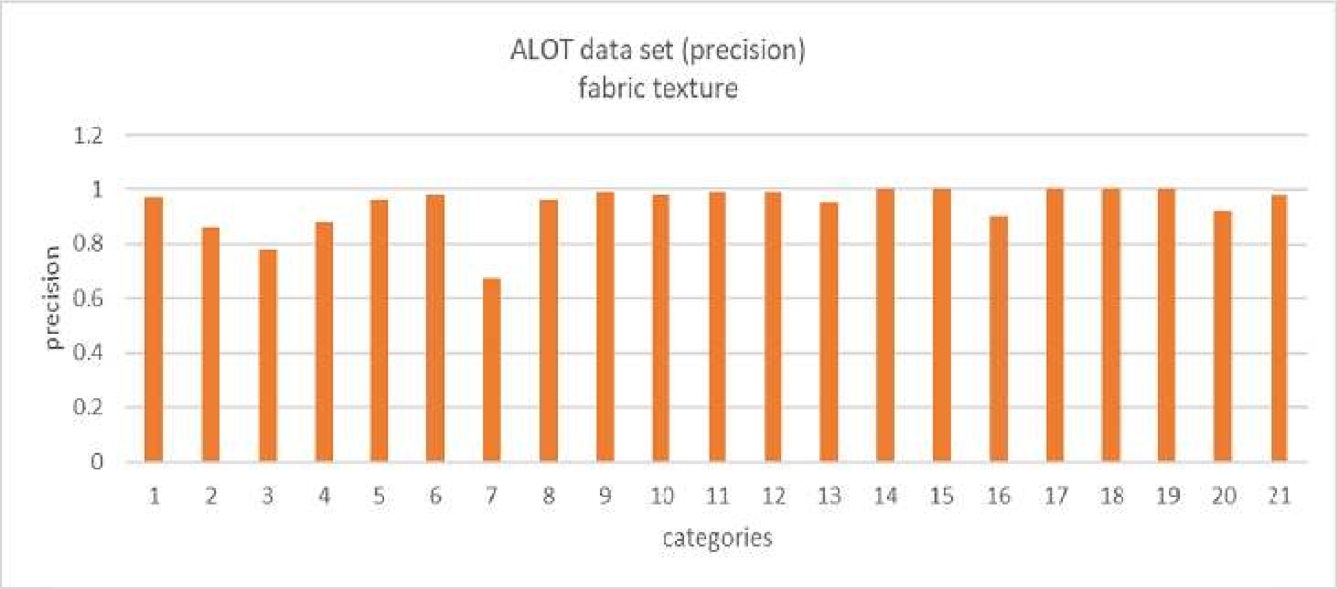

FIGURE 13. The precision rate for categories falling in fabric texture type available in the ALOT (250) dataset.

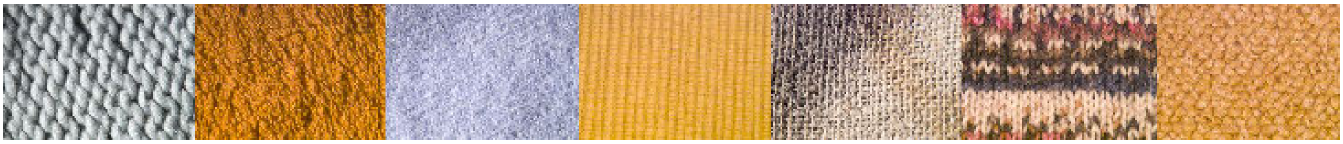

FIGURE 14. ALOT (250) dataset showing different sample images of categories falling in the weaving pattern.

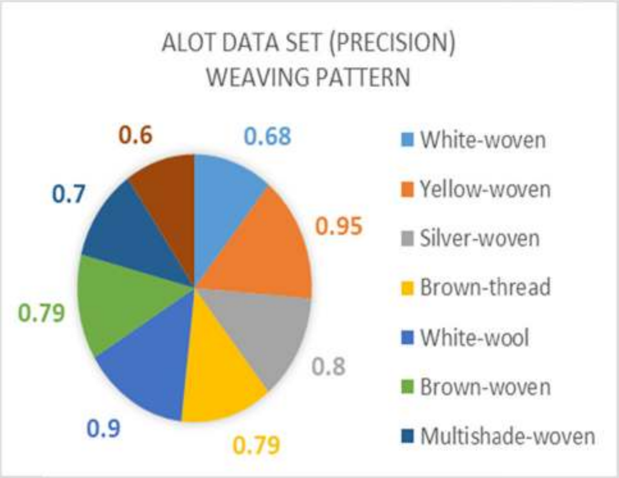

FIGURE 15. The precision rate for categories falling in weaving pattern available in ALOT 250 dataset.

receptive fields, the analogous Gaussian kernel had used. S is a 'string' shaped by the structure of the 1-bit Difference of Gaussians (DoG) [17].

$$S = \sum\nolimits_{0 \leq d < k} 2^d M(L_d) \tag{3}$$

TABLE 7. Value of precision represented in tabular form for categories falling in weaving pattern in ALOT dataset.

No.	folder number	Precision	Recall	F-score
1	34	0.68	0.15	0.24
2	48	0.95	0.11	0.19
3	55	0.8	0.13	0.22
4	64	0.79	0.13	0.22
5	74	0.9	0.11	0.2
6	109	0.79	0.13	0.22
7	177	0.7	0.14	0.24
8	205	0.6	0.17	0.26

Where $\mathrm{L_d}$ is a couple of receptive fields, k is the preferred size of the descriptor, then[17]

$$\mathrm{M(L_d)} = \begin{cases} One\ if\ I(L_d^{r_1} - L_d^{r_2}) > 1 \\ 0\ \ otherwise \end{cases} \tag{4}$$

With $I(L_d^{r_1})$ is the leveled intensity of the 1st receptive field of the duo $\mathrm{L_d}$.

Enormous pairs had obtained due to the mishmash of receptive fields. Not every pair need to transmit beneficial

TABLE 8. Value of precision represented in tabular form for categories falling in fruit texture in ALOT dataset.

No	folder name	Precision	No	Folder name	Precision
1	11	0.65	**12**	200	1
2	45	0.77	**13**	216	0.95
3	59	0.87	**14**	217	0.7
4	72	0.85	**15**	218	0.7
5	73	0.97	**16**	219	1
6	90	0.77	**17**	220	0.55
7	122	0.9	**18**	221	1
8	126	0.69	**19**	222	0.85
9	167	0.45	**20**	234	0.9
10	168	1	**21**	237	0.87
11	169	0.8	**22**	238	1

TABLE 9. Value of precision represented in tabular form for categories falling in leaf type in the ALOT dataset.

Category	folder name	Precision	category	folder name	Precision
1	14	1	10	129	0.97
2	16	0.98	11	163	0.55
3	20	0.75	12	164	0.89
4	22	0.75	13	119	1
5	42	0.98	14	203	0.83
6	79	0.97	15	204	0.85
7	80	0.75	16	208	0.93
8	101	1	17	212	0.9
9	110	0.9			

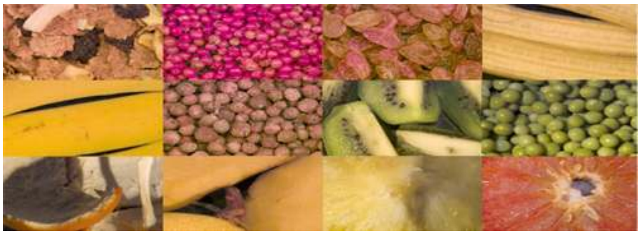

FIGURE 16. ALOT (250) dataset showing different sample images of categories falling in fruit texture.

information, so one tactic is to do a computation of their spatial difference. Loss of some valuable information takes place by applying this exceedingly correlated pairs. Another algorithm to select suitable pair, make a matrix A such that A ≈ 50 nK where n = 1000, and K indicates extracted vital points. Each row of A denotes key points that occur in the retina sampling arrangement using 43 accessible fields. To find the discriminant features, we calculate the Mean of each stake and to attain the wanted rate of the alteration. The highest conflict had reported in the case of the mean value is 0.5. Columns had organized in ascending in the next step order relating to the amount of variance. A stake with a small correlation had added to the queue using a mean value of 0.5. In that couple course of action, a coarse-to-fine organization of Difference of Gaussians happened. To estimate the position of an object, the perifoveal receptive field is implemented. In the fovea area for the validation procedure, compact distributed receptive fields had been used. In saccadic searching viewer eye agitate with irregular singular movement, FREAK descriptor work on a similar rule. To find out the exact location of the desired object, high-resolution information is required—fovea, which has a high-density photoreceptor, capture the high resolution. In contrast, low-frequency information is captured by perifoveal. Further proceeding

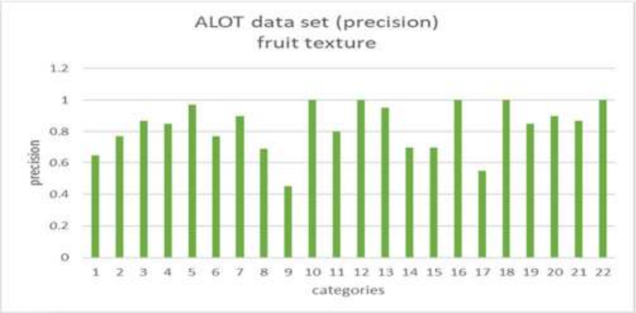

FIGURE 17. Precision rate for categories falling in fruit texture available in ALOT 250 dataset.

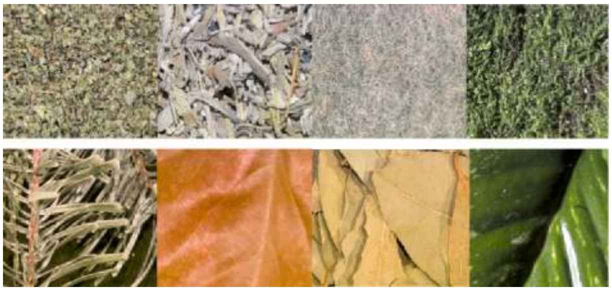

FIGURE 18. ALOT (250) dataset demonstrating dissimilar model images of categories falling in leaf texture.

TABLE 10. Value of precision represented in tabular form for categories falling in sponge texture in ALOT database.

No	folder number	Precision	Recall	F-score
1	23	0.9	0.11	0.2
2	24	0.98	0.1	0.18
3	25	0.75	0.13	0.23
4	26	0.89	0.11	0.2
5	166	0.99	0.1	0.18
6	176	0.67	0.15	0.24

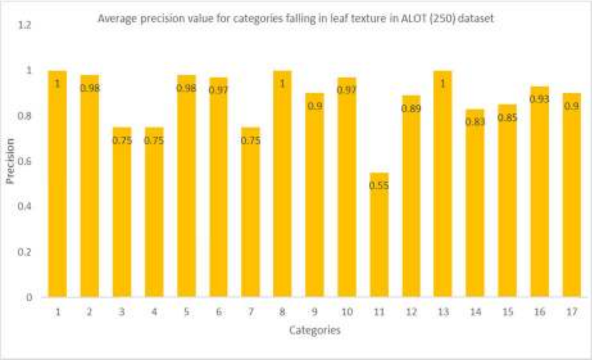

FIGURE 19. Precision rate for categories falling in leaf texture available in ALOT 250 dataset.

FIGURE 20. ALOT (250) dataset showing different sample images of categories falling in sponge texture.

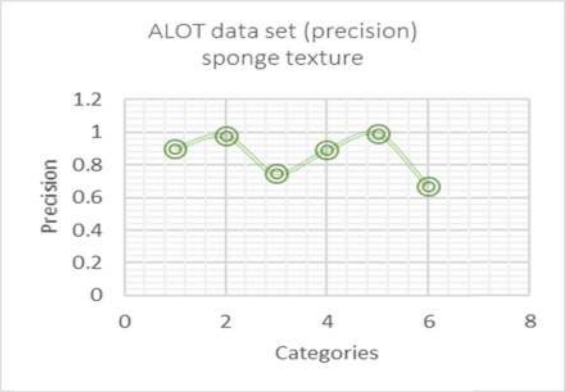

FIGURE 21. Precision rate for categories falling in sponge texture available in ALOT 250 dataset.

FIGURE 22. ALOT (250) dataset showing different sample images of categories falling in seashell type.

analysis is done with the following bytes to investigate more acceptable level data.

In short, we comprise the whole saccadic research practically such that a cascade of comparison is performed for matching, most of the entrants are cast-off with the initial 16 bytes of FREAK descriptor. The same object is to search in the next image and so on in the Intel processor, which works on it.

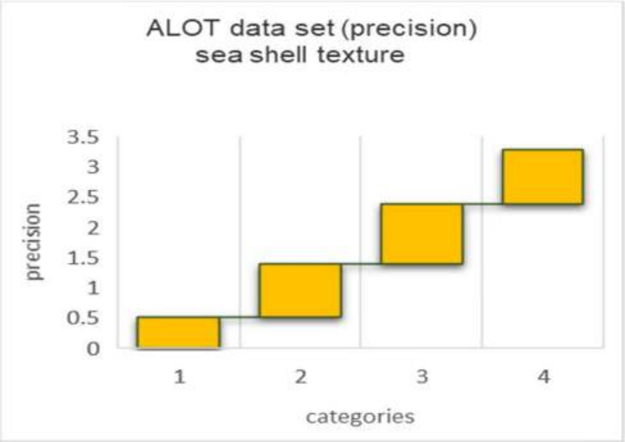

FIGURE 23. Average precision results for seashell texture in ALOT dataset.

TABLE 11. Value of precision represented in tabular form for categories falling in seashell type in alot dataset.

No.	folder number	Precision	Recall	F-score
1	8	0.5	0.2	0.29
2	31	0.88	0.11	0.2
3	174	0.99	0.1	0.18
4	175	0.9	0.11	0.2

Single Instruction and Multiple Data (SIMD) as operations have accomplished in parallel, so to relate 1 byte or 16 bytes is almost the same. An object of interest is described by FREAK through the size of its bounding circles. The first cascade of 16 bytes discards many candidates fewer selected for the next comparison at last, and the final cascade gives the desired results irrespective of the illumination and viewpoints.

Sum is applied to predictable local gradients over particular pairs analogous to BRISK so that the approximation is resulted from the rotation of these key point. The last one for computation of global orientation is using long pairs, whereas primarily the teams with symmetric amenable fields is selected for the center. Let J be all duos used to define local gradients, then [1]

$$J = \frac{1}{N} \sum\nolimits_{L_0 \epsilon J} (I(L_o^{r_1}) - I\left(L_o^{r_2}\right)) \frac{L_o^{r_1} - L_o^{r_2}}{||L_o^{r_1} - L_o^{r_2}||} \tag{5}$$

where N defining the number of duos in J and $L_o^{r_1}$ donating 2-D vector of the three-dimensional matches; of the middle of the receptive field.

C. EXTRACTION OF SPATIAL COLOR FEATURES

This approach is used to solve the image retrieval issues like sub-region querying, object localization, and object detection. Color histogram, although solve these issues somehow as it focuses on color distribution but does not deliberate the spatial correlation statistics. Suppose ther is an image Ï with a × a dimension where the quantized colors had represented

FIGURE 24. ALOT (250) dataset showing different sample images of categories falling in lego texture.

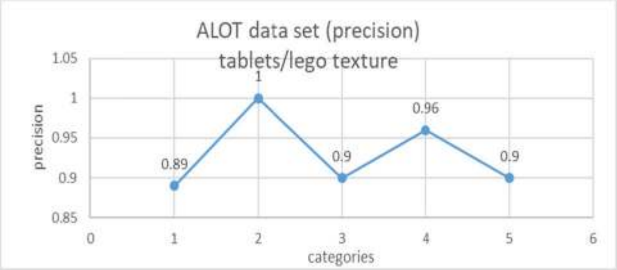

FIGURE 25. The precision rate for categories falling in tablets/lego texture available in ALOT 250 dataset.

as **k** colors

$$P = (s, t) \in \ddot{I}$$

For pixels P_1 (s_1, t1) and P_2 (s_2, t_2) distance metric is defined as [20]

$$|P_1 - P_2| \triangleq \max\{|(s_1 - s_2)|, |t_1 - t_2|\} \qquad (6)$$

The color histogram H of an image Ï could be defined as for [20]

$$H_{k_i}(\ddot{I}) \triangleq l^2.\Pr[P \in \ddot{I}_{k_i}] \qquad (7)$$

P∈ Ï $\boldsymbol{H_{k_i}}$ (Ï) /$\boldsymbol{l^2}$ Gives the probability that the color of the pixel is k_i.

A linear function H representing a histogram had computed O (l^2) times. Suppose distance has fixed [20]

$$i, j \in [k], m \in [d]$$

$$\beta^m_{k_i,k_j}(\ddot{I}) \triangleq Pr_{\substack{P_1 \in \ddot{I}k_i \\ P_2 \in \ddot{I}}} [P_2 \in \ddot{I}_k \, |P_1 - P_2| = m] \qquad (8)$$

Equation (8) signifies in the spatial image organization of color pixels. The above Equation (8) β represents the likelihood that a color pixel k_i at space m far from a given color pixel. The spatial correlation between the identical color esteems is describe in Equation (9)[20] as following:

$$\alpha^m_k(\ddot{I}) \triangleq \beta^m_{k,k}(\ddot{I}) \qquad (9)$$

By considering eq (8), eq (9) could be derived, where α had denoted by the likelihood of k color pixel with m distance.

D. CNN ARCHITECTURES

1) VGG 19

One of the CNN based architectures being proposed is VGG 19 network with a multilayered operation. It consists of 16 convolutional layers for feature extraction at the training phase [14], for transfer learning, 19 learnable weights layers utilized, three fully connected (FC) layers, and at the termination, an output layer. For feature withdrawal from the inserted images at the earliest convolutional layer, 64 kernels (3 × 3 filter size) are applied [21].

A progressive pre-trained CNN model is inception. It contains 316 layers and 350 connections. The number of convolution layers is 94 of different kernels/filter sizes, where the magnitude of the 1st input layer is 299 × 299 × 3. Far ahead, the batch normalization and ReLu activation layers are supplementary. A max-pooling layer had also been inserted between convolution layers. Finally, in the testing phase, to categorize the images, grounded on the softmax activation procedure, 10-fold cross-validation is functional. The performance of the proposed VGG-19 based system is compared with other features obtained with another CNN architecture GoogLeNet.

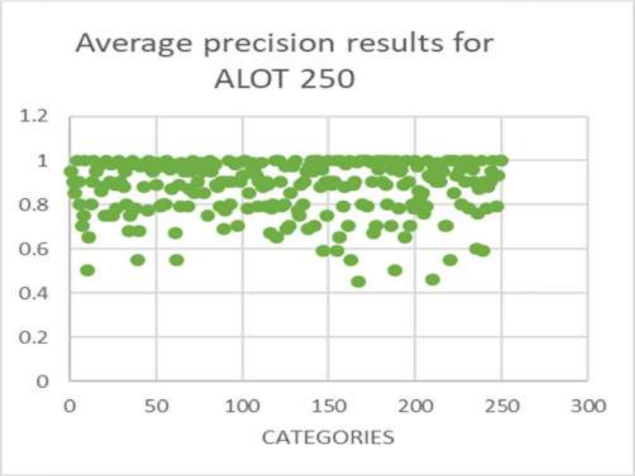

FIGURE 26. Average precision results for ALOT dataset.

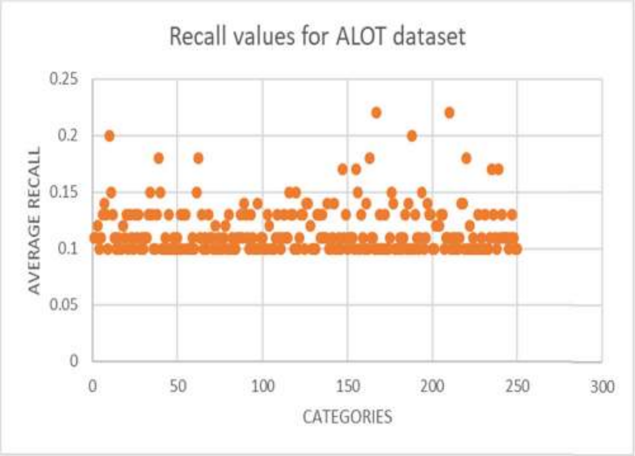

FIGURE 27. Recall values for ALOT250 dataset.

2) GOOGLENET ARCHITECTURE

In ILSVRC 2014, GoogLeNet, which is positioned first, is a 22-layer deep convolutional network that consists of nine

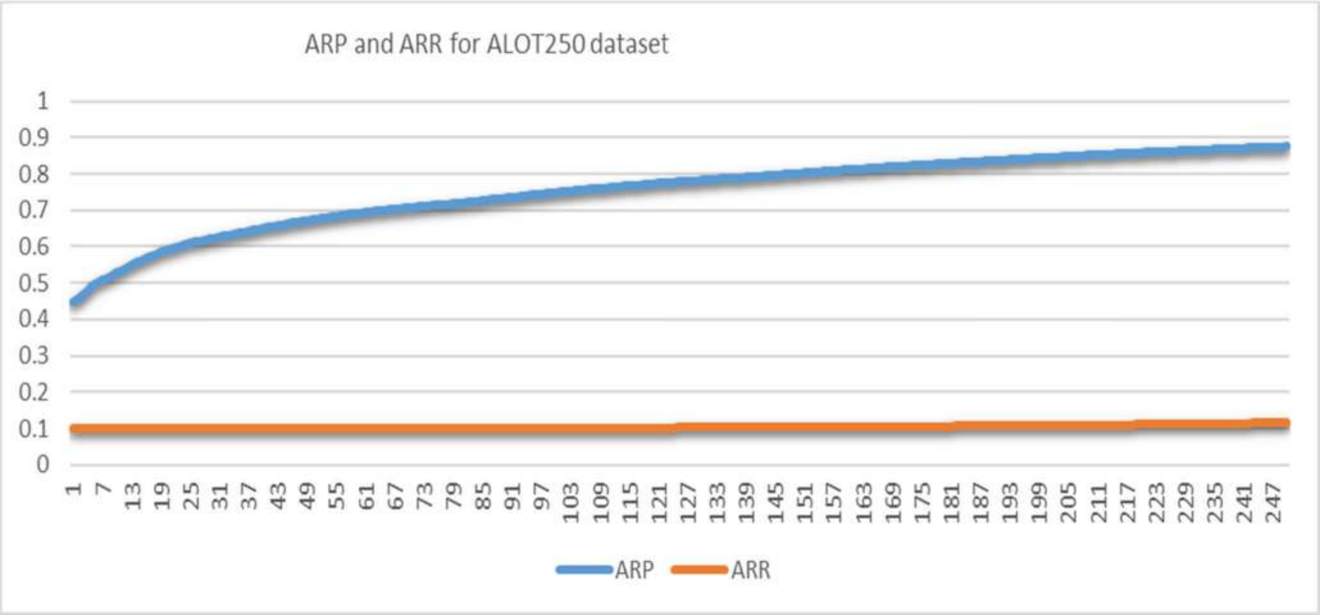

FIGURE 28. ARP and ARR for ALOT250 dataset.

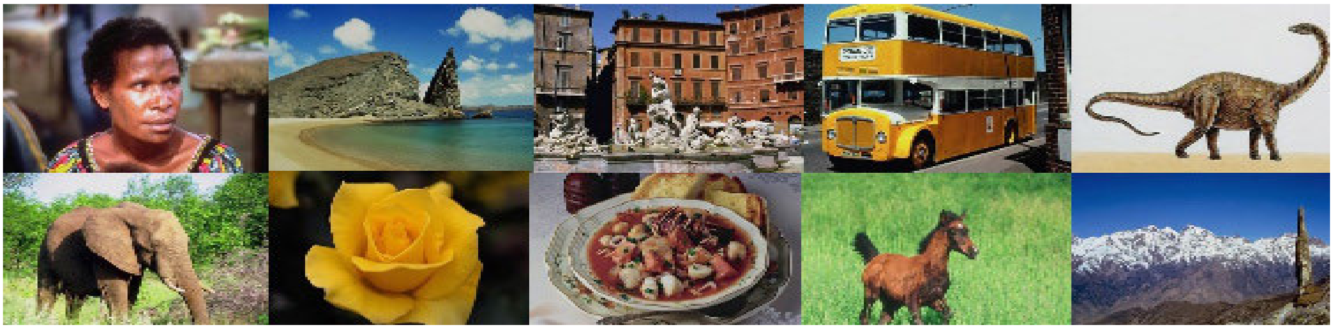

FIGURE 29. Different model images of corel-1k (1000) dataset showing from per capita category.

"inception modules" set on the head of one another. Every inception module is confined to kernel sizes of 1 × 1, 3 × 3, and five-into-five to keep away from fixed arrangement issues. To decrease computational unpredictability, one-into-one convolutions are utilized first for dimensionality decrease before the costly 3 × 3 and five × 5 convolutions. GoogLeNet eliminates associated layers and uses the successive layer to adjust to other name sets [22].

E. IMAGE RANKING AND INDEXING USING BOW

At the last step, after assembling deep-textured, cascade, color features, the next Bag-of-Words (BoW) model is implemented for the reckless image indexing and retrieval. Each image in the BoW model is denoted by an isolated linear vector. BoW model pays the powers of local feature descriptors such as SIFT (Scale Invariant Features Transform), and it increases its strength. Secondly, by easy manipulation, a single vectored BoW contrast is accomplished by using a divergence score. Besides, the sparse space design of massive dimensional data results in fast searching and indexing. The patches are represented as numerical vectors by SIFT. Equal size dimensional vectors SIFT created, which are 128.

Clustering is performed on these local features; cluster center represented as code words. For each image, the event check of each codeword (additionally called visual words) is spoken to as a histogram. Because of histograms, a reversed image list is produced for effective image retrieval. Each list speaks to one visual word. Mapping the terms with images, a list of images is created. Many images with analogous visible words are indexed contingent on the query image by applying the association of plans.

Finally, the picture positioning is accomplished by including the quantity of visual words communal among the inquiry image and ordered images. An image carries its position to top, which shares a higher number of words. Proposed spatial color withdrawal strategy installs the spatial statistics in feature vectors at the hour of highlight extraction,that brings about more significant recovered images as BoW unable to do so.

TABLE 12. Precision, F-score, and recall values for corel 1000 dataset.

Corel 1000				
No.	category name	precision	Recall	F-score
1	African people and village	0.9	0.11	0.2
2	Beach	0.85	0.12	0.21
3	Buildings	1	0.1	0.18
4	Buses	0.99	0.1	0.18
5	Dinosaurs	0.95	0.11	0.2
6	Elephants	1	0.1	0.18
7	Flowers	0.98	0.1	0.18
8	Food	0.9	0.11	0.2
9	Horses	1	0.1	0.18
10	Mountains	0.8	0.13	0.22

TABLE 13. ARP and ARR values for corel 1000.

P ascending order	ARP	R ascending order	ARR
0.8	0.8	0.1	0.1
0.85	0.825	0.1	0.1
0.9	0.85	0.1	0.1
0.9	0.8625	0.1	0.1
0.95	0.88	0.1	0.1
0.98	0.896667	0.11	0.101667
0.99	0.91	0.11	0.102857
1	0.92125	0.11	0.10375
1	0.93	0.12	0.105556
1	0.937	0.13	0.108

IV. EXPERIMENTATION

A. DATA SETS

Proposed Image retrieval system design maximum response and accuracy are tried on appropriate datasets. Datasets are selected based on project nature. Results accuracy depends upon the visual features of images like shape, color, texture, and shape and cluttering, and occlusion, while different datasets have great versatility in these. Three other standardized datasets are used to perform these experiments named Caltech-256, ALOT (250), Corel-1000, Cifar 100, and Cifar10. Diversity of characteristics, which datasets preserve, are various image categories; images are taken from different areas, and types have multiple objects placed in the background and foreground. Retrieval system results centered on the choice of subcategories, which signifies the particular database, and comprise of numerous categories from diverse areas.

B. INPUT PROCESS

Firstly a color image selected from a dataset is inserted into the system and converted into a greyscale image. The idea is also inserted to CNN architectures VGG19 and Googlenet for in-depth features, MRF classifiers for a textured pattern, and through color, channels to attain the color features. The input images are taken from Caltech-256, Corel-1000, and ALOT (250). Succeeding step images from datasets are passed through training and testing with 70% and 30%. For the query image features are extracted, and at the end BoW is utilized for directory and examine the t-nearest images in the database. The proposed descriptor can distinguish prospective objects, textures, figures, and colors.

C. OBTAINED RESULTS AND DISCUSSION

1) EVALUATION OF PRECISION AND RECALL

Performance accuracy is evaluated by Precision and recall metrics. Precision is the positive anticipated value and recall is the true positive (TP) rate estimation. Precision and recall are counted for each category. [17]

$$\mathrm{Precision} = \mathrm{M_{i(r)}/M_{o(r)}} \tag{10}$$

$$\mathrm{Recall} = \mathrm{M_{i(r)}/M_{p}} \tag{11}$$

While $\mathrm{M_{i(r)}}$ signifies the applicable images corresponding to the requested image, $\mathrm{M_{o(r)}}$ signifies the recovered images alongside the requested image and the overall amount of relevant images in existing database.

2) CALTECH-256 DATASET RESULTS

To check the efficiency of the proposed method, the complete Caltech-256 dataset with 257 categories is experimented and the precision is calculated for each category. Caltech-256 is an extensive dataset comprises of 30,607 images. As the dataset is explored, images are further grouped into more than 200 categories. 256_Object Categories or 'Caltech-256' contains multipart images as paralleled to the previous dataset 'Caltech-101'. At first, we made almost 40 different common types (to reduce the complexity in dealing 256 categories) like animals, birds, crockery, plants, balls, vehicles, food, cartoon characters, insects, sea animals, computer hardware, music, fruits, scientific instruments, etc in these multiple categories are falling. The selected categories falling in one type contribute to common characteristics of that type like shapes, common features, objects, and cluttered scenes. Within categories of dataset, image size is not static and differs from type to type. The proposed method outclasses in most of the classes since the technique reflects the local and global features innovatively. Concentration points constructed on extremely stable sections, shape, and texture pattern combine result in improved truthfulness in maximum image classes. Additionally, it is witnessed that the projected method displays noteworthy outcomes for the cluttered object with overlap, and multipart background images. 256 Object Categories also known as Caltech-256 is a thought-provoking dataset including the multifaceted nature of images. It is grouped with similar type of categories together to calculate the recision of our proposed model. Some of the categories that group in a specific type results are discussed here.

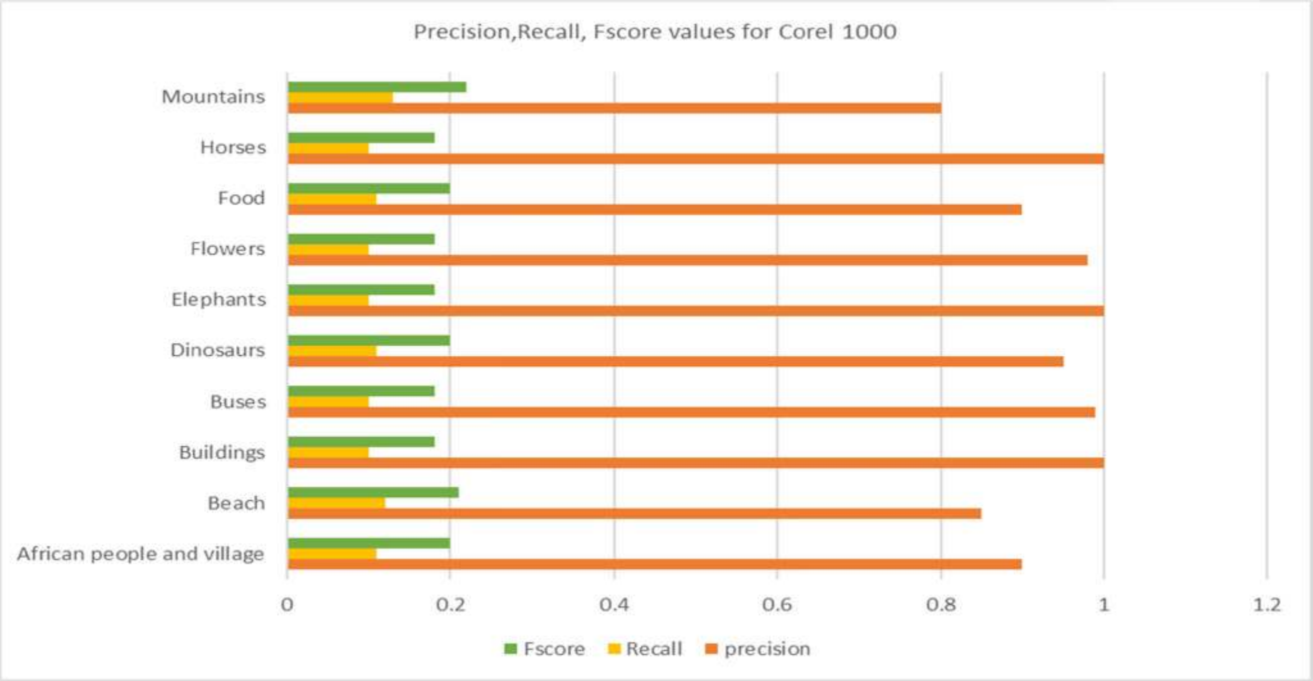

FIGURE 30. Precision, fscore and recall values for corel 1000 dataset.

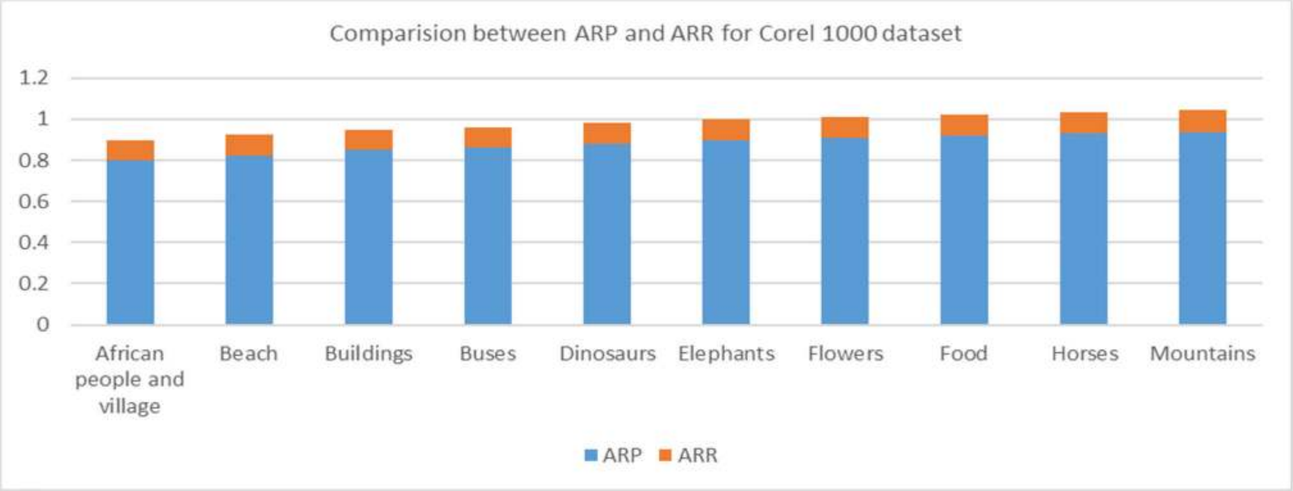

FIGURE 31. ARP and ARR values for corel 1000.

a: BIRDS TYPE CATGORIES

Different types of bird's categories are available in our dataset that is counted as ten categories named as a bat, cormorant, duck, goose, hummingbird, ibis-101, ostrich, owl, penguin, and swan.

These birds detect by our model by their standard features like feathers, beak, claws, etc. Results show that ibis has the highest 0.55%, and the bat has the lowest precision 0.1% value because bat came in the category of reptiles. The precision value of each category falling in bird type is mentioned in tabular form as follows:

Each type has same basic features like wings, main body, and tail but difference came colors, shape edge detection. The precision of airplanes-101 is highest while the lowest for a blimp. Because the shape of the blimp is slightly different from the airplanes so our proposed method differentiates accurately and outperforms very well. Results of our proposed model is demonstrated in tabular as well as in graphical form.

b: TYPE OF AIRCRAFT CATEGORIES

Aircraft types of 5 classes are present in the dataset, namely airplanes-101, blimp, fighter-jet, helicopter, and hot-air-balloon. Background is mostly same for all categories the difference came when it is focused on the foreground objects. Each type has same basic features like

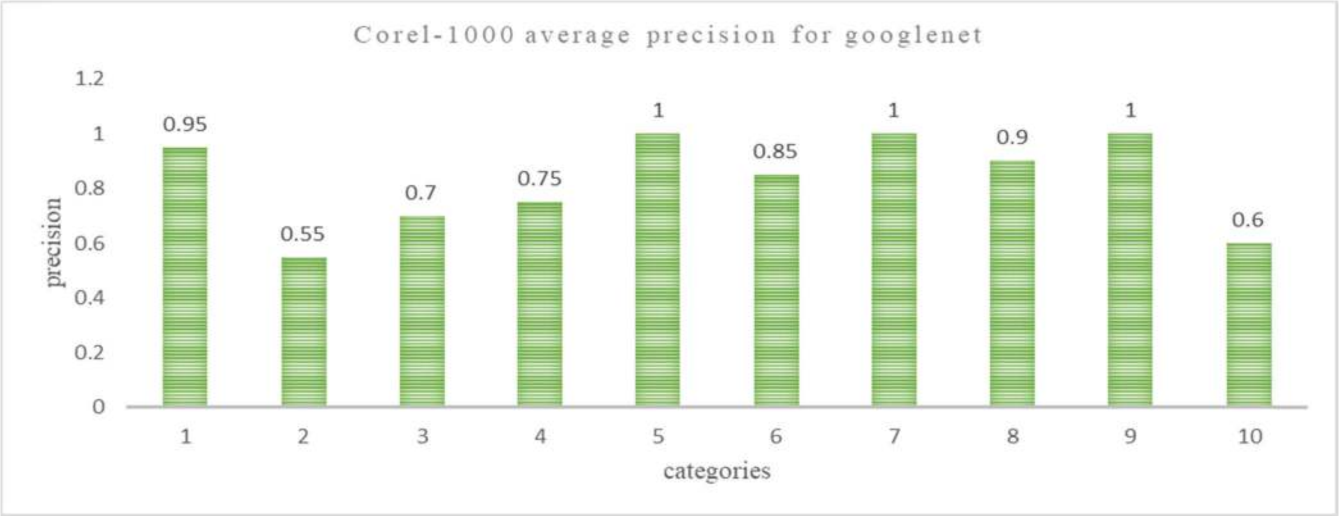

FIGURE 32. Graphical representation of corel-1000 average precision for VGG-19.

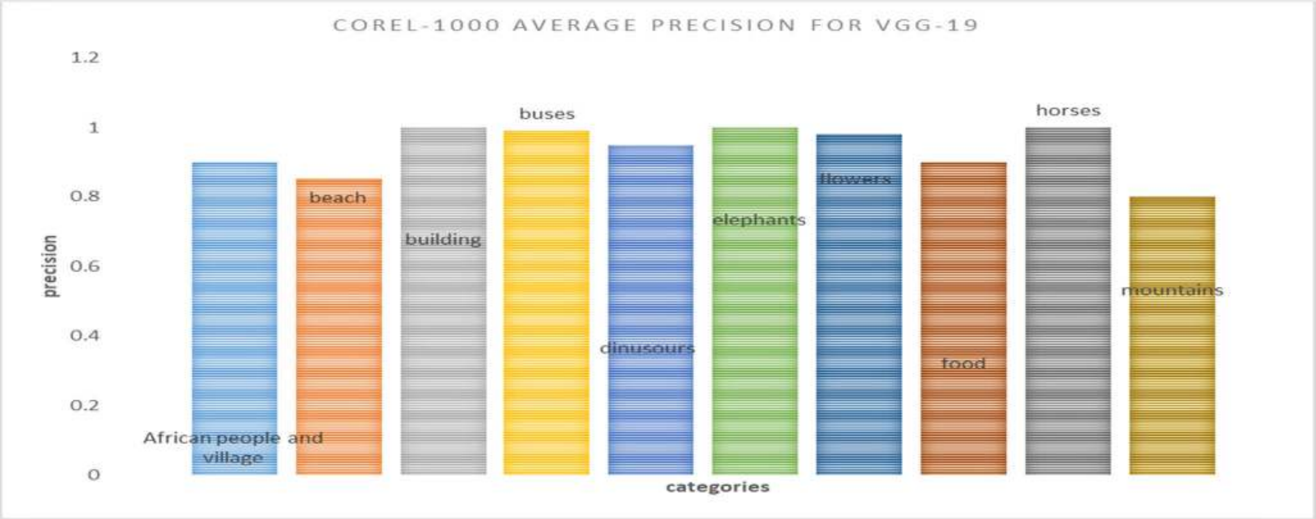

FIGURE 33. Graphical representation of corel-1000 average precision for VGG-19.

FIGURE 34. Different model images of Cifar 100 dataset showing from per capita category.

wings, main body, and tail but difference came colors, shape edge detection. The precision of airplanes-101 is highest while the lowest for a blimp. Because the shape of the blimp is slightly different from the airplanes so our proposed method differentiates accurately and outperforms very well. Results of our proposed model is demonstrated in tabular as well as in graphical form.

c: TYPE OF ANIMAL CATEGORIES

Categories that fall in animal type are 19 in the Caltech-256 dataset. Images are taken from worldwide common animals like Bear, chimp, dog, el, elephant-101, elk, giraffe, goat, gorilla, greyhound, horse, kangroo-101, leopards-101, llama-101, porcupine, raccoon, skunk, teddy-bear, and triceratops. As shown below one sample image taken from each category.

Animal body structure/texture and shape helps out our proposed system to calculate the average precision.

Raccoon has the highest average precision of 1 because its body is covered with black and white hairs which cause our model to differentiate it easily from remaining categories.

TABLE 14. Comparison between precision for VGG19 and GoogLeNet.

Category	Precision for VGG19	Precision for Googlenet
African people and village	0.9	0.95
Beach	0.85	0.55
Buildings	1	0.7
Buses	0.99	0.75
Dinosaurs	0.95	1
Elephants	1	0.85
Flowers	0.98	1
Food	0.9	0.9
Horses	1	1
Mountains	0.8	0.6

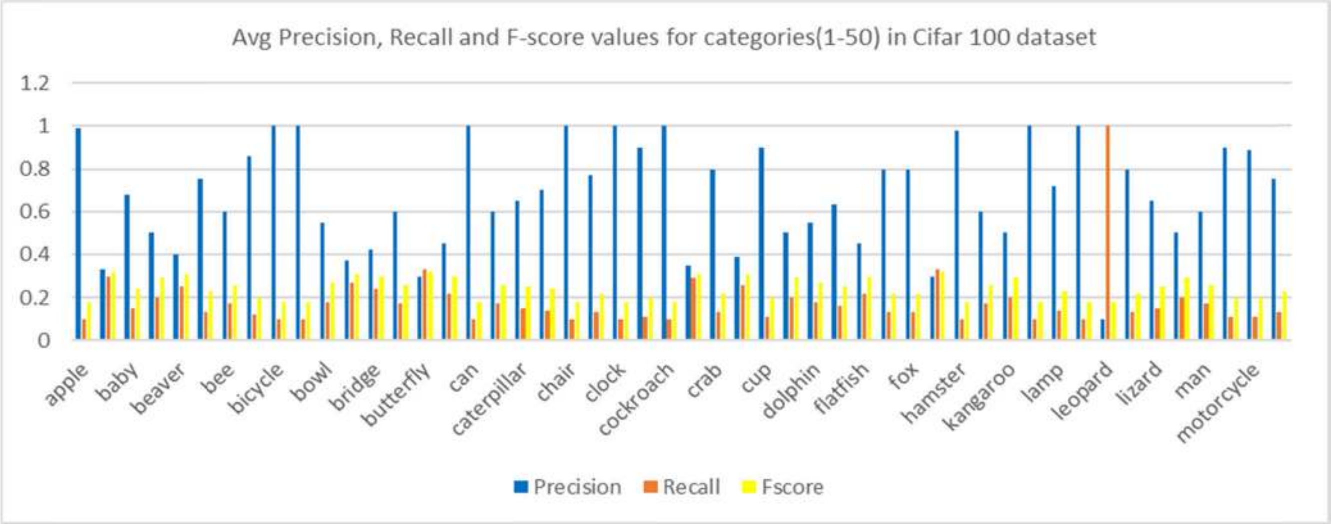

FIGURE 35. Avg precision, recall and F-score values for categories(1-50) in Cifar 100 datase.

However skunk, teddy-bear, and triceratops had the lowest value due to the fact that teddy bear is a toy, skunk has smaller legs and long heavy tail, and triceratops category is a type of dinosaur is the last known generation so it contain imaginary images, so our proposed model shows lowest value for it 0.1.

d: TYPES OF COMPUTER HARDWARE CATEGORIES

A computer consist upon multiple input and output devices common known as keyboard, mouse, cpu, speakers, headphones etc Caltech 256 also had some of the categories related to computer hardware some of them are computer types like laptop-101, ipod, palm-pilot, and some are computer components like computer-keyboard, computer-mouse, computer-monitor, joystick, video projector, pic-card, and vcr.

Pie chart best demonstrates the average precision values for computer hardware types laptop-101 and palm-pilot had the highest average precision because both contain the basic function of computer altogether. Whereas computer keyboard, computer monitor and computer-mouse had the lowest why? The reason is input devices, work on a specific task, while the auto computer device is itself capable for computatuion to produce the precise results according to the methodology.

Graphical representation of Precision, Recall, and F-score shows outstanding results. The recall had values nearer to 1, which indicates 100% results in most of the categories.

3) ALOT (250) DATASET RESULTS

ALOT dataset consists of 25,000 images divided into 250 categories, where each class contains 100 images. This dataset is useful for testing the texture competence of the projected descriptor. Same size of images 384 × 235 present in each category. The ALOT (250) targets image categorization and classification categorize texture images from different groups—furthermore, the number of categories in the territory of content-based image retrieval. ALOT database

TABLE 15. Precision, recall, and F-score value for each category in cifar 100 dataset.

Cifar 100									
No	**Category**	**Precision**	**Recall**	**F-score**	**No**	**Category**	**Precision**	**Recall**	**F-score**
1	apple	0.99	0.1	0.18	**51**	mouse	0.4	0.25	0.31
2	aquarium_fish	0.33	0.3	0.32	**52**	mushroom	0.64	0.16	0.25
3	baby	0.68	0.15	0.24	**53**	oak_tree	0.7	0.14	0.24
4	bear	0.5	0.2	0.29	**54**	orange	1	0.1	0.18
5	beaver	0.4	0.25	0.31	**55**	orchid	0.65	0.15	0.25
6	bed	0.75	0.13	0.23	**56**	otter	0.25	0.4	0.31
7	bee	0.6	0.17	0.26	**57**	palm_tree	0.42	0.24	0.3
8	beetle	0.86	0.12	0.2	**58**	pear	0.7	0.14	0.24
9	bicycle	1	0.1	0.18	**59**	pickup_truck	0.77	0.13	0.22
10	bottle	1	0.1	0.18	**60**	pine_tree	0.65	0.15	0.25
11	bowl	0.55	0.18	0.27	**61**	plain	1	0.1	0.18
12	boy	0.37	0.27	0.31	**62**	plate	0.9	0.11	0.2
13	bridge	0.42	0.24	0.3	**63**	poppy	0.7	0.14	0.24
14	bus	0.6	0.17	0.26	**64**	porcupine	0.83	0.12	0.21
15	butterfly	0.3	0.33	0.32	**65**	possum	0.5	0.2	0.29
16	camel	0.45	0.22	0.3	**66**	rabbit	0.5	0.2	0.29
17	can	1	0.1	0.18	**67**	raccoon	0.43	0.23	0.3
18	castle	0.6	0.17	0.26	**68**	ray	0.5	0.2	0.29
19	caterpillar	0.65	0.15	0.25	**69**	road	0.8	0.13	0.22
20	cattle	0.7	0.14	0.24	**70**	rocket	0.6	0.17	0.26
21	chair	1	0.1	0.18	**71**	rose	0.8	0.13	0.22
22	chimpanzee	0.77	0.13	0.22	**72**	sea	0.65	0.15	0.25
23	clock	1	0.1	0.18	**73**	seal	0.3	0.33	0.32
24	cloud	0.9	0.11	0.2	**74**	shark	0.69	0.14	0.24
25	cockroach	1	0.1	0.18	**75**	shrew	0.4	0.25	0.31
26	couch	0.35	0.29	0.31	**76**	skunk	0.69	0.14	0.24
27	crab	0.8	0.13	0.22	**77**	skyscraper	0.6	0.17	0.26
28	crocodile	0.39	0.26	0.31	**78**	snail	0.33	0.3	0.32
29	cup	0.9	0.11	0.2	**79**	snake	0.9	0.11	0.2
30	dinosaur	0.5	0.2	0.29	**80**	spider	0.8	0.13	0.22
31	dolphin	0.55	0.18	0.27	**81**	squirrel	0.32	0.31	0.32
32	elephant	0.63	0.16	0.25	**82**	streetcar	0.5	0.2	0.29
33	flatfish	0.45	0.22	0.3	**83**	sunflower	0.89	0.11	0.2
34	forest	0.8	0.13	0.22	**84**	sweet_pepper	0.8	0.13	0.22
35	fox	0.8	0.13	0.22	**85**	table	0.45	0.22	0.3
36	girl	0.3	0.33	0.32	**86**	tank	0.4	0.25	0.31
37	hamster	0.98	0.1	0.18	**87**	telephone	0.97	0.1	0.19
38	house	0.6	0.17	0.26	**88**	television	1	0.1	0.18
39	kangaroo	0.5	0.2	0.29	**89**	tiger	0.38	0.26	0.31

TABLE 15. ***(Continued.)*** **Precision, recall, and F-score value for each category in cifar 100 dataset.**

40	keyboard	1	0.1	0.18	**90**	tractor	0.65	0.15	0.25
41	lamp	0.72	0.14	0.23	**91**	train	0.4	0.25	0.31
42	lawn_mower	1	0.1	0.18	**92**	trout	0.9	0.11	0.2
43	leopard	0.1	1	0.18	**93**	tulip	0.45	0.22	0.3
44	lion	0.8	0.13	0.22	**94**	turtle	0.45	0.22	0.3
45	lizard	0.65	0.15	0.25	**95**	wardrobe	1	0.1	0.18
46	lobster	0.5	0.2	0.29	**96**	whale	0.52	0.19	0.28
47	man	0.6	0.17	0.26	**97**	willow_tree	0.74	0.14	0.23
48	maple_tree	0.9	0.11	0.2	**98**	wolf	0.93	0.11	0.19
49	motorcycle	0.89	0.11	0.2	**99**	woman	0.45	0.22	0.3
50	mountain	0.75	0.13	0.23	**100**	worm	0.86	0.12	0.2

consisting of 250 types, it is used to test the efficiency and flexibility. The various groups that we made in the ALOT dataset for precise discussion of results include grains, vegetables, weaving baskets, spices, sponge, pasta, sands, fruits, lego, seashells, cooked food, dried grass, fabrics, bubbles, embossed fabrics, viscous liquid, small repeated patterns and so on. These groups pay to diverse two-dimensional information, objects key points, and texture information to catalog images. Efficiently the projected method ordered the texture images from analogous groups with similar forefront and back objects. It is divided into similar categories into one common type so that the proposed approach remarkable results can be explained precisely for images with similar textures. The images are effectively classified using CCN features with cascaded samples, texture patterns, and object recognition based filtering by the proposed method. In the ALOT dataset, the greater part of the classes contains texture pictures with comparative examples and hues, though different sorts incorporate distinctive object designs. The offered strategy demonstrates imperative outcomes, with up to 80% accuracy rates in most baffling classes.

a: FABRIC TEXTURE

ALOT (250) dataset contains many categories regarding the fabric. It is shown that there are multiple types of fabrics available in the world, so do in our dataset like emboss, strips, cotton, stripe pattern, color pattern, etc. our proposed method gives remarkable results regarding fabric type.

b: WEAVING TEXTURE

Handmade weaving pattern has its texture, practices, and thread type; where all types are discussed separately in fabric class, although it is also a piece of fabric. The precision results for this type is remarkable for our model.

c: FRUIT TEXTURE

Categories falling in fruit texture contain the texture of both dry fruits and fresh fruits and peel skin and pulp of fresh fruits. The precision results for our model is 90% for most of the categories. Categories that are slightly different from others have fewer precision values.

Precision values of all the categories falling in fruit texture are demonstrated in tabular as well as in graphical form below. The remarkable performance is achieved in different categories which are otherwise very difficult to categorized.

d: LEAF TEXTURE

Leaf texture categories contain dried leaf, grass, and leaf pattern. Some of these categories consist of dried leaves while others contain fresh leaves. So differences in color and texture affect the average precision values. The green color is the prominent color of the leaf so categories that show more single green color shade had given more precise values. Besides this fact, our model gave 90% results mostly due to the emphasis on a similar texture.

e: SPONGE TEXTURE

A sponge is a soft substance that is full of holes and can absorb a lot of liquid. The sponge had a different texture from other categories in ALOT (250). Due to its unique texture and color, all the categories showing such types of images put all together in the sponge texture group/type. Some categories images showpieces of sponges while others show repeated texture and color. The first mentioned categories result in low average precision value while others show nearest to the highest. Results are demonstrated in tabular and graphical form for average precision values. Our proposed model shows above 90% precision results for maximum categories, which made it unique from another state of the art models.

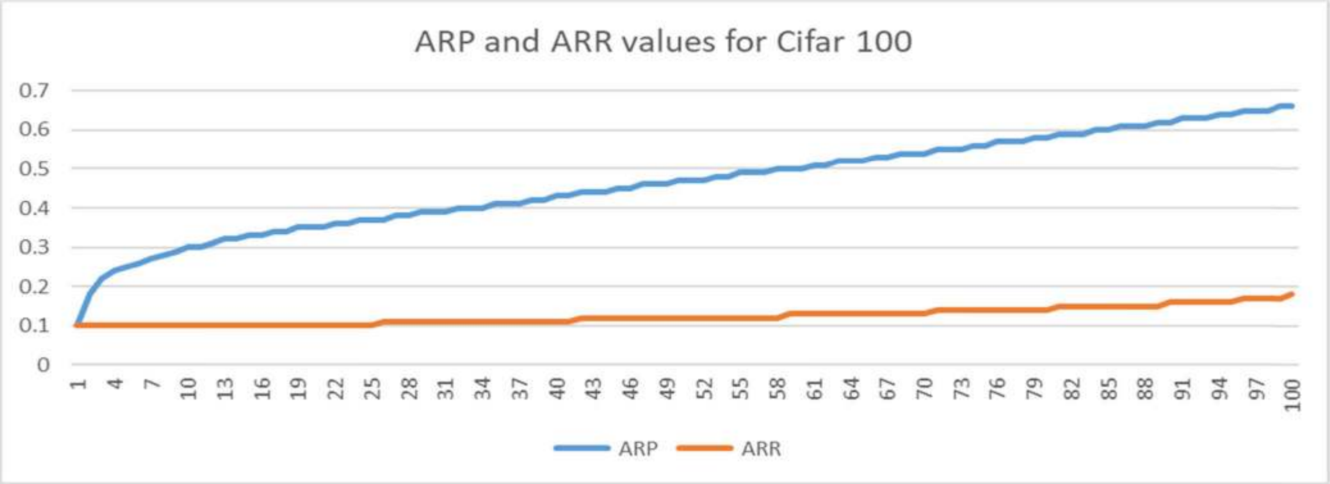

FIGURE 36. ARP and ARR values for Cifar 100.

FIGURE 37. Different model images of Cifar 100 dataset showing from per capita category.

f: SEASHELL

A seashell is a firm, defensive outside layer shaped by an animal existing in the sea. The shelter is part of the body of the animal. Unfilled seashells are frequently found pounded up on beaches by beachcombers.

g: LEGO/TABLETS

LEGO blocks are vibrant plastic that can be linked together speedily to style a tower, car, and more. LEGO bricks are combined by studs on the upper, and holes in the bottom of the brick normally recognized as the brick-and-knob assembly. LEGO is the widely held building toy famous all over the world.

When we see the values for average Precision, Recall, and F-score of the whole dataset it shows remarkable results. Precision describes the ratio between predicted corrected vs total correct observations our model shows average precision for the ALOT dataset is above 90% in most of the categories.

Recall demonstrating the proportion between the correctly predicted positive interpretations overall observations in genuine class. F-measure is commonly known as the weighted average of Precision and Recall. F1 Score is the weighted normal of Precision and Recall. Consequently, this score considers both False positives and False negatives. Naturally, it isn't as straightforward as precision, yet F1 is generally more helpful than exactness, particularly if you have a lopsided class appropriation.

Precision works best if False positives and False negatives have a comparative expense. On the off chance that the expense of False positives and False negatives are different, it's smarter to take a gander at both Precision and Recall.

Average retrieval precision and Average retrieval Recall which shows the average value for each category.

4) COREL-1000 DATASET RESULTS

The Corel-1000 dataset is utilized for image characterization and recovery normally. Corel datasets contain different image classes covering plain foundation images of multifaceted items. The dataset consists of one thousand images in ten classes. The Corel-1k (1000) dataset covers various groups such as flowers, food, animals, natural scenes, buildings, mountains, buses, and people. For the object detection and versatility of the image semantics, Corel-1k (1000) is tested. For each group, 100 images had a tenacity of 384 × 256 pixels or 256 × 384 pixels for every group.

The Corel-1k (1000) dataset proficiently categorized images owing to the DL feature of the advocated methods. Cascading samples, texture patterns, object key point's detection filtering, and RGB channels with GoogLeNet and VGG-19 made it conceivable to efficiently categorize the images. The precision results for the Corel-1k (1000) dataset display the superiority of the projected technique. The significant performance showed by the proposed method in most of the categories, such as beaches, buildings, buses, dinosaurs, flowers, mountains, horses, and food. For complex types including dinosaurs, flowers, and horses, the proposed technique stated 0.95, 0.98 and 1 precision rates respectively. The categories of buses and mountains revealed 0.99 and 0.8 precision rate, respectively. Other categories showed above 0.74 precision rates.

The precision results are tested for both nets VGG-19 and GoogLeNet and the final results are shown in tabular as well as in graphical form.

5) CIFAR-100 DATASET RESULTS

Cifar 100 dataset consists of one hundred different categories of image size 32 × 32 in each category. The size of the images in this dataset is much smaller as well as images

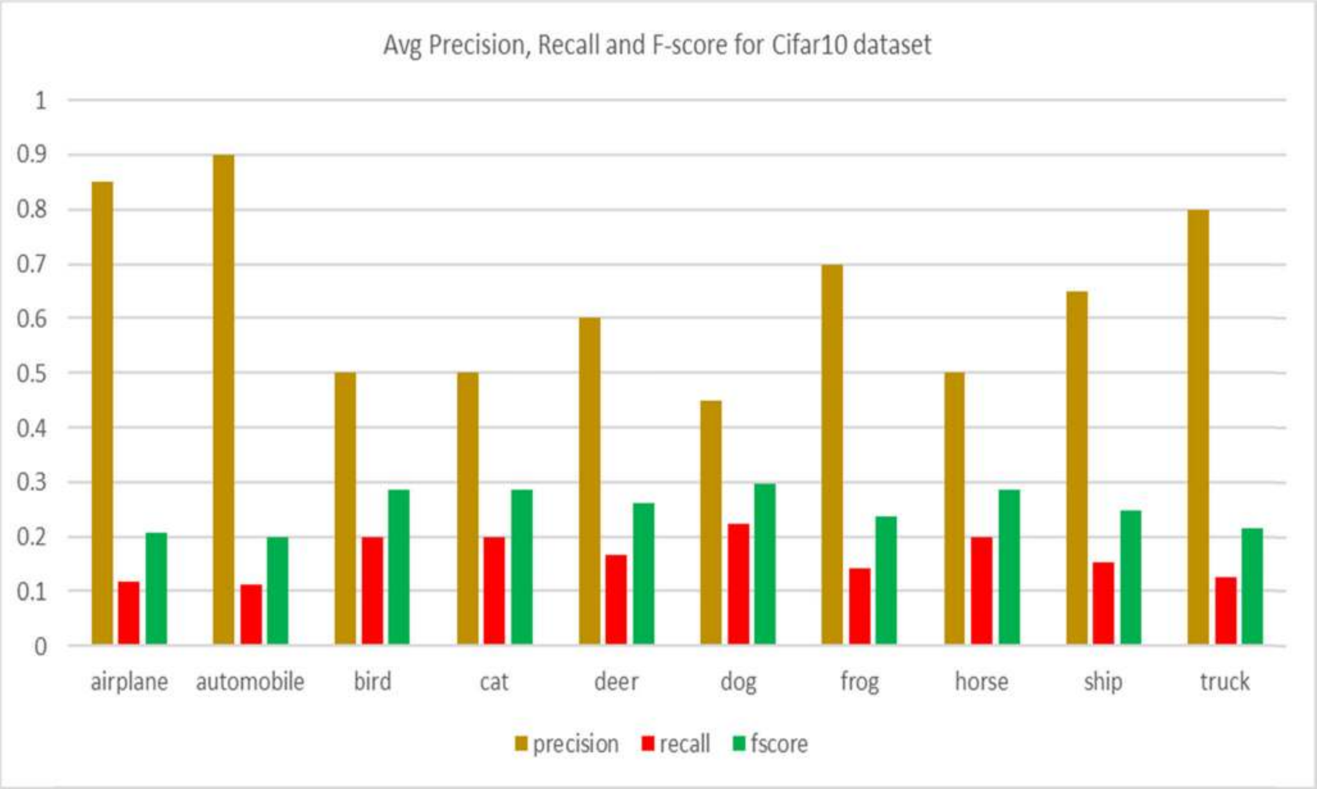

FIGURE 38. Avg precision, recall, and F-score for Cifar10 dataset.

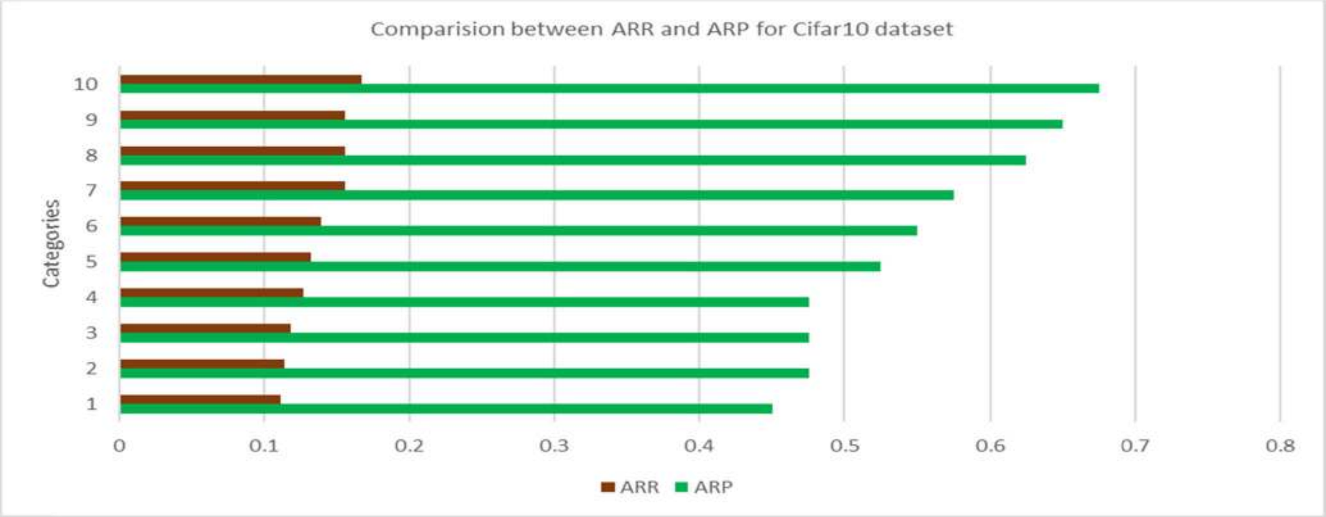

FIGURE 39. ARP and ARR values for Cifar10 dataset.

are blurred these factors affect the image retrieval efficiency. Although it is difficult to categories these images due to previously mentioned problems still proposed model works out efficiently than other present models. Categories like white or single-color background show good retrieval results because objects place on the background recognized more clearly. Categories like apple, bottle, bicycle, can, chair, etc show good average precision results due to their vibrant color foreground objects and single color background. Whereas categories like butterfly, girl, seal, snail, etc did not show good average precision results as compared to others. The reason for slightly lower results due to the complexity in object detection, blurry edges which cause difficulty in differentiating between foreground and background objects. Our model

TABLE 16. Average precision, recall and F-score for cifar10 dataset.

No.	category	precision	Recall	F-score
1	airplane	0.85	0.12	0.21
2	automobile	0.9	0.11	0.2
3	bird	0.5	0.2	0.29
4	cat	0.5	0.2	0.29
5	deer	0.6	0.17	0.26
6	dog	0.45	0.22	0.3
7	frog	0.7	0.14	0.24
8	horse	0.5	0.2	0.29
9	ship	0.65	0.15	0.25
10	truck	0.8	0.13	0.22

TABLE 17. Computation time.

No. of images	Features extraction time (sec.)	Computation Time Total time (sec.)
Corel-1000	0.219	2.55
Cifar-10	0.189	2.07
Cifar-100	0.197	2.64
Caltech-256	0.287	3.08
ALOT	0.141	1.95
FTVL	0.186	2.01

shows a good result of precision, recall, and fscore for this dataset. Most of the categories show a 99% average precision results. Recall values go for categories from 1 to 0.1 which make our model high achiever than others.

The quality metrics of this large dataset for all categories are shown in tabular form.Graphical representation of ARR and ARP values for Cifar 100 dataset is as follows:

6) CIFAR 10 DATASET RESULTS

Cifar 10 dataset consist of ten categories named as airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. Each category is different from remaning. Each category contain 1000 images with dimension of 32 × 32 similar as Cifar 100 dataset. ARP and ARR results shows the efficiency of our proposed model. ARP shows almost 70% result for each category while ARR shows near about 10% results for each category.

Average precision value is highest for automobile and lowest for dog. Recall value highest for dog category while lowest for automobile. F-score ranges from 0.2 to 0.3.

V. CONCLUSION

The typical image content analysis solutions depend upon the classification & retrieval techniques that is strengthened from the revealed image contents. It is inevitable to achieve high level metrics at primitive and deep learning level by thorough image analysis, synthesis. This contribution innovatively combines texture, shape, and color features with the involvement of VGG-19 and GoogLeNet architectures to address the key challenges of image retrieval in the world of large datasets. Large datasets including Caltech-256, ALOT (250), Cifar 100, Cifar 10 and Corel-1000 are efficiently experimented to check the effectiveness of the proposed model in AP, AR ARP, and ARR metrics. The proposed model is capable to detect, describe, recognize and bind the image signatures correctly reflecting the real image contents with category distinction for almost similar image semantic groups.

REFERENCES

[1] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: Fast retina keypoint," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 510–517.

[2] R. R. Saritha, V. Paul, and P. G. Kumar, "Content based image retrieval using deep learning process," *Cluster Comput.*, vol. 22, no. S2, pp. 4187–4200, Mar. 2019.

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

[4] A. Dhillon and G. K. Verma, "Convolutional neural network: A review of models, methodologies and applications to object detection," *Prog. Artif. Intell.*, vol. 9, no. 2, pp. 85–112, 2020.

[5] J. Singh, A. Bajaj, A. Mittal, A. Khanna, and R. Karwayun, "Content based image retrieval using Gabor filters and color coherence vector," in *Proc. IEEE 8th Int. Advance Comput. Conf. (IACC)*, Dec. 2018, pp. 290–295.

[6] P. Narloch, A. Hassanat, A. S. Tarawneh, H. Anysz, J. Kotowski, and K. Almohammadi, "Predicting compressive strength of cement-stabilized rammed earth based on SEM images using computer vision and deep learning," *Appl. Sci.*, vol. 9, no. 23, p. 5131, Nov. 2019.

[7] S. Singh and S. Batra, "An efficient bi-layer content based image retrieval system," *Multimedia Tools Appl.*, vol. 79, pp. 1–29, Feb. 2020.

[8] R. Ashraf, M. Ahmed, U. Ahmad, M. A. Habib, S. Jabbar, and K. Naseer, "MDCBIR-MF: Multimedia data for content-based image retrieval by using multiple features," *Multimedia Tools Appl.*, vol. 79, nos. 13–14, pp. 8553–8579, Apr. 2020.

[9] M. Gao, H. Chen, S. Zheng, and B. Fang, "Feature fusion and non-negative matrix factorization based active contours for texture segmentation," *Signal Process.*, vol. 159, pp. 104–118, Jun. 2019.

[10] A. Shakarami and H. Tarrah, "An efficient image descriptor for image classification and CBIR," *Optik*, vol. 214, Jul. 2020, Art. no. 164833.

[11] R. Pires de Lima and K. Marfurt, "Convolutional neural network for remote-sensing scene classification: Transfer learning analysis," *Remote Sens.*, vol. 12, no. 1, p. 86, Dec. 2019.

[12] Y. Liu, W. Chen, H. Qu, S. M. H. Mahmud, and K. Miao, "Spatial division networks for weakly supervised detection," *Neural Comput. Appl.*, pp. 1–14, Aug. 2020.

[13] W. W. Y. Ng, J. Li, X. Tian, H. Wang, S. Kwong, and J. Wallace, "Multilevel supervised hashing with deep features for efficient image retrieval," *Neurocomputing*, vol. 399, pp. 171–182, Jul. 2020.

[14] M. Mateen, J. Wen, S. Song, and Z. Huang, "Fundus image classification using VGG-19 architecture with PCA and SVD," *Symmetry*, vol. 11, no. 1, p. 1, Dec. 2018.

[15] X. Liu, S. Zhang, T. Huang, and Q. Tian, "E2BoWs: An end-to-end bag-of-words model via deep convolutional neural network for image retrieval," *Neurocomputing*, vol. 395, pp. 188–198, Jun. 2020.

[16] J. Yin, X. Liu, J. Yang, C.-Y. Chu, and Y.-L. Chang, "PolSAR image classification based on statistical distribution and MRF," *Remote Sens.*, vol. 12, no. 6, p. 1027, Mar. 2020.

[17] K. T. Ahmed, S. Ummesafi, and A. Iqbal, "Content based image retrieval using image features information fusion," *Inf. Fusion*, vol. 51, pp. 76–99, Nov. 2019.

[18] R. Scherer and S. Ditzinger, *Computer Vision Methods for Fast Image Classification and Retrieval*. Springer, 2020.

[19] D. Jatmiko and S. Prini, "Study and performance evaluation binary robust invariant scalable keypoints (BRISK) for underwater image stitching," in *Proc. IOP Conf., Mater. Sci. Eng.* Bristol, U.K.: IOP Publishing, 2020.

[20] K. Kanwal, K. T. Ahmad, R. Khan, A. T. Abbasi, and J. Li, "Deep learning using symmetry, FAST scores, shape-based filtering and spatial mapping integrated with CNN for large scale image retrieval," *Symmetry*, vol. 12, no. 4, p. 612, Apr. 2020.

[21] M. Rashid, M. A. Khan, M. Alhaisoni, S.-H. Wang, S. R. Naqvi, A. Rehman, and T. Saba, "A sustainable deep learning framework for object recognition using multi-layers deep features fusion and selection," *Sustainability*, vol. 12, no. 12, p. 5037, Jun. 2020.

[22] Y. Ge, S. Jiang, Q. Xu, C. Jiang, and F. Ye, "Exploiting representations from pre-trained convolutional neural networks for high-resolution remote sensing image retrieval," *Multimedia Tools Appl.*, vol. 77, no. 13, pp. 17489–17515, Jul. 2018.

KHAWAJA TEHSEEN AHMED received the M.S. degree in computer science from Bahauddin Zakariya University, Multan, Pakistan, in 2010, and the Ph.D. degree in computer science from the University of Central Punjab (UCP), Lahore, Pakistan, in 2019. He is currently working as an Assistant Professor with the Department of Computer Science, Bahauddin Zakariya University. His research interests include computer vision, deep learning, pattern recognition, and machine learning.

SAROOSH JAFFAR received the B.S. degree in telecommunication from Bahauddin Zakariya University, Multan, Pakistan, in 2014, where she is currently pursuing the M.S. degree in information technology. Her research interests include computer vision, deep learning, pattern recognition, image enhancement, and machine learning.

MALIK GHULAM HUSSAIN received the M.S. degree in computer engineering. He is currently pursuing the Ph.D. degree in computer science with Bahauddin Zakariya University, Multan. He is also working as an Assistant Professor with the Department of Computer Science, Bahauddin Zakariya University. His field of specialization is code optimization.

SHAHID FAREED received the M.S.C.S. degree in software engineering and the Ph.D. degree in computer sciences. He is currently working as an Assistant Professor with the Department of Computer Science, Bahauddin Zakariya University, Multan. His field of specialization is software engineering.

ARIF MEHMOOD received the Ph.D. degree from the Department of Information and Communication Engineering, Yeungnam University, South Korea, in November 2017. He is currently working as an Assistant Professor with the Department of Computer Science and IT, The Islamia University of Bahawalpur, Pakistan. His research interests include data mining, mainly working on AI and deep Learning-based text mining, and data science management technologies.

GYU SANG CHOI received the Ph.D. degree from the Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA, USA, in 2005. From 2006 to 2009, he was a Research Staff Member with the Samsung Advanced Institute of Technology (SAIT), Samsung Electronics. Since 2009, he has been a Faculty Member with the Department of Information and Communication, Yeungnam University, South Korea. His research interests include non-volatile memory and storage systems.