Turkish Journal of Computer and Mathematics Education Vol.12 No.13(2021), 2499-2510

Research Article

Image Retrieval Using Hierarchical Nested Clusters

[1] S.Banuchitra, [2] Dr. K.Kungumaraj

[1]Research Scholar in Computer Science, Mother Teresa Women's University, Kodaikanal.

[2]Assistant Professor, PG Department of Computer Science, Arulmigu Palaniandavar Arts College For Women, Palani.

Article History: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 4 June 2021

ABSTRACT: Digital image collections are rapidly being created and made available to multitudes of users through the World Wide Web, how to retrieve the image quickly and find the desired information from the massive data becomes a big issue. And now a days, Content-based image retrieval is used for regular process of retrieving images according to image visual contents as a replacement for textual annotations. Image retrieval can be used to retrieve similar images, and the effect of image retrieval depends on the selection of image features to a certain extent. Based on recent successes of deep learning techniques especially Convolutional Neural Networks (CNN) in solving computer vision applications, to extract more conducive to the high-level semantic feature of image retrieval using convolution neural network. Deep learning methodology combined with distance-based learning and Gaussian kernel features can be seen as recursive supervised algorithm to create new features, and hence used to provide optimal feature space for any classification method. Implementation of RSBL used in this paper is based on Euclidean distance and Gaussian kernel features.

Keywords: Neural Network, Digital Image retrieval, Hierarchical clustering.

1. INTRODUCTION

A content-based image retrieval (CBIR) system works on the low-level visual features of a user input query image, which makes it difficult for the users to formulate the query and also does not give satisfactory retrieval results. In the past image annotation was proposed as the best possible system for CBIR which works on the principle of automatically assigning keywords to images that help image retrieval users to query images based on these keywords. Image annotation is often regarded as the problem of image classification where the images are represented by some low-level features and the mapping between low-level features and high-level concepts (class labels) is done by some supervised learning algorithms. In a CBIR system learning of effective feature representations and similarity measures is very important for the retrieval performance. Semantic gap has been the key challenge in the past for this problem. A semantic gap exists between low-level image pixels captured by machines and the high-level semantics perceived by humans. Machine learning has been exploited to bridge this gap in the long term. The recent successes of deep learning techniques especially Convolutional Neural Networks (CNN) in solving computer vision applications has inspired me to work on this thesis so as to solve the problem of CBIR using a dataset of annotated images.

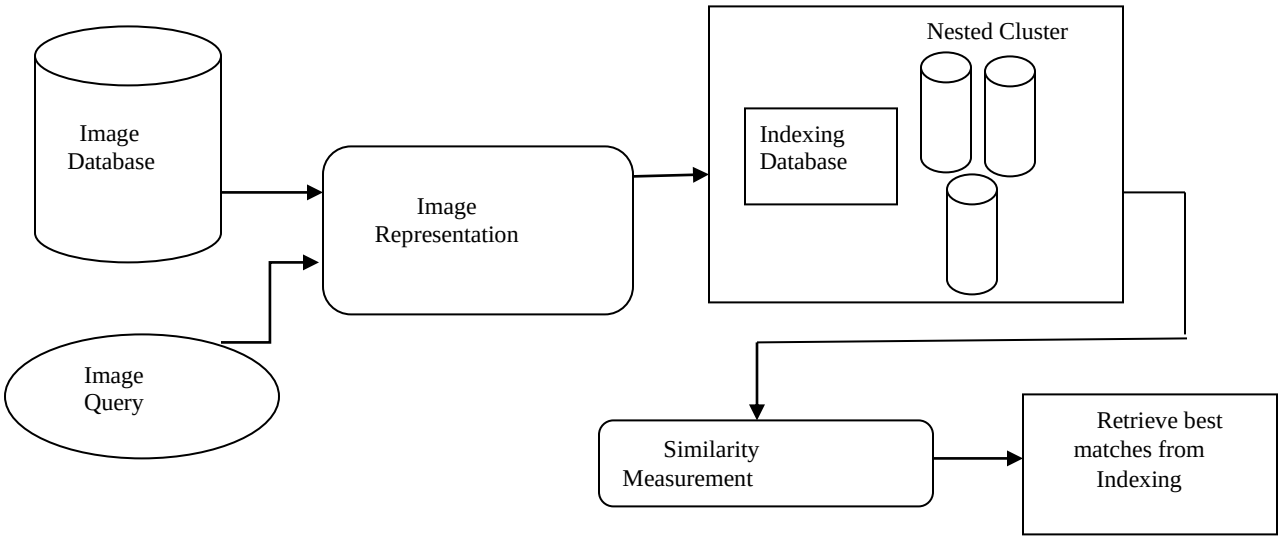

Figure-1: CBIR system using Hierarchical Nested Cluster

1.1 Hierarchical Nested Clustering

Hierarchically nested data clusters are structured in which data clusters at higher layers represent one or multiple clusters at a lower layer based on mean values of the cluster centers. The first layer clusters are generated based on feature representations derived from the CNN model. Data clusters are formed by grouping the relevant data points using a partition-based clustering approach known as K-means clustering

A new concept of CBIR is employed to exploit the opportunities presented by large image-based repositories, particularly in Convolutional Neural Network. The proposed approach, which relies solely on the contents of the images, will pave the way for a computationally efficient and real-time image querying through an unstructured image database. An end-to-end CBIR framework is conducted without supervision. First, we utilize a deep CNN model as a feature extractor to obtain the feature representations from the activations of the Convolutional layers. In the next step, a hierarchically nested database indexing structure and local recursive density estimation are developed to facilitate an efficient and fast retrieval process.

2. PROBLEM DEFINITION

CBIR techniques are beginning to find a toehold in many applications, such as biology, remote sensing, satellite imaging, etc., the technology still suffers from lack of maturity due to a significant gap towards semantic-aware retrieval from visual content. A major challenge associated with CBIR systems is to extract information from an image which is unique and representative, to overcome the issue of so called semantic-gap. The semantic-gap refers to low-level features of images such as colors and texture, but those features might not be able to extract a higher level of understanding of the image perceived by humans. Due to the absence of solid evidence on the effectiveness of CBIR techniques for high-throughput datasets with varied collections of images, opinion is still sharply divided regarding the reliability and performance of such systems in real-time. It is essential to standardize CBIR for easy access to data and speed up the retrieval process.

The proposed approach, which relies solely on the contents of the images, will pave the way for a computationally efficient and real-time image querying through an unstructured image database. An end-to-end CBIR framework is conducted without supervision. First, we utilize a deep CNN model as a feature extractor to obtain the feature representations from the activations of the convolutional layers. In the next step, a hierarchically nested database indexing structure and local recursive density estimation are developed to facilitate an efficient and fast retrieval process. Finally, the key elements of CBIR, accuracy and computational efficiency, are evaluated and compared with the state-of-the-art CBIR techniques.

2.1 Framework

In this research study, the following are includes in our research framework,

- Images are presented in a convolutional way by using AlexNet
- Images are indexing in Hierarchical Nested Cluster
- Images are retrieved in Similarity measures technique by using RSBL

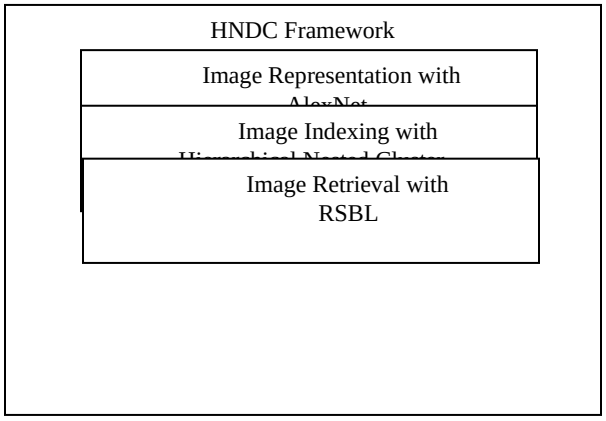

Figure.2 Framework for CBIR with Hierarchical Nested Cluster

3. CLUSTERING IN CBIR

Content-based image retrieval (CBIR) refers to the automatic process of retrieving images from a large image database that have similar visual content to a query image. The process normally involves extracting the visual content from database images when they are first loaded, extracting the same kind of visual content from the query image, comparing it against those of the stored images, and returning a list of ranked images according to some measurement of degree of similarity. Therefore, effectively extracting the right visual contents that are relevant to the user's query interest, meaningful measurement of degree of similarity upon the extracted visual content and organizing an efficient search structure for retrieval of relevant images from a large database are naturally the main areas of interest. Clustering is a process of grouping data objects in a given data set according to their similarities. Clustering solutions have been studied for more than five decades in fields such as statistical analysis, machine learning and data mining. As a result, many algorithms of different categories have been developed and extensively used in various application areas including CBIR. Broadly, clustering algorithms can be categorized into either hierarchical or partition based where a hierarchical algorithm produces a hierarchy of possible groupings at different levels of data granularity, and a partition-based algorithm produces only one version of grouping, i.e. a partitioning of the data into a number of disjoint groups. More specifically, from understanding of the result clusters, clustering algorithms can be divided into prototype-based, model-based, density-based, or graph-based solutions.

A prototype-based algorithm initially divides data objects into imprecise prototype clusters and then iteratively refines the prototypes into final clusters. A model based algorithm attempts to model the data objects with multivariate distributions, and the final best fit distributions, i.e. a statistical model, are then taken as clusters.

A density-based algorithm seeks to find dense regions where there is a greater concentration of data objects and hence the regions form clusters of similar data objects. A graph-based method treats a data set as a weighted connected graph where vertices represent the individual data objects and edges signify pair-wise similarities between the data objects, and clustering becomes a graph partition problem where separated subgraphs of closely inter-connected vertices become clusters. All clustering algorithms have their strengths and limitations. In CBIR, the prototype-based K-means algorithm is frequently used despite its limitations in forming only convex shaped clusters and vulnerability towards noise data. Very little work has been done on using other types of algorithms. We are interested in any effect of the different types of clustering algorithms in forming homogeneous regions, and hence have decided to take one algorithm of each category and test their performances. A clustering method is then applied to group the local features into homogenous segments where the descriptors or moments of each of the segments like the mean vector, variance and segment size are then used to represent the visual content of the entire image. A similarity function is then applied to two sets of descriptors of the query image and the stored

image to measure and select images of similar contents Depending on different local features used, different clustering methods used, and the similarity function deployed, the retrieved results can vary significantly, posing a bigger question regarding the effectiveness of the methods chosen for the performance at various stages of the retrieval process.

4. RESEARCH METHODOLOGY

4.1 Image Representations

In content based image retrieval, the key problem is how to efficiently measure the similarity between images. Since the visual objects or scenes may undergo various changes or transformations, it is infeasible to directly compare images at pixel level. Usually, visual features are extracted from images and subsequently transformed into a fix-sized vector for image representation. Considering the contradiction between large scale image database and the requirement for efficient query response, it is necessary to pack the visual features to facilitate the following indexing and image comparison. To achieve this goal, quantizations with visual codebook training are used as a routine encoding processing for feature aggregation/pooling. Besides, as an important characteristic for visual data, spatial context is demonstrated vital to improve the distinctiveness of visual representation. Based on the above discussion, we can mathematically formulate the content similarity between two images X and Y as follow,

$$S(x, y) = \sum_{x \varepsilon X} \sum_{y \varepsilon Y} k(x, y)$$

$$= \sum_{x \varepsilon X} \sum_{y \varepsilon Y} \Phi(x)^T \Phi(y)$$

$$= \Psi(X)^T \ \Psi(Y)$$

In the recent past the advancement in computer and multimedia technologies has led to the production of digital images and cheap large image repositories. The size of image collections has increased rapidly due to this, including digital libraries, medical images etc. To tackle this rapid growth it is required to develop an image retrieval system which operates on a large scale. The primary aim is to build a robust system that creates, manages and query image databases in an accurate manner. CBIR is the procedure of automatically indexing images by the extraction of their low-level visual features, like shape, color, and texture, and these indexed features are solely responsible for the retrieval of images. Thus, it can be said that through navigation, browsing, query-by-example etc. we can calculate the similarity between the low-level image contents which can be used for the retrieval of relevant images. Images are a representation of points in a high dimensional feature space and a metric is used to measure the similarity or dissimilarity between images on this space. Therefore, those images which are closer to the query image are similar to it and are retrieved. Feature representation and similarity measurement are very crucial for the retrieval performance of a CBIR system and for decades researchers have studied them extensively. A variety of techniques have been proposed but even then it remains as one of the most challenging problems in the ongoing CBIR research, and the main reason for it is the semantic gap issue that exists between the low-level image pixels captured by machines and high level semantic concepts perceived by humans. Such a problem poses fundamental challenge of Artificial Intelligence from a high-level perspective that is how to build and train 2 intelligent machines like human to tackle real-world tasks. One promising technique is Machine Learning that attempts to address this challenge in the long-term. In the recent years there have been important advancements in machine learning techniques. Deep Learning is an important breakthrough technique, which includes a family of machine learning algorithms that attempt to model high-level abstractions in data by employing deep architectures composed of multiple non-linear transformations. Deep learning impersonates the human brain that is organized in a deep architecture and processes information through multiple stages of transformation and representation, unlike conventional machine learning methods that are often using shallow architectures. By exploring deep architectures to learn features at multiple levels of abstracts from data automatically, deep learning methods allow a system to learn complex functions that directly map raw sensory input data to the output, without relying on human-crafted features using domain knowledge. In the recent studies encouraging results have been reported for applying deep learning techniques in applications like image retrieval, natural language processing, object recognition among

others. The success of deep learning inspired me to explore deep learning techniques with application to CBIR task for annotated images. There is limited amount of attention focusing on CBIR applications even though there has been much research attention of applying deep learning for image classification and recognition in computer vision.

The CIFAR-10 and CIFAR-100 are labelled subsets of the 80 million tiny images dataset. They were collected by Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The CIFAR-10 dataset consists of 30000 32x32 colour images in 10 classes, with 6000 images per class. There are 20000 training images and 10000 test images. The dataset is divided into two training batches and one test batch, each with 10000 images. The test batch contains exactly 1000 randomly-selected images from each class. The training batches contain the remaining images in random order, but some training batches may contain more images from one class than another. Between them, the training batches contain exactly 10000 images from each class.

AlexNet is the name of a convolutional neural network which has had a large impact on the field of machine learning, specifically in the application of deep learning to machine vision. It famously won the 2012 ImageNet LSVRC-2012 competition by a large margin (15.3% VS 26.2% (second place) error rates). It consisted of 11×11, 5×5,3×3, convolutions, max pooling, dropout, data augmentation, ReLU activations, SGD with momentum. It attached ReLU activations after every convolutional and fully-connected layer. AlexNet was trained for 6 days simultaneously on two Nvidia Geforce GTX 580 GPUs which is the reason for why their network is split into two pipelines. The architecture of AlexNet is shown in Figure-2 The first convolutional layer performs convolution and max pooling with Local Response Normalization (LRN) where 96 different receptive filters are used that are 11×11 in size. The max pooling operations are performed with 3×3 filters with a stride size of 2. The same operations are performed in the second layer with 5×5 filters. 3×3 filters are used in the third, fourth, and fifth convolutional layers with 384, 384, and 296 feature maps respectively. Two fully connected (FC) layers are used with dropout followed by a Softmax layer at the end. Two networks with similar structure and the same number of feature maps are trained in parallel for this model. Two new concepts, Local Response Normalization (LRN) and dropout, are introduced in this network. LRN can be applied in two different ways: first applying on single channel or feature maps, where an N×N patch is selected from same feature map and normalized based on the neighborhood values. Second, LRN can be applied across the channels or feature maps (neighborhood along the third dimension but a single pixel or location). AlexNet has 3 convolution layers and 2 fully connected layers. When processing the ImageNet dataset, the total number of parameters for AlexNet can be calculated as follows for the first layer: input samples are 224×224×3, filters (kernels or masks) or a receptive field that has a size 11, the stride is 4, and the output of the first convolution layer is 55×55×96.

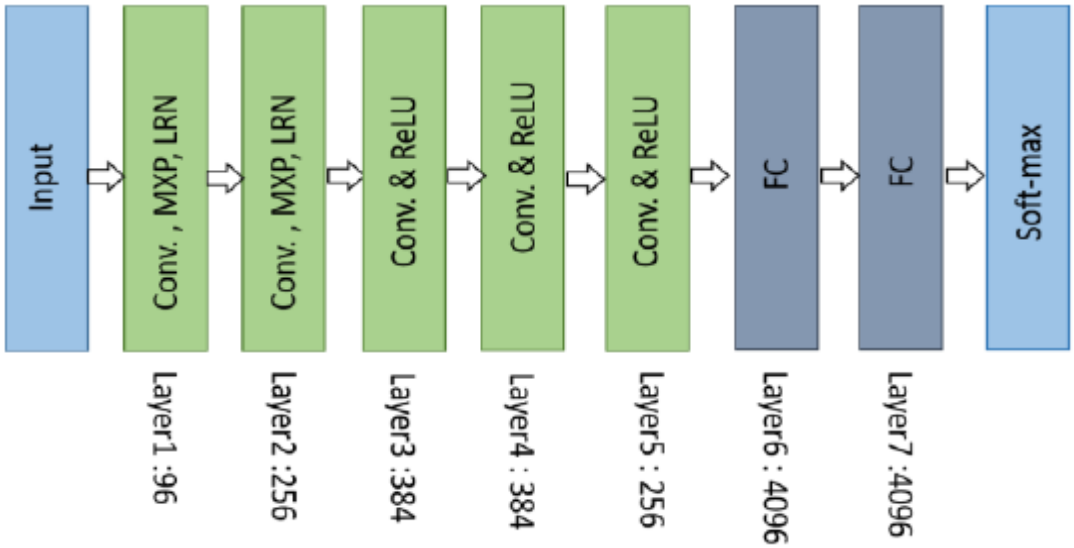

Figure-2: Architecture of AlexNet

The architecture depicted in Figure-2, the AlexNet contains eight layers with weights; the first five are convolutional and the remaining three are fully connected. The output of the last fully-connected layer is fed to a 1000-way softmax which produces a distribution over the 1000 class labels. The network maximizes the multinomial logistic regression objective, which is equivalent to maximizing the average across training cases of the log-probability of the correct label under the prediction distribution. The kernels of the second, fourth, and fifth convolutional layers are connected only to those kernel maps in the previous layer which reside on the same GPU. The kernels of the third convolutional layer are connected to all kernel maps in the second layer. The neurons in the fully-connected layers are connected to all neurons in the previous layer.

AlexNet takes 90 epochs which were trained for 6 days simultaneously on two Nvidia Geforce GTX 580 GPUs which is the reason for why their network is split into two pipelines. SGD with learning rate 0.01, momentum 0.9 and weight decay 0.0005 is used. Learning rate is divided by 10 once the accuracy plateaus. The learning rate is decreased 3 times during the training process.

5. IMAGE INDEXING USING HIERARCHICAL NESTED CLUSTER

In an attempt to address the challenges faced by retrieval information on a large-scale dataset, we present a hierarchically nested structure. The introduced database was indexing aims at arranging and structuring the image database into a simple yet effective form of data clusters and hierarchies. Although forming a hierarchical structure for retrieval optimization has been explored before, the method presented in this study is quite different. In math, the relation between a parent node u and a child node v in the hierarchy is called partial order relation, defined as $v \approx u$. In the application of taxonomy, for concept u and v, $v \approx u$ means every instance of category v is also an instance of category u, but not vice versa. We call such partial order on probability densities as the notion of nested. Let g and f be the densities of u and v respectively, if $v \approx u$, then $f \approx g$, i.e. f is nested in g. quantitatively measuring the loss violate the nested relation between f and g is not easy. According to the definition of partial order, strictly measuring that can be done as:

$$\{x : f(x) > \eta\} - \{x : g(x) > \eta\}$$

Where $\{x:f(x) > \eta\}$ is the set where f is greater than a nonnegative threshold η. Threshold η indicates the nested degree required by us. Small value of η means high requirement for the overlap between f and g to satisfy $f \approx g$. Above Eqn. describes how many regions with densities greater than η of f are not nested in those of g.

Hierarchically nested data clusters are structured in which data clusters at higher layers represent one or multiple clusters at a lower layer based on mean values of the cluster centers. The first layer clusters are generated based on feature presentations derived from the CNN model. Data clusters are formed by grouping the relevant data points using a cluster centre point approach. μ and X are abstract values and denote mean values and scalar products.

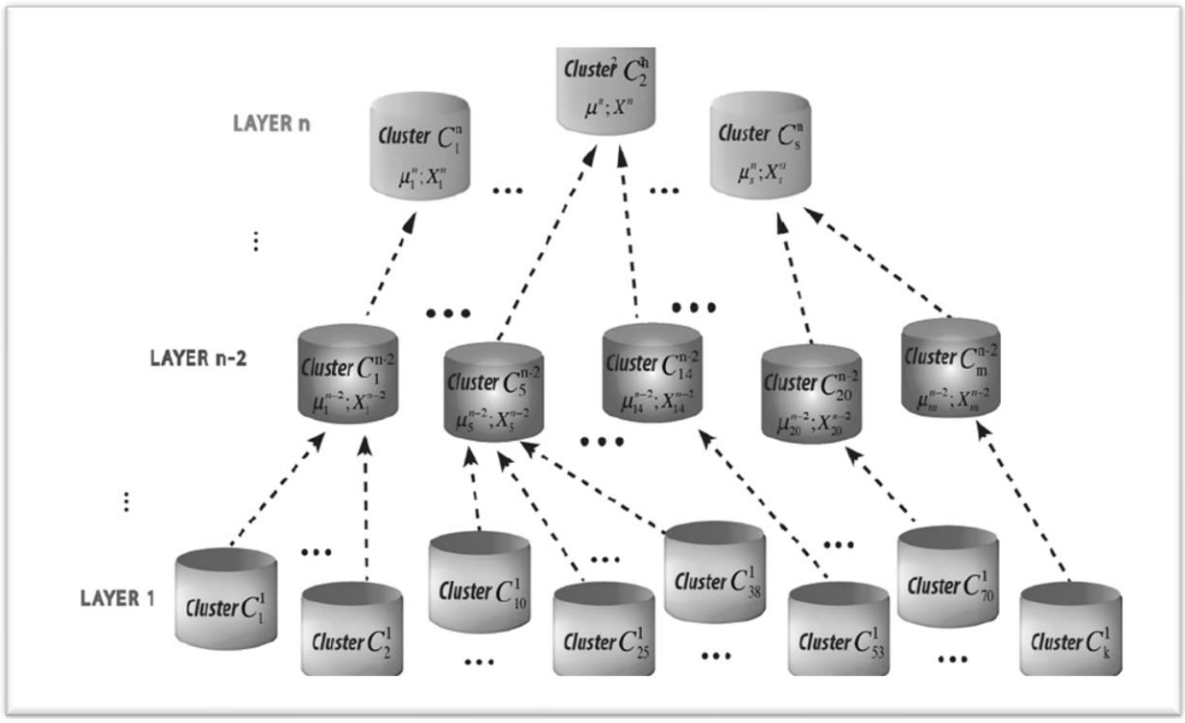

Figure-3: Hierarchical Nested Cluster

6. RECURSIVE SIMILARITY-BASED LEARNING (RSBL)

Deep learning methodology combined with distance-based learning and Gaussian kernel features can be seen as recursive supervised algorithm to create new features, and hence used to provide optimal feature space for any classification method. Implementation of RSBL used in this paper is based on Euclidean distance and Gaussian kernel features with fixed σ=0.1, providing new feature spaces at each depth level. The Algorithm sketched below presents steps of the RSBL; in each case parameters k_{max} = 20 and σ = 5 were used. In essence the RSBL algorithm at each level of depth transforms the actual feature space into the extended feature space $H^{(m)}$, discovering useful information by creating new redundant features. Note that the initial space covers d original features xj that are available at each depth, preserving useful information. The final analysis in the H(α) space (and optimization of parameters at each level of RSBL algorithm, including feature selection) may be done by various machine learning methods. The emphasis is on generation of new features using deep-learning methodology rather than optimization of learning. RSBL may be presented as a constructive algorithm, with new layers representing transformations and procedures to extract and add to the overall pool more features, and a final layer analyzing the image of data in the enhanced feature space.

Recursive similarity-based learning Algorithm

1: Standardize the dataset, n vectors, d features.
2: Set the initial space $H^{(0)}$ using input features x_{ij} , i = 1::n vectors and j = 1::d features.
3: Set the current number of features d(0) = d.
4: for m = 1 to α do
5: for k = 1 to k_{max} do
6: For every training vector x_i find k nearest neighbors $x_{j;i}$ in the $H^{(m-1)}$ space.
7: Create nk new kernel features $z_{j;i}(x) = K(x; x_{j;i})$; j = 1::k; i = 1::n for all vectors using kernel functions as new features.
8: Add new nk features to the $H^{(m-1)}$ space, creating temporary $H^{(m;k)}$ space.
9: Estimate error E(m, k) in the $H^{(m;k)}$ space on the training or validation set.
10: end for
11: Choose k' that minimizes E(m; k') error and retain $H^{(m,k')}$ space as the new $H^{(m)}$ space.
12: end for

13: Build the final model in the enhanced feature space $H^{(\alpha)}$.
14: Classify test data mapped into the enhanced space.

The final step after forming the hierarchically nested data clusters is to find the cluster which contains the most similar images to a query image. We applied recursive similarity to measure a similarity between the query image and all images inside each cluster recursively. The main idea of the recursive similarity function is to estimate the probability function by a Cauchy type kernel and to recursively calculate it. The method is also applied for novelty detection in real time data streams and video analytics. The recursive calculation allows us to discard each data once it has been processed and only store the accumulated information in memory concerning the local mean (per cluster), μ and scalar product X. In order to speed up the retrieval process by an order of magnitude, the searching process is performed from the top of the pyramid in an ordered hierarchy based on "winner takes all" principle with maximum local recursive density estimation at each level. The degree of similarity between the query images to images inside each cluster is measured by the relative local density with regards to the query image, which is defined by a suitable kernel over the distance between the current image sample and all the other images inside the cluster.

7. RESULTS AND DISCUSSION

The proposed approach was tested with a dataset of 30,000 images collected within the CIFAR database. Once the winning cluster at the lowest layers (layer 1) is selected, the images inside the cluster is re-arranged based on their similarity to the query image using Self Organization Map. Small distance implies that the corresponding image is more similar to the query image and vice versa. The execution time of the proposed approach with hierarchy of nested clusters was tested on several randomly selected queries and compared with 1) no clustering or hierarchical structure and direct comparison of the query image with each of 30,000 images 2) plain clustering with no hierarchical nested structure. The comparison starts at the top layer first and continuous at the lower layers; however, only with the clusters correspond with the winner cluster determined at the layer above.

Methods	Execution Time (s)
Proposed method of Clustering Nested Dynamic Clusters	0.0015
No Clustering with image sets	0.205
Non-hierarchical clustering with image sets	0.019

Table-1: Execution time compare with different clusters setup

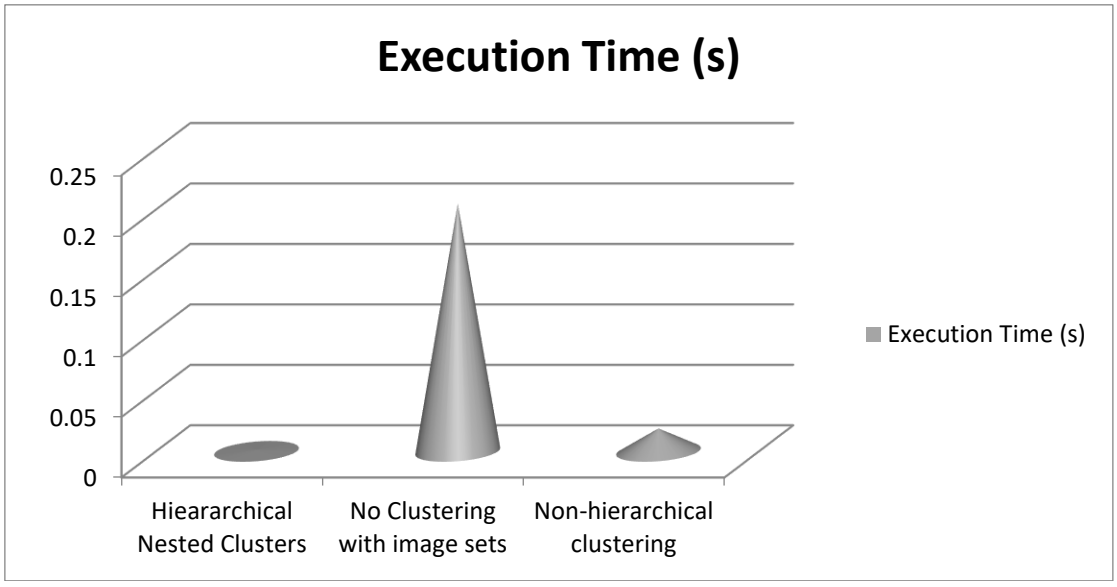

Figure-4: Execution time compare with different clusters setup

In this experiment, the hierarchical system is made in two layers; however, the approach is scalable; thus, more layers can be integrated if necessary. Nested Clustering was used to form the Hierarchical nested clusters. At the lower layer all 30,000 images were grouped into 10 major clusters while at the higher layer it reduced to 30 clusters. In this experiment, the extracted features from Convolutional Neural Network(CNN) are used to form the clusters and to assign the indices of the particular images. The 10,000 images were trained by CNN for identifying their statistical features. Agglomerative algorithm was applied to the mean values of extracted features of the images. Dendrogram is the graphical representation of the result of clustering. Dendrogram of the major cluster (Image classes) are represented in Figure-5.

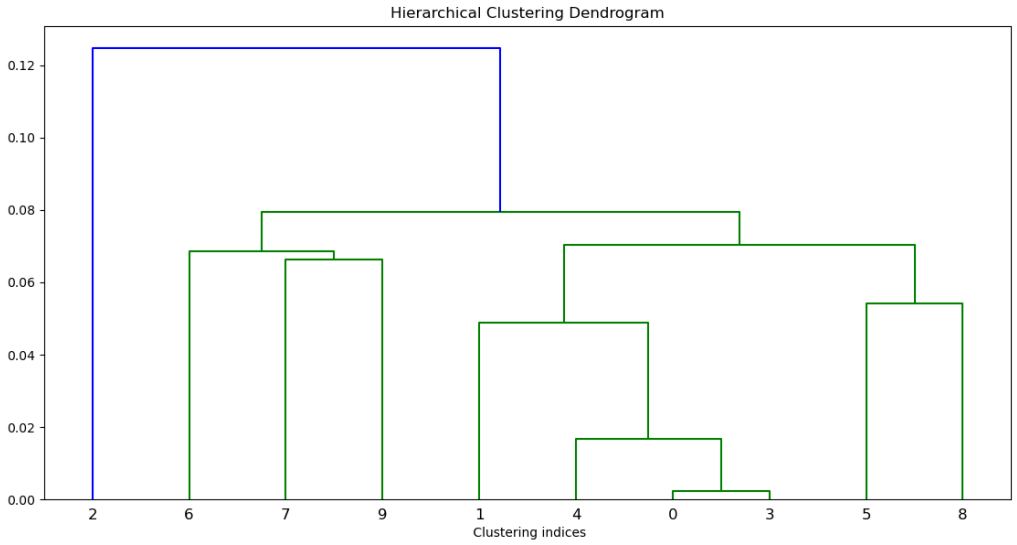

Figure- 5: Dendrogram of the major classes (0-9)

Here, the cluster indices are assigned from 0 to 9 for representing the classes Airplane, Auto mobile, Bird, Cat, Deer, Dog, Frog, Horse, Ship and Truck respectively. Hierarchical clustering method produces the nested clusters as dendrogram. In this chapter, Agglomerative algorithm is applied to the each image class for forming the nested clusters. Nested clusters are formed as per the similarity measurements. Clusters of clusters (Nested clusters) leads to get the more relevant images when compared with hierarchical clustering without nesting. Based on Agglomerative algorithm, each image class is treated as singleton clusters. The distance matrices are computed by calculating the similarity among the data elements using Euclidean distance. Calculated distance matrices are depicted in Tale 5.2 and 5.3. Closest pairs of data elements are formed as a cluster. The process of clustering is stopped when there is no data element to split into cluster. For nested clusters, each image class is divided into three sub clusters for easy accessing. Like this, all the classes are divided into three sub clusters and new indices are assigned. The indices of nested clusters are also start at 0 and are stored in the databases for the fast retrieval. The dendrogram of nested clusters of image classes are depicted in Figure-6.

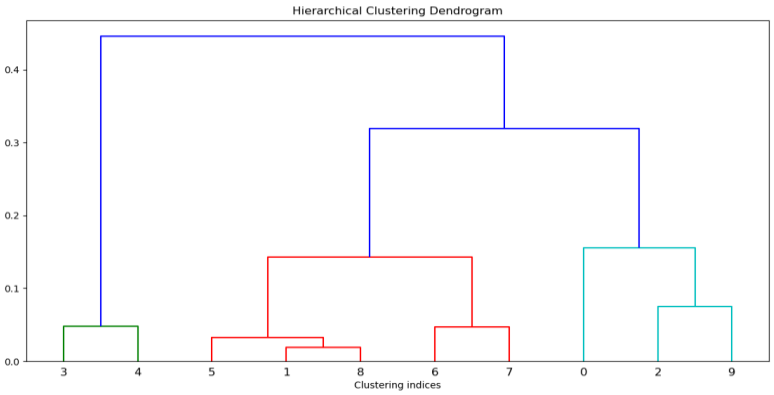

Figure-6: Dendrogram of nested clusters of Truck class

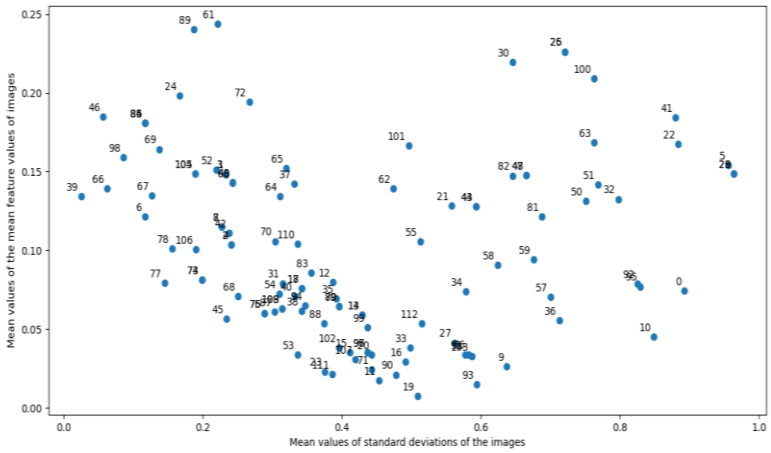

Figure-7: Input values

The 100 images from each class is taken as input for the clusters. Dendrogram of the nested clusters denotes nested cluster mean as labeled as 0 to 9. The proposed method is also tested with 113 of new images. The dendrogram and clusters are represented in Figure-8 and Figure-9.

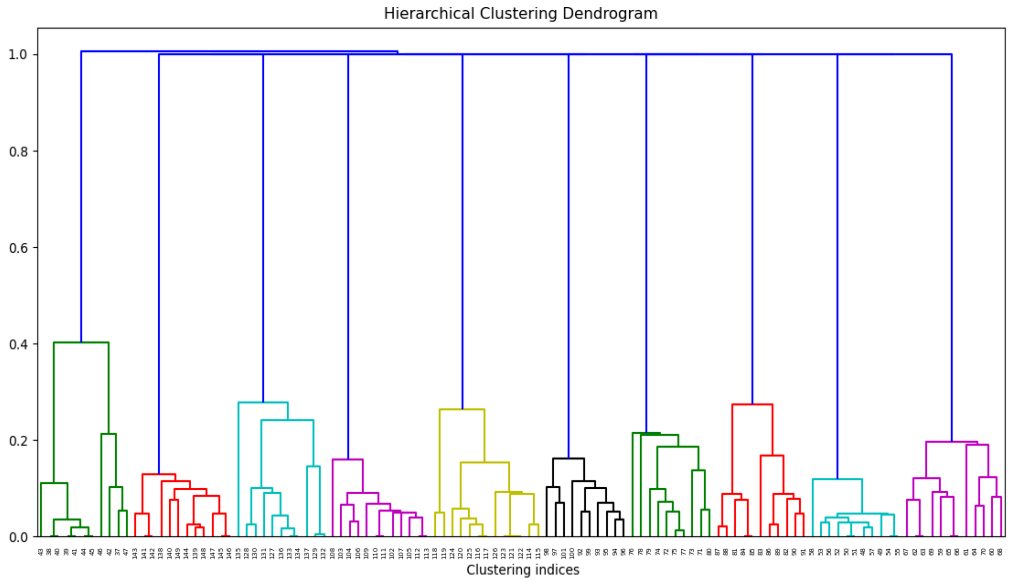

Figure-8: Dendrogram

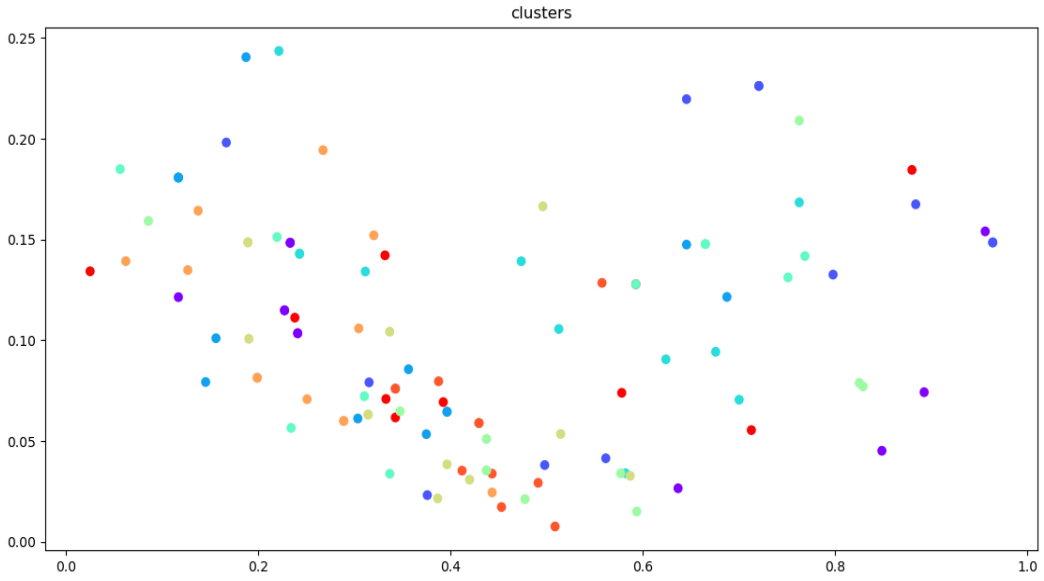

Figure-9: Major Clusters (10 classes)

The query image is feed into system then the system calculates its feature values and compare that computed values with the mean of the cluster. If it matches the corresponding images in that clusters are retrieved this has the highest similarity scores. The similarity score of the query images are calculated by using Euclidean equations as per the RSBL algorithm. RSBL algorithm was implemented by python and it takes less time to retrieve the image which has high similarity score. The kernel features were also extracted from the CNN and were used by the RSBL algorithm. The distance matrices were created by taking the difference between the kernel features and the features from the input images.

8. CONCLUSION

This algorithm is a new fast approach for organization of Hierarchical with Nested Cluster and search within CBIR context has more accurate. Its main idea is to organize cluster with multi-dimensional data in which evolving hierarchically nested clusters structure using a combined multiple sets of features and a computationally efficient with Recursive similarity measure. The approach was tested on a data base which contains 10,000 images from about 10 different genres/rubrics. The proposed algorithm was able to automatically form 9417 visually very relevant results within few milliseconds making only about simple calculations. The obtained accuracy is **94.17%**. The approach is scalable and parallelizable in nature. It can be realized as a web service. It is also possible to include user feedback in a future application.

REFERENCES

[1] Tomasz Maszczyk and Włodzisław Duch Recursive Similarity-Based Algorithm for Deep Learning, Lecture Notes in Computer Science 7665, 390-397. Springer, Heidelberg, 2012

[2] Christian Bongiorno, Salvatore Miccich, and Rosario N. Mantegna Nested partitions from hierarchical clustering statistical validation, arXiv:1906.06908v1 [q-bio.GN] 17 Jun 2019

[3] Plamen Angelov and Pouria Sadeghi-Tehran, Procedia Computer Science INNS Conference on Big Data , Volume 53, 2015, Pages 1–8

[4] Chien-Hao Kuo1, Yang-Ho Chou, and Pao-Chi Chang; Using Deep Convolutional Neural Networks for Image Retrieval, Society for Imaging Science and Technology, 2016.2.VIPC-231

[5] Hailong Liu, Baoan Li, Xueqiang Lv and Yue Huang Image Retrieval Algorithm Based on Convolutional Neural Network, Advances in Intelligent Systems Research, volume 133, 2016

[6] Hongbo Du, Hanan Al-Jubouri Effectiveness of Image Features and Similarity Measures in Cluster-based Approaches for Content-based Image Retrieval, The International Society for Optical Engineering · May 2014

[7] ImageNet Classification with Deep Convolutional Neural Networks by Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, 2012

[8] Angelov, P.; Sadeghi-Tehran, P.; Ramezani, R. A Real-time Approach to Autonomous Novelty Detection and Object Tracking in Video Stream. Int. J. Intell. Syst. 2011, 26, 189–205.

[9] Angelov, P. Evolving Rule-Based Models: A Tool for Design of Flexible Adaptive Systems; Springer: Berlin/Heidelberg, Germany, 2002.