Cosine Similarity Based Hierarchical Skeleton and Cross Indexing for Large Scale Image Retrieval Using Mapreduce Framework

Zhong Qianwen

Electronic and Communication Engineering, Shangdong University, Shandong, Weihai, Huancui District (China)

Received 9 September 2020 | Accepted 23 September 2021 | Published 31 January 2023

Abstract

The imaging data in various fields like industries, institutions, medical, and so on has grown exponentially in recent years. An innovative software solution is required for the efficient management of image data. The MapReduce framework is used for large-scale image data processing. Various cross-indexing techniques are developed to transform the image into binary sequences but retrieving the image from the reducer on the feature vector results in a major challenge. Image retrieval using large-scale image databases attained major attention, where cross-indexing plays a key role in the research community. Therefore, in this research, a new method for image retrieval, named Cosine Similarity-based hierarchical skeleton and cross-indexing, is proposed to perform the retrieval process in the MapReduce framework effectively. The feature vector of the images is converted to binary sequences. The Most Significant Bit (MSB) of the binary code is used to store the images in the mapper using the cross-indexing model. The image retrieval process is achieved through the reducer based on the tanimoto similarity measure. The binary sequence for the query image is calculated based on the feature vector. The MSB bit of the binary code is matched with the MSB code of the images in the mapper to achieve the retrieval process. The proposed method effectively achieved better performance through the cross-indexing model with the usage of the feature vector. The performance of the proposed method is compared with the existing techniques using the UK bench dataset. The proposed method attains the values of 0.784, 0.729, 0.75, 31.23, 17.84secfor F1-score, precision, recall, computational cost, and computational time with the query set-1 by considering four mappers.

Keywords

Cosine Similarity, Cross Indexing, Hierarchical Skeleton, Image Retrieval, Tanimoto Similarity Measure.

DOI: 10.9781/ijimai.2023.01.008

I. Introduction

In the multimedia environment, the large-scale information processing system attracted significant attention in the research area due to the growth of information, such as videos and images [1]. However, to search an image in the huge volume of the dataset with high accuracy based on the semantic features has become a significant problem. A single image can convey more details than the usage of the number of words, such that image retrieval is considered an important topic in the research area for the past few decades [2]. One of the most challenging issues is helping the users retrieve their estimated images from the huge database [3]. Image retrieval is the process of finding the suitable image with the appropriate content or feature based on the image set or image content description. Due to the issues in human cognition and the subjectivity of image content, defining the universal and efficient image retrieval process becomes a challenging task. Therefore, the image retrieval process has become an important area in the information retrieval and the computer vision field in recent years [4]. Two different stages are used: cognitive load-based complexity and the second complexity classification to address the complexity of imager retrieval [5]. Moreover, generating digital images is rapidly increased in the advancement of network and multimedia technology. Due to the huge amount of data information, retrieving it securely and rapidly requires considerable effort [6], [7]. Secure big data transmission should require efficient sharing in all environmental conditions [8].

Due to the enormous amount of digitized image creation, the image retrieval process has become a complex issue because encryption is required while transmitting the image [9], [10]. The image sharing between different entities is done by encrypting the image [11].The Content-based image retrieval (CBIR) model is developed to perform the retrieval process, which effectively retrieves the suitable images according to the low level features, like texture, shape, and color [12]. The main aim of the CBIR is to retrieve the images from the large database with the most related visual information. It is required to analyze the image content to perform the content-based retrieval. Hence, similarity measurement and the feature representation factor are crucial elements in CBIR, but there is a challenging issue in CBIR called semantic gap [1]. CBIR is effectively used to manage the huge

* Corresponding author.

E-mail address: zhongqianwen77@gmail.com

volume of image database [13]; hence it acts as a possible solution in image retrieval. Furthermore, quick access to the large database needs an efficient and effective computing model. Therefore, the Hadoop framework is considered a suitable distributed computing scheme in the retrieval process based on the MapReduce framework. The MapReduce framework is widely used as the parallel device in the computing environment that processes the data based on peta byte and terabyte scales. Facebook, Google, and Amazon are the largest users in the MapReduce computing model, allowing the processes of the data in a distributed manner with intensive computing over different machines [14]. Due to the presence of semantic gap, the accuracy of the CBIR model was not adequate, such that the gap exists between the visual low level features, like colors and textures, and the high level image concepts are usually used by the user in the searching process [12], [15].Compressing the image before the transmission will help improve the accuracy of the image [16].

Due to the enormous growth in large-scale data technology, the application of image processing shows the characteristic features of large-scale processing technology in the image retrieval model. In general, some of the recent works are developed to acquire the fast searching time based on the large scale processing technology, such that the authors in [2] and [17] revealed to enhance the CBIR model in Hadoop. In [4], a new method is developed that focuses on the enhanced parallel K-Means algorithm based on the MapReduce scheme [18]. However, Artificial intelligence (AI) is considered a machine learning technique that attracted significant attention in the past decades [1], [19], [20]. The AI model's key objective is to allow computers to handle real-world tasks and simulate human intelligence. Minimizing the semantic gap in the CBIR model is essential in image retrieval. Some efforts are taken to reduce the gap in the machine learning methods. Especially, the deep learning methods attained significant progress in the recent years in image retrieval [21], namely deep brief network [22], deep Boltzmann machine [23], deep neural network [24], Region-based Convolutional Neural Network (R-CNN) [25], and so on. Among these methods, the deep convolutional neural network (DCNN) attained a great achievement in computer vision, like image classification, object recognition [26], and image segmentation [1], [27].

This research focuses on a new method for image retrieval based on the proposed CS-based hierarchical skeleton and cross-indexing. The features extracted using the Speeded-Up Robust Features (SURF) and CS-based hierarchical skeleton are converted to the binary sequences to perform the retrieval process effectively. The performance of the proposed method is increased with the representation of binary sequences rather than computing it through the decimal value. The cross-indexing increases retrieval performance by storing the image in the mappers using the feature vector. The feature vector enables the binarization of features to enhance the efficiency of the image retrieval process. The images retrieved from the mappers are responded to the user through the reducer in the MapReduce framework.

The contribution of the paper is image retrieval using MapReduce Framework with CS-based hierarchical skeleton and cross-indexing. The CS-based hierarchical skeleton and cross-indexing effectively enable the retrieval process by achieving better performance through the cross-indexing model. The conversion of features from the decimal to the binary sequences increases the performance of feature vector, which effectively makes the cross indexing model store the images in the mappers. The tanimoto similarity measure provides robust and accurate results in the perspective of retrieving the images.

The paper is organized as follows: Section II describes the motivation of the image retrieval model. Section III elaborates the proposed CS-based hierarchical skeleton and cross-indexing for image retrieval using the MapReduce framework. Section IV describes the results and discussion of the proposed image retrieval method, and finally, section V concludes the paper.

II. Motivation

In this section, the image retrieval method's motivation is discussed using various existing retrieval techniques, which motivates the researchers to develop a new model based on the CS-based hierarchical skeleton and cross-indexing.

A. Literature Survey

Various existing image retrieval methods are surveyed in this section: Bai, C *et al.* [1] developed a DCNN model for retrieving the large-scale images. It used the features of convolutional layers and attained a better extraction process. It was highly suitable for a large volume image database for mapping the high-dimensional feature vectors into the binary codes. The performance of the online retrieval processing time was very less. Sakr, N.A *et al.* [2] introduced a Chain Clustering Binary search tree (CC-BST) algorithm to model the visual statements to represent the features of an image. It was an effective solution in the retrieval mechanism for high-dimensional features. It offered a significant enhancement in the time cost factor than other competitive systems. However, it utilized more time to perform the retrieval process. Gao, X *et al.* [3] introduced a progressive image retrieval model to guarantee image quality. This model was parallelized using the MapReduce framework and attained enhanced performance with the MapReduce framework. It failed to use the adaptive learning methods. Cao, J *et al.* [4] modeled a parallel *k*-Means algorithm for selecting the cluster center in the image retrieval process. It maximized retrieval accuracy and reduced the overhead of retrieval time. To expand the node number in the Hadoop distributed platform, and the adjustment of relevant parameters used for retrieval in the system was not considered.

Xibing, S *et al.* [28] developed a MapReduce-based remote sensing image retrieval method to increase the efficiency of the retrieval process. It accurately retrieved the remote sensing image and attained better retrieval efficiency and accuracy. However, the node's information processing capacity was not effectively used to enhance the data efficiency at each node. Meng, Z [29] developed a remote sensing image retrieval algorithm. It effectively extracted the texture and the color features. It increased the retrieval accuracy and the efficiency of image retrieval. It failed to balance the system load. Mezzoudj, S *et al.* [18] developed a Content-Based Image Retrieval System using Spark (CBIR-S). It attained better performance with the spark. It failed to use the efficient model in terms of accuracy with larger datasets and clusters. Li, X *et al.* [30] introduced a parallel data processing model. It attained better performance in the single node system at both the capability and the retrieval speed of dealing with the huge volume of data. The accuracy of the searching process was effective only for few features.

B. Challenges

Some of the challenges associated with the image retrieval methods are explained as follows:

- Due to the various manners of viewing the image among the computer and human, there exists a gap in the CBIR approach. As no direct link is established between the low and high-level features, the semantic gap arises. However, it is a complex issue to solve the gap problem [1].
- With the enormous growth in the large-scale multimedia technology on the internet, especially images, constructing the CBIR system in the large scale environment becomes a challenging issue, as it used a large execution time [18].

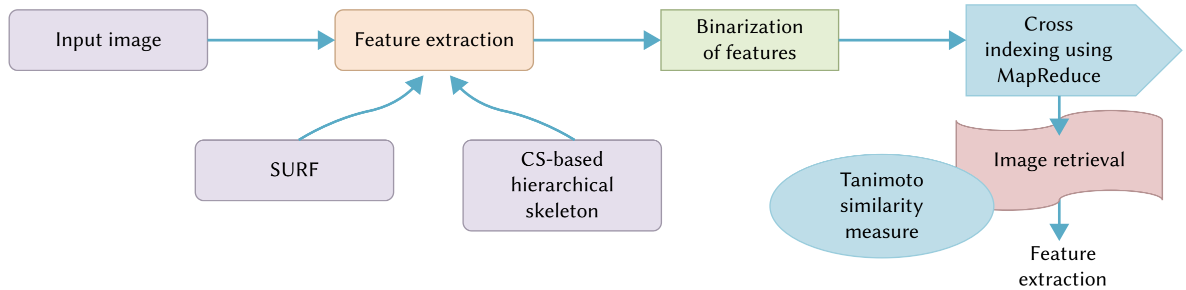

Fig. 1. Schematic diagram of the proposed CS-based hierarchical skeleton and cross-indexing.

- When a new image query is given to retrieve the specified object from the database, matching the image with the large pair of image sets poses a significant challenge [31].
- The explosive growth in the images also faces numerous challenges. One of the key challenges is helping the user retrieve their expected contents from the large volume database [3].
- The CCBST model developed in [2] was highly suitable to retrieve the images present at the top level based on the user query, but categorizing the features remains a crucial challenge in computer-based applications.

III. Proposed Cosine Similarity-based Hierarchical Skeleton and Cross-indexing for Image Retrieval Using MapReduce Framework

Image retrieval is a major consequence in the research domain to specify images based on the feature vector. Even though various image retrieval methods are available, defining the feature-based image representation poses a major challenge. Hence, a new method named CS-based hierarchical skeleton and cross-indexing is proposed in this research to solve the above issues. Initially, the input image is passed to the feature extraction stage, where the features are effectively extracted using the SURF and CS-based hierarchical skeleton. After extracting the features, the binarization of the feature vector is performed on the extracted features. Then, the binarization of the feature vector is given as the input to cross-indexing. The cross-indexing is performed in the mapper based on the binary sequences of the feature vector. Finally, the retrieval process is carried out through the reducer in the MapReduce framework based on the tanimoto similarity measure. Fig. 1 shows the block diagram of the proposed cross-indexing method.

A. Get the Input Image

At first, the input image used to perform the retrieval process using the MapReduce framework is collected from the dataset. Equation (1) refers to the database α with n number of images.

$$\alpha = \{M_1, M_2, \dots M_j, \dots M_n\} \tag{1}$$

where, α represents the database, and M denotes the images, M_n represents the total number of images.

B. Feature Extraction

Feature extraction is the process to reduce the dimensionality value through which the original input image is reduced to a manageable form. Feature extraction is highly important in the image retrieval mechanism to minimize the number of resources required to process the image without losing the relevant or important information. Feature extraction helps to minimize redundant values, which facilitates the speed of the retrieval process. The input image M_j is selected to perform the feature extraction process using SURF and CS-based hierarchical skeleton.

i) ***SURF***: The input image M_j is applied to the feature extraction module named as SURF, where the features from the image are effectively extracted and is denoted as f_1. SURF [32] uses the determinants of hessian matrices for locating the significant points of images based on the location and scale. Here, the dominant orientation is determined by computing the sum of all the responses that lies in the sliding orientation window with an angle of 60 degrees. SURF effectively extracts the robust features locally through the hessian matrix and the distribution-based descriptor. It uses the hessian detector to subtract the pyramid layers. Hessian detector identifies the interesting points or interest features for modifying the viewpoints. Here, the scale space is divided into various octaves and levels to achieve the scale invariance by examining the interesting points. The scale points are used to construct the pyramid levels based on the sub-sampling and Gaussian kernels. Non-maximal suppression of hessian matrices is the heart of SURF, which approximates the kernel with the rectangular box named as box filter. SURF is computationally faster and simpler without losing the performance. It is robust and stable because it has Hessian based detector. Moreover, surf uses only 64 dimensional vector. Equation (2) refers to the features extracted using the SURF based on the hessian matrix.

$$f_1 = X(y,z) = \begin{bmatrix} U_{aa}(y,z) & U_{ab}(y,z) \\ U_{ab}(y,z) & U_{bb}(y,z) \end{bmatrix} \tag{2}$$

where, $X(y, z)$ denotes the Hessian matrix, $U_{aa}(y, z)$ and $U_{ab}(y, z)$ denotes the convolution of the image with the second-order derivative of Gaussian $\lambda(z)$. Equations (3) and (4) define the terms $U_{aa}(y,z)$ and $U_{ab}(y,z)$.

$$U_{aa}(y,z) = J(p) * \frac{\partial^2 g}{\partial p^2} \lambda(z) \tag{3}$$

$$U_{ab}(y,z) = J(p) * \frac{\partial^2 g}{\partial p^2} \lambda(z) \tag{4}$$

$J(p)$ denotes the integral image and $p = (p,g)^T$ is used to store the sum of all the pixels in the rectangular area. Here, f_1 it represents the features extracted using SURF. Accordingly, the features extracted using SURF are specified in the matrix form as depicted in Fig. 2.

2	10	12	8	7
4	8	3	15	20
25	30	31	17	35
4	9	6	21	37
40	39	18	15	25

Fig. 2. Features extracted using SURF.

The output obtained from the SURF is in the form of a matrix and the pixel values are specified in the decimal format.

ii) ***CS-based hierarchical skeleton***: The input image M_j is applied to the CS-based hierarchical skeleton feature extraction module in order to extract the features, which is further used to achieve the image retrieval process. Moreover, the features extracted by the CS-based hierarchical skeleton feature extraction module are represented as f_2. CS-based hierarchical skeleton [33] is highly effective in extracting the features from the image. The CS-based hierarchical skeleton is derived by inheriting CS features with the hierarchical skeleton. The major advantage behind the CS-based hierarchical skeleton is that it provides additional information to the image retrieval process through which the accuracy of the retrieval mechanism is enhanced. It does not require any extra computational cost and effectively captures the topological and geometric features at various levels based on the skeleton pruning. The CS-based hierarchical skeleton takes the benefit of skeleton pruning, which eliminates the skeleton branches of insignificant parts of the shape. Equation (5) refers to the features extracted using the CS-based hierarchical skeleton.

$$f_2 = A(q_1, q_2) = A_1(q_1, q_2) + A_2(q_1, q_2) \tag{5}$$

Equations (6) and (7) define the terms $A_1\ (q_1, q_2)$ and $A_2\ (q_1, q_2)$.

$$A_1(q_1, q_2) = \frac{\mu(q_1, q_2)k(q_1)k(q_2)}{k(q_1)+k(q_2)} \tag{6}$$

$$A_2(q_1, q_2) = k(q_1).k(q_2)\cos\big(\mu(q_1, q_2)\big) \tag{7}$$

where, $\mu\ (q_1, q_2)$ denotes the angle of corner location of q_1 and q_2, k denotes the length function, and f_2 represents the features extracted using CS-based hierarchical skeleton. The above equation (6) is modified with the CS measure based on the Cosine of two non-zero vectors using Euclidean dot product. Accordingly, the features extracted using a CS-based hierarchical skeleton are specified in the matrix form as represented in Fig. 3.

0	1	0	0	1
1	1	0	0	1
1	0	1	1	1
0	1	1	1	0
1	0	1	0	1

Fig. 3. Features extracted using CS-based hierarchical skeleton.

The output features obtained from the CS-based hierarchical skeleton are in the form of a matrix, and the pixel values are specified in the binary format.

C. Binarization of Features

The binarization process is required in the image retrieval process to enhance the retrieval accuracy in the MapReduce framework. Binarization is a tool in the image retrieval process for specifying the interested region image and its background. It is converting the pixel values of the image into the binary equivalent vector form to differentiate the interesting regions. The binarization process is performed by applying the 3×3 non-overlapping windowing mechanism to the feature matrices. In the binarization process, the pixel values are separated into two binary forms as either '0' or '1' based on the values of the Hessian matrix. The output features obtained from the SURF and CS-based hierarchical skeleton are allowed to perform the binarization process to generate the binary vector for the pixel values. The output features extracted from the SURF are in the form of a matrix with decimal pixel values. Hence, it is required to convert the matrix from the decimal to the binary format. The binarization of features will generate the binary vector by applying the 3×3 non-overlapping window to the binary form matrix. Here, the output obtained from the SURF is the decimal value matrix, while the output obtained from the CS-based hierarchical skeleton is the binary form matrix.

Let us first convert the output features obtained using SURF from the decimal form matrix into the binary form matrix by averaging the neighborhood pixels. Let us consider the first decimal pixel value '2' from Fig.2, and form a 3×3 matrix using the pixel's neighboring pixel value '2', which is depicted in Fig. 4.

0	0	0
0	2	10
0	4	8

Fig. 4. A 3×3 matrix formation for pixel value '2'.

Let us select the pixel value '2' and take the average of neighborhood pixel values to generate a decimal form matrix. For the pixel value '2', the value obtained by averaging the neighborhood pixels is 2.75, compared with the original pixel value '2'. When the original pixel value is smaller than the pixel value obtained by averaging the neighborhood pixels, then the binary value placed in the binary form matrix is '0'. If the original pixel value is greater than the pixel value obtained by averaging the neighborhood pixels, then the binary value '1' is placed in the binary form matrix. Similarly, the entire pixel values present in the matrix of Fig. 2 are converted into the binary form matrix as depicted in Fig. 5.

0	1	1	1	1
1	0	1	0	1
1	0	0	1	0
0	1	1	0	0
1	0	0	1	1

Fig. 5. Binary representation of the features extracted using SURF.

The binary form of the features extracted from the CS-based hierarchical skeleton and SURF is represented in Fig.3, and Fig.5, respectively. From the binary matrix, the binary vector of the pixel values is computed by applying the 3×3 non-overlapping windowing process.

i) ***Compute the binary vector for SURF features***: The binary matrix generated for the features obtained using SURF is represented in Fig. 5. The binary matrix value is converted into the binary vector by applying the 3×3 non-overlapping windowing process. Let us form a 3×3 binary matrix including all the pixel values from Fig. 5 and apply the 3×3 non-overlapping windowing process to generate the binary vector, which is shown in Fig. 6.

0	1	1
1	0	1
1	0	0

(a)

0	1	1
1	0	0
0	0	0

(c)

1	1	0
0	1	0
1	0	0

(b)

0	0	0
1	1	0
0	0	0

(d)

Fig. 6. 3×3 binary matrix formation for SURF features.

The binary matrix representation for the SURF-based features is partitioned into various 3×3 binary matrices and is represented in Fig. 6 a), Fig. 6 b), Fig. 6 c), and Fig. 6 d), respectively. The binary values are grouped as 011101100, 110010100, 011100000, and 000110000 and are converted into the decimal form. The binary values present in each of the 3×3 matrix is converted into the decimal format and is specified as 236 for Fig.6 a), 404 for Fig.6 b), 224 for Fig. 6 c), and 48 for Fig. 6 d), respectively. Therefore, the binarization of the feature vector is formed using the decimal values and is represented in Fig. 7.

236	404	224	48

Fig. 7. Binarization of feature vector based on SURF features.

ii) ***Compute the binary vector for CS-based hierarchical skeleton features***: The binary matrix generated for the features obtained using the CS-based hierarchical skeleton is depicted in Fig. 3. Here, the binary vector for the pixel values is generated by applying the 3×3 non-overlapping windowing process to the binary matrix. Let us form the 3×3 binary matrix including all the pixel values from Fig. 3 and apply the 3×3 non-overlapping windowing process to generate the binary vector, which is represented in Fig. 8.

0	1	0
1	1	0
1	0	1

(a)

0	1	1
1	0	1
0	0	0

(c)

0	1	0
0	1	0
1	1	0

(b)

1	0	0
0	1	0
0	0	0

(d)

Fig. 8. 3×3 binary matrix formation for CS-based hierarchical skeleton features.

The binary matrix representation for CS-based hierarchical skeleton features is partitioned into various 3×3 binary matrices and is represented in Fig. 8 a), Fig. 8 b), Fig. 8 c), and Fig. 8 d), respectively. The binary values are grouped as 010110101, 010010110, 011101000, and 100010000 and are converted into the decimal form. The binary values present in each of the 3×3 matrix is converted into the decimal format and is specified as 181 for Fig. 8 a), 150 for Fig. 8 b), 232 for Fig. 8 c), and 272 for Fig. 8 d), respectively. Therefore, the binarization of feature vector is formed using the decimal values and is represented in Fig. 9.

181	150	232	272

Fig. 9. Binarization of feature vector based on CS-based hierarchical skeleton features.

Finally, the resultant binarization of feature vector obtained based on the SURF features is added with the resultant binarization of feature vector obtained using the CS-based hierarchical skeleton features in order to perform the cross-indexing process using MapReduce framework. Fig. 10 portrays the binarization of the feature vector.

236	404	224	48

+

181	150	232	272

417	554	456	320

Fig. 10. Binarization of feature vector.

By adding the feature vector of SURF-based features with the feature vector of the CS-based hierarchical skeleton, the feature vector's final output is generated based on the decimal representation. Finally, the decimal form of the feature vector is converted into the binary format and is denoted as P. P specifies the binary equivalent of the decimal value H; ($H = 417 + 554 + 456 + 320$).

D. Proposed Cross Indexing Based on the Feature Vector

Cross indexing is the process of transforming the features into binary codes to simplify the retrieval process. The features of the images are indexed with the binary value to make the retrieval process more effective. Most of the existing techniques failed to perform the mapping process to save the images in the mapper. The cross-indexing model [34] executes the binarization of the feature vector to make the image retrieval process highly robust and effective. The output of the binarized feature vector is used to perform cross-indexing based on the MapReduce framework. The feature vector for the input image M_j is denoted as P, which specifies the binary equivalent of H. The MSB of P is considered to record the image in the mappers. Based on the MSB of the features, the input images are placed in the mappers simultaneously.

The proposed CS-based hierarchical skeleton and cross-indexing effectively perform the image retrieval process based on the MSB code of features. Hence, it is required to transform the feature vector in the respective binary equivalent. Saving the images in the mappers based on the MSB code helps increase the retrieval process's efficiency, which further increases the overall system performance. When the MSB bits are selected to two bits as, 00, 01, 10, and 11, it requires four different mappers, namely 00 in mapper 1, 01 in mapper 2, 10 in mapper 3, and 11 in mapper 4, to save the images. When the MSB bits are selected as 000, 001, 010, 011, 100, 101, 110, and 111 by considering three bits, then it requires eight different mappers, such as 000 in mapper 1, 001 in mapper 2, 010 in mapper 3, 011 in mapper 4, 100 in mapper 5, 101 in mapper 6, 110 in mapper 7, and 111 in mapper 8 for saving the images. Hence, based on the MSB bits of the binary feature vector, the images are stored in the mappers. For example, if the input image and their MSB bits of the image are given as 10, then the image will be stored in mapper 3. Similarly, the MSB bits for all the incoming images are verified based on the MSB feature code. The images are stored in the mapper through the MapReduce framework.

E. Image Retrieval Using MapReduce Framework

Image retrieval is the process of extracting the images based on the feature vector. The binary sequence for each image is computed and stored in the mapper based on the MSB code of the features. The retrieval process is carried out using the reducer in the MapReduce framework. The MapReduce framework consists of two different tasks, namely mapper and reducer. The images are stored in the mapper based on the MSB bits of the feature vector, whereas the retrieval process is performed using the reducer. Based on the binary sequence, the images are stored in different mappers. Similarly, there exist various reducers through which the retrieval process can be achieved. The number of reducers is equivalent to the number of mappers present in the image retrieval process. When the user sends the request to retrieve the mapper's image, the query image will be transformed to the vector form by considering the binary sequences. The process of converting the query image into the binary sequence is done using the tanimoto measure. Equation (8) refers to the tanimoto measure.

$$S = \frac{\sum_{t=1}^{m} X_t R_t}{\sum_{t=1}^{m} X_t^2 + \sum_{t=1}^{m} R_t^2 - \sum_{t=1}^{m} X_t R_t} \tag{8}$$

where, S represents the tanimoto measure, X_t and R_t represents the feature vector of query image and mapper image, respectively. For each user query image, the binary sequence of the feature vector is calculated based on the tanimoto similarity measure. The MSB code of the binary sequences is selected as either two-bit or three-bit sequence, and the MSB code is verified with the mapper. As the images are saved in the mapper based on the MSB code, it must convert the query image into a binary representation. When the MSB code of the query image is verified with the mapper's MSB code, then the image is retrieved through the reducer in the MapReduce framework. When the query image is matched with the mapper 3, then the image in mapper in mapper 3 is retrieved through reducer 3. If the query image is matched with the image present in mapper 4, then the matched image is retrieved through reducer 4, respectively. Fig. 11 portrays the image retrieval process based on the feature vector using the MapReduce framework.

In Fig. 11, the user sends the query image to the mapper in order to retrieve the matched image result. The binary sequence for the user's query image is calculated in the MapReduce framework based on the feature vector. However, the MSB code of the binary sequence is 01. The request is to the mapper 2, where the matched results are gathered and retrieved through reducer 2 using the tanimoto similarity measure.

IV. Results and Discussion

The proposed method is evaluated with respect to the existing methods based on precision, F-measure, and recall parameters. The analysis is done by employing two images taken from the UK bench dataset.

A. Experimental Setup

The experimentation is done in MATLAB tool using PC with 4 GB RAM, Windows 10 OS, and Intel I3 processor. The proposed method employs two different images to analyze the performance.

1. Description of Datasets

The UK bench dataset [35] is designed by Nister and Stewenius. The dataset is a collection of several research domains, which include computer vision and image retrieval. The dataset is published in 2006 with open source_media collection.

2. Performance Metrics

The Proposed CS-based hierarchical skeleton system's performance, the metrics like Precision, Recall, F1-score, computational cost, and computational time have been considered.

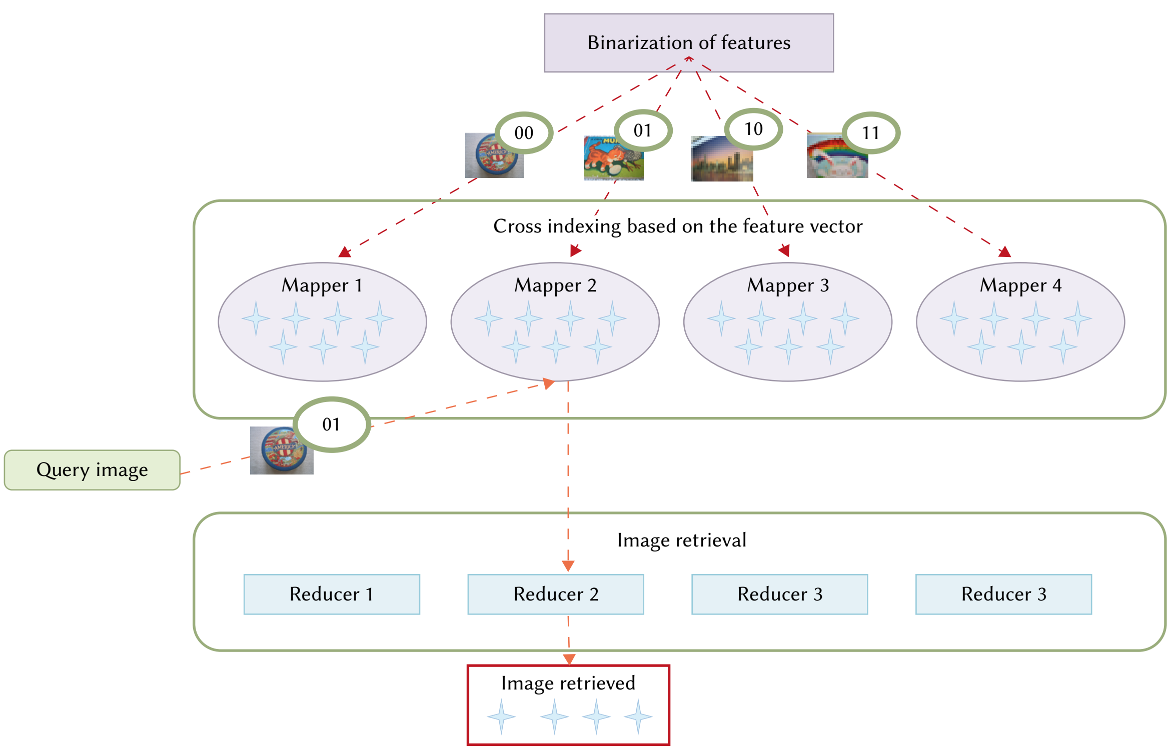

Fig. 11. Image retrieval based on the feature vector using MapReduce framework.

Fig. 12. Experimental results of proposed CS based hierarchical skeleton using a) Original image b) Extracted SURF feature c) Hierarchical skeleton image d) Query image e) Retrieved images from four reducers.

3. Comparative Methods

The analysis is made using comparative methods like Deep convolution neural network (DCNN) [1], Progressive image retrieval [3], MKSIFT+ Cross indexing [34], and Proposed CS-based hierarchical skeleton.

B. Experimental Results

This section elaborates the analysis of performance using images taken from UK bench dataset. Fig. 12 represents the experimental results of the proposed method using two different input images. Fig. 12a) represents two input image obtained from the UK bench dataset. Fig. 12b) represents the image in which the SURF features are extracted using the provided input image. Fig. 12c) portrays the hierarchical skeleton feature extracted from the input image. The query image is provided for retrieving required images, which is represented in Fig. 12d). Finally, Fig. 12e) illustrates the retrieval images obtained from the four sets of reducers.

C. Comparative Analysis

The comparison between proposed CS based hierarchical skeleton and existing techniques is done based on Recall, F-measure, and Precision using two images. The methods are analyzed using four and eight mappers using two query sets.

1. Analysis Using Four Mappers

The analysis of methods using four mappers with two query sets is elaborated in the subsections. The effectiveness of proposed method is evaluated using precision, recall and F1-Score parameters.

a) Based on Query Set-1

Fig. 13 presents the analysis of methods using four mappers considering query set-1 in terms of F1-score, precision, and recalls parameters. The analysis based on F1-Score using query set-1 is illustrated in Fig. 13a). When the number of retrieval is 2, the corresponding F1-Score values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.664, 0.719, 0.720, and 0.778, respectively. Likewise, for 4 retrievals, the corresponding F1-Score values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.721, 0.686, 0.727, and 0.784, respectively. The analysis based on precision parameter using query set-1 is illustrated in Fig. 13b). When the number of retrieval is 2, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.742, 0.764, 0.776, and 0.816, respectively. Likewise, for 4 retrievals, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.646, 0.650, 0.660, and 0.729, respectively. The analysis based on recall parameter using query set-1 is illustrated in Fig. 13c). When the number of retrieval is 2, the corresponding recall values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.55, 0.6, 0.6, and 0.65, respectively. Likewise, for 4 retrievals, the corresponding recall values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.7, 0.65, 0.7, and 0.75, respectively. The analysis based on recall parameter using query set-1 is illustrated in Fig. 13d). When the number of retrieval is 2, the corresponding computational cost values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 34.870, 32.985, 33.931, and 32.772, respectively. Likewise, for 4 retrievals, the corresponding recall values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 49.185, 46.814, 48.585, and 46.285, respectively. The analysis based on recall parameter using query set-1 is illustrated in Fig. 13e). When the number of retrieval is 2, the corresponding computational time values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 30.059, 27.238, 28.403, and 26.189, respectively. Likewise, for 4 retrievals, the corresponding recall values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 34.597, 32.512, 33.909, and 29.9830, respectively.

b) Based on Query Set-2

Fig. 14 presents the analysis of methods using four mappers considering query set-2 in terms of F1-score, precision, and recalls parameters. The analysis based on F1-Score using query set-1 is illustrated in Fig. 14a). When the number of retrieval is 2, the corresponding F1-Score values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.648, 0.638, 0.648, and 0.707, respectively. Likewise, for 4 retrievals, the corresponding F1-Score values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.580, 0.562, 0.579, and 0.634, respectively. The analysis based on precision parameter using query set-1 is illustrated in Fig. 14b). When the number of retrieval is 2, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.680, 0.626, 0.682, and 0.735, respectively. Likewise, for 4 retrievals, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.575, 0.527, 0.570, and 0.635, respectively. The analysis based on recall parameter using query set-1 is illustrated in Fig. 14c). When the number of is 2, the corresponding recall values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.484, 0.517, 0.551, and 0.556, respectively. Likewise, for 4 retrievals, the corresponding recall values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.591, 0.595, 0.596, and 0.727, respectively. The analysis based on computational cost parameter using query set-1 is illustrated in Fig. 14d). When the number of is 2, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 41.615, 33.598, 32.079, and 31.238, respectively. Likewise, for 4 retrievals, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 49.680, 42.376, 42.356, and 37.742, respectively. The analysis based on computational time parameter using query set-1 is illustrated in Fig. 14e). When the number of retrieval is 2, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 21.540, 16.656, 18.112, and 15.838, respectively. Likewise, for 4 retrievals, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 22.942, 21.019, 22.481, and 19.505, respectively.

2. Analysis Using Eight Mappers

The analysis of methods using eight mappers with two query sets is elaborated in subsections. The efficiency of proposed method is evaluated with precision, recall and F1-Score measures.

a) Based on Query Set-1

Fig. 15 presents the analysis of methods using eight mappers considering query set-1 in terms of F1-score, precision, and recalls parameters. The analysis based on F1-Score using query set-1 is illustrated in Fig. 15a). When the number of retrieval is 2, the corresponding F1-Score values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.619, 0.675, 0.681, and 0.758, respectively. Likewise, for 4 retrievals, the corresponding F1-Score values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.676, 0.648, 0.686, and 0.735, respectively. The analysis based on precision parameter using query set-1 is illustrated in Fig. 15b). When the number of retrieval is 2, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.670, 0.611, 0.650, and 0.734, respectively. Likewise, for 4 retrievals, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.557, 0.491, 0.550, and 0.618, respectively. The analysis based on recall parameter using query set-1 is illustrated in Fig. 15c). When the number of retrieval is 2, the corresponding recall values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.519, 0.570, 0.597, and 0.629, respectively. Likewise, for 4 retrievals, the corresponding recall values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.693, 0.620, 0.658, and 0.705, respectively. The analysis based on computational cost parameter using query set-1 is illustrated in Fig. 15d). When the number of retrieval is 2, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 41.615, 33.598, 32.079, and 31.238, respectively. Likewise, for 4 retrievals, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 49.680, 42.376, 42.356, and 37.742, respectively. The analysis based on computational time parameter using query set-1 is illustrated in Fig. 15e). When the number of retrieval is 2, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 28.804, 27.858, 28.516, and 27.510, respectively. Likewise, for 4 retrievals, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 34.171, 32.572, 33.308, and 31.160, respectively.

b) Based on Query Set-2

Fig. 16 presents the analysis of methods using eight mappers considering query set-2 in terms of F1-score, precision, and recalls parameters. The analysis based on F1-Score using query set-1 is illustrated in Fig. 16a). When the number of retrieval is 2, the corresponding F1-Score values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.648, 0.638, 0.648, and 0.707, respectively. Likewise, for 4 retrievals, the corresponding F1-Score values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical

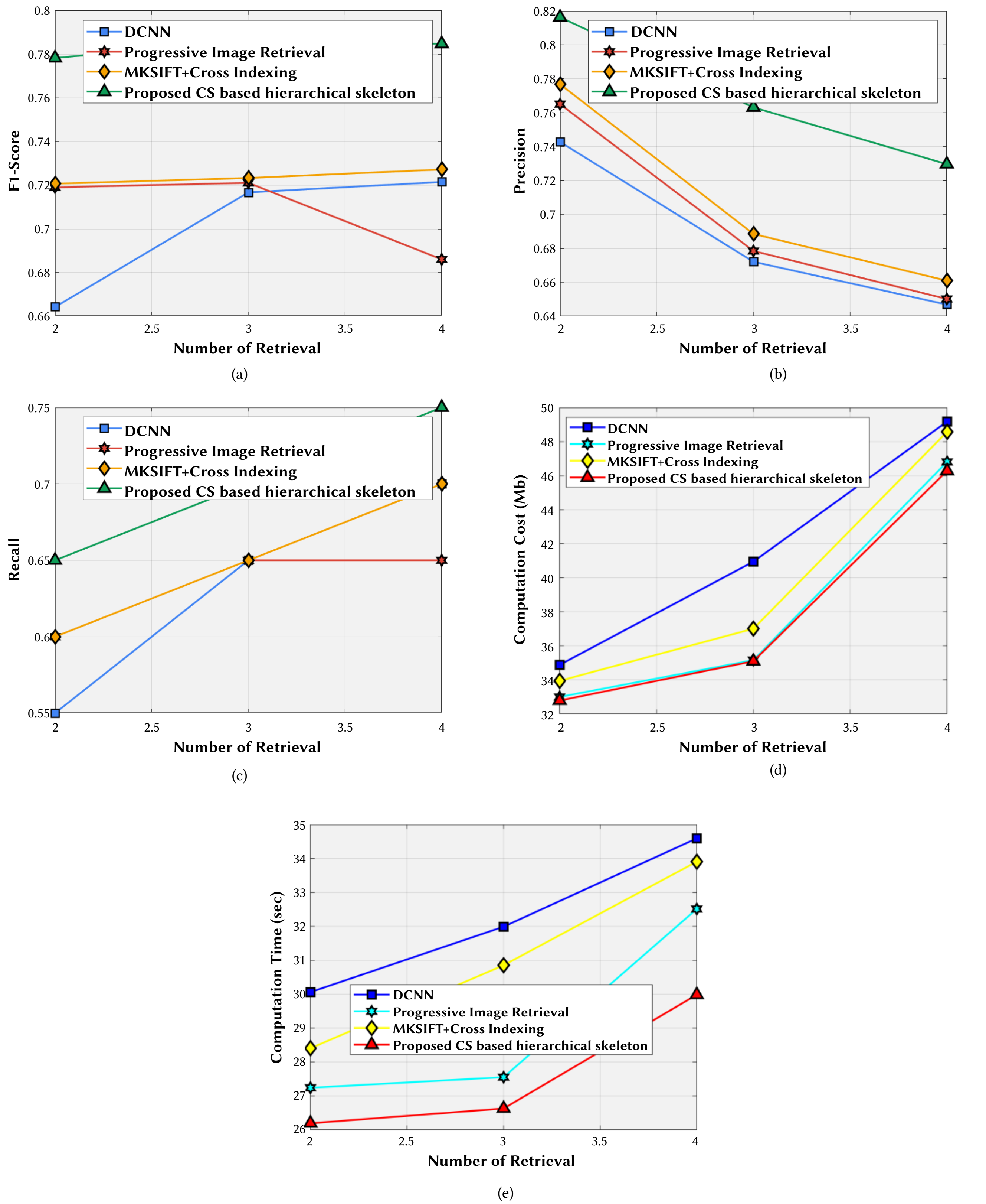

Fig. 13. Comparative analysis of methods using query set-1 with a) F1-Score b) Precision c) Recalld) Computational cost(Mb) and e) Computational Time (sec).

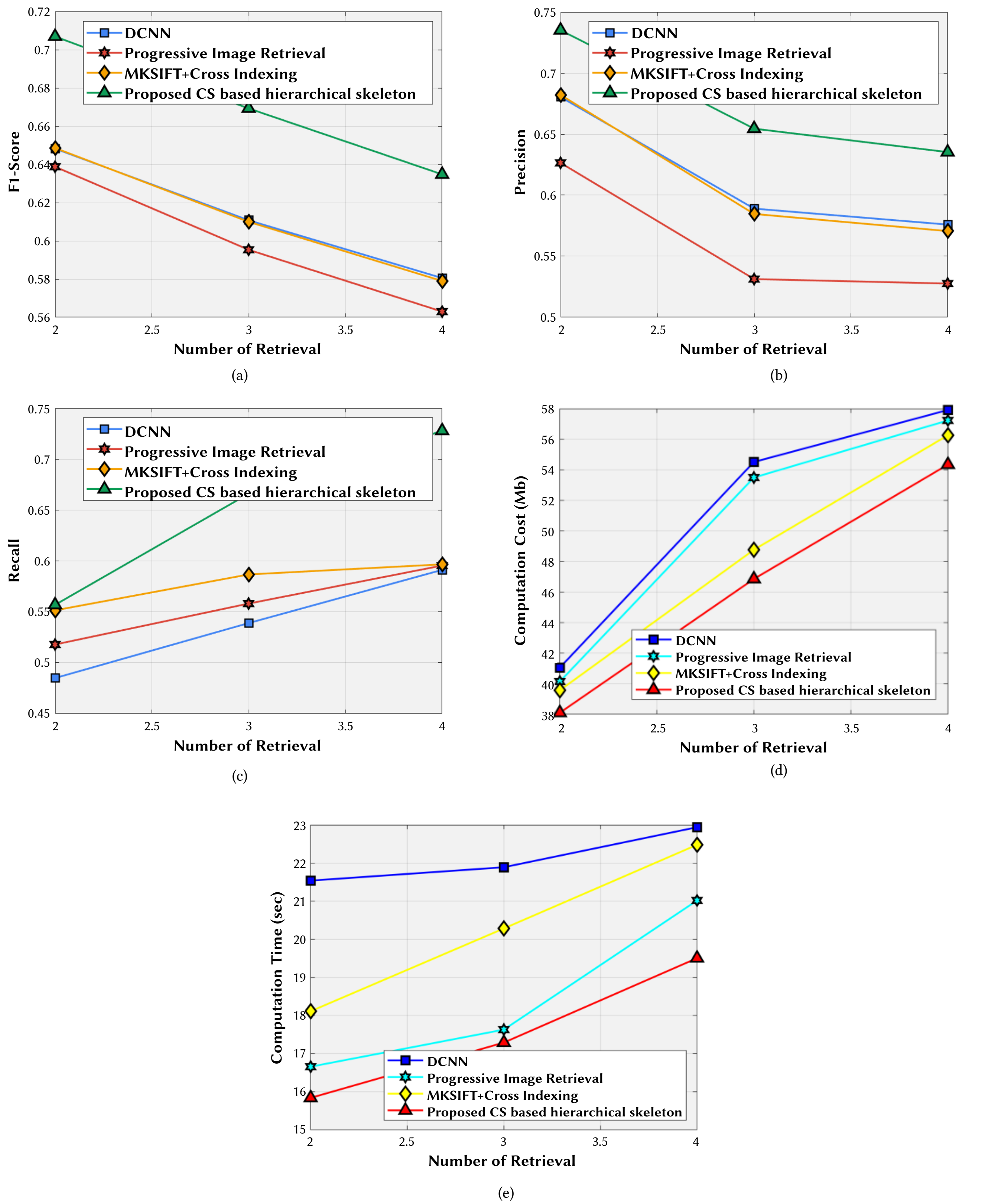

Fig. 14. Comparative analysis of methods using query set-2 with a) F1-Score b) Precision c) Recall d) Computational cost (Mb) and e) Computational Time (sec).

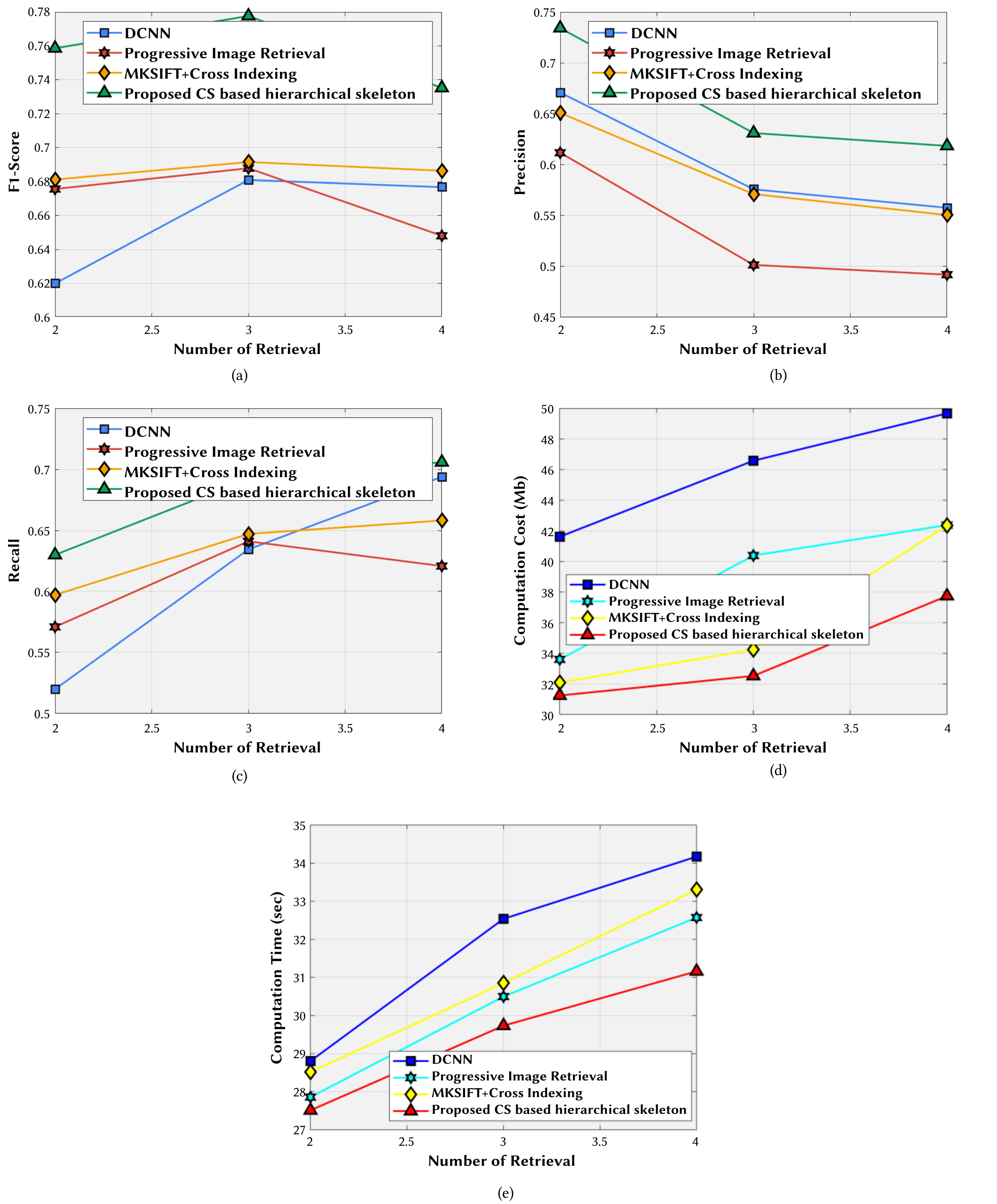

Fig. 15. Comparative analysis of methods using query set-1 with a) F1-Score b) Precision c) Recall d) Computational cost (Mb) and e) Computational Time (sec).

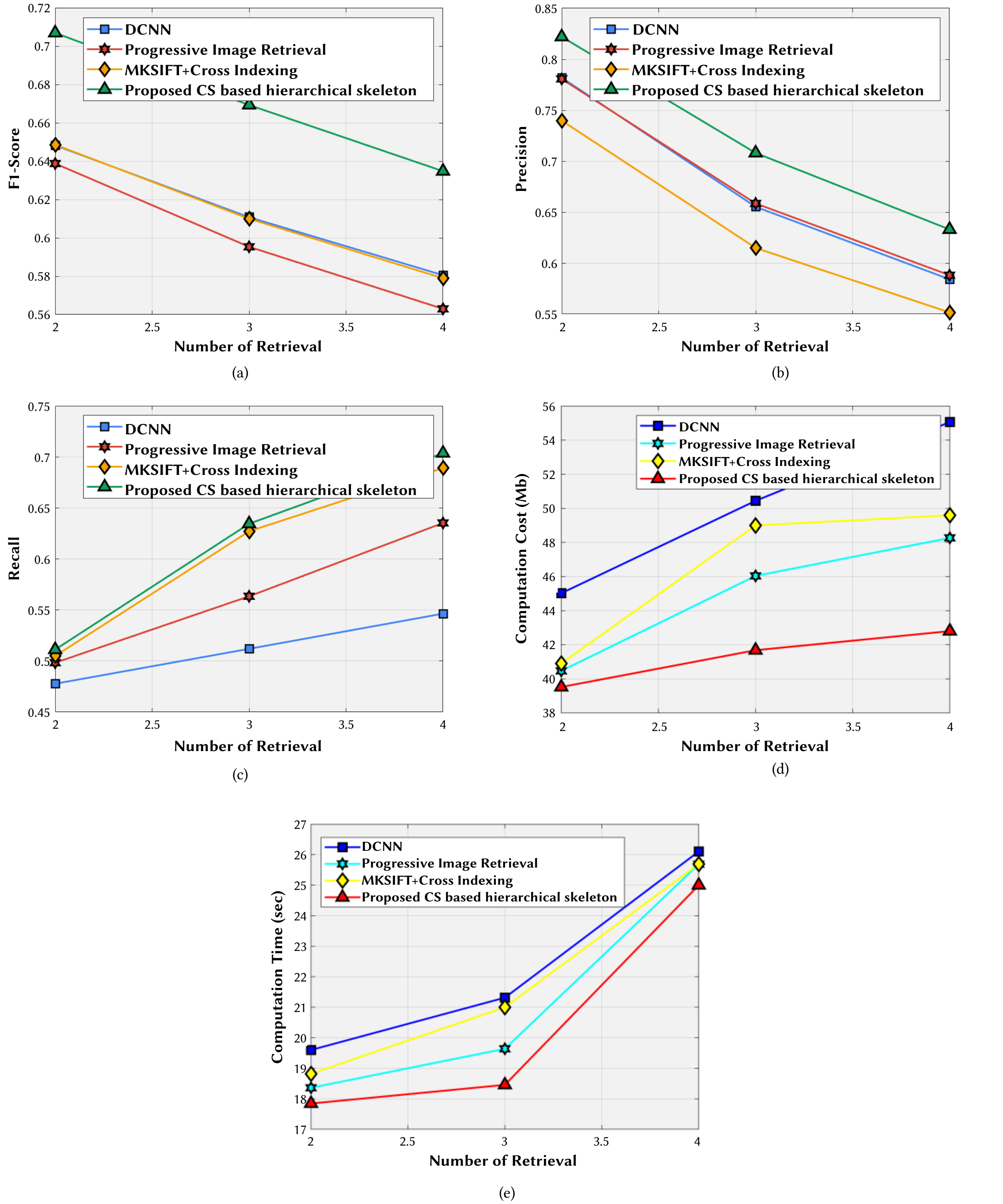

Fig. 16. Comparative analysis of methods using query set-2 with a) F1-Score b) Precision c) Recall d) Computational cost (Mb) and e) Computational Time (sec).

TABLE I. Comparative Analysis

Mappers	Query set	Metrics	DCNN	Progressive image retrieval	MKSIFT + Cross indexing	Proposed CS-based hierarchical skeleton
Using 4 mappers	Query set-1	F1-score	0.721	0.686	0.727	**0.784**
		Precision	0.646	0.65	0.66	**0.729**
		Recall	0.7	0.65	0.7	**0.75**
		CC	34.87	32.98	33.93	**32.77**
		CT	30.05	27.23	28.4	**26.18**
	Query set-2	F1-score	0.58	0.562	0.579	**0.634**
		Precision	0.575	0.527	0.57	**0.635**
		Recall	0.591	0.595	0.596	**0.727**
		CC	41.04	40.13	39.56	**38.09**
		CT	21.54	16.65	18.11	**15.83**
Using 8 mappers	Query set-1	F1-score	0.676	0.648	0.686	**0.735**
		Precision	0.557	0.491	0.55	**0.618**
		Recall	0.693	0.62	0.658	**0.705**
		CC	41.61	33.59	32.07	**31.23**
		CT	28.804	27.858	28.51	**27.51**
	Query set-2	F1-score	0.58	0.562	0.579	**0.634**
		Precision	0.584	0.588	0.552	**0.633**
		Recall	0.546	0.635	0.689	**0.703**
		CC	45.019	40.46	40.89	**39.519**
		CT	19.598	18.36	18.818	**17.844**

skeleton are 0.580, 0.562, 0.579, and 0.634, respectively. The analysis based on precision parameter using query set-1 is illustrated in Fig. 16b). When the number of retrieval is 2, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.782, 0.780, 0.740, and 0.822, respectively. Likewise, for 4 retrievals, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.584, 0.588, 0.552, and 0.633, respectively. The analysis based on recall parameter using query set-1 is illustrated in Fig. 16c). When the number of retrieval is 2, the corresponding recall values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.477, 0.498, 0.504, and 0.511, respectively. Likewise, for 4 retrievals, the corresponding recall values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 0.546, 0.635, 0.689, and 0.703, respectively. The analysis based on computational cost parameter using query set-1 is illustrated in Fig. 16d). When the number of retrieval is 2, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 45.019, 40.466, 40.899, and 39.519, respectively. Likewise, for 4 retrievals, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 55.060, 48.264, 49.594, and 42.798, respectively. The analysis based on computational time parameter using query set-1 is illustrated in Fig. 16e). When the number of retrieval is 2, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 19.598, 18.360, 18.818, and 17.844, respectively. Likewise, for 4 retrievals, the corresponding precision values computed by existing DCNN, Progressive image retrieval, MKSIFT + Cross indexing, and Proposed CS-based hierarchical skeleton are 26.106, 25.686, 25.692, and 25.000, respectively.

D. Comparative Discussion

Table I illustrates the analysis of methods using four and eight mappers along with two query sets. The analysis is done by adapting the maximal performance attained by each methods using precision, F1-Score and recall parameters. Considering four mappers, the proposed method attained better performance with maximal F1-Score of 0.784, maximal precision of 0.729 and maximal recall of 0.75, minimal computational cost of 32.77, and minimal computational time of 26.18 for query set-1. Using eight mappers, the proposed method attained maximal F1-Score of 0.735, and maximal recall of 0.705 for query set-1. The maximal precision is obtained in query set-2 with precision value of 0.633. Also, using 8 mappers, the proposed method has the minimal computational cost and computational time of 31.23 and 17, 84 sec, respectively for query set-1. From the analysis, it is noted that the best performance is achieved by proposed method, which shows its effectiveness in image retrieval.

The proposed CS-based Hierarchical Skeleton and Cross Indexing for Large Scale Image Retrieval Using Mapreduce Framework has the better results than the existing methodologies. This is happened because the proposed method has the benefit like low complexity for sparse vector. The cross indexing in image retrieval mechanism helps to improve the accuracy of image retrieval.

V. Conclusion

In this research, a new image retrieval method using CS-based hierarchical skeleton and cross indexing is proposed to perform the image retrieval mechanism based on the feature vector. The proposed method effectively achieves better retrieval performance through the cross indexing model. The features extracted using SURF and CS-based hierarchical skeleton are transformed into the binary sequences. The feature vector with the representation of binary code ensures the effectiveness of retrieval process based on the MSB code. The images stored in the mapper are retrieved based on the binary sequences using the tanimoto similarity measure. The retrieval process is carried out through the reducer in the MapReduce framework based on the MSB code of the binary sequences. The proposed image retrieval is effectively operated with the binary sequence rather than considering the decimal value. The proposed attained better performance with the values of 0.784, 0.729, 0.75, 31.23, and 17.84sec for F1-score, precision, recall, computational cost and computational time with the query set-1 by considering 4 mappers. The proposed image retrieval is widely useful in large scale image processing field for image retrieval in effective and efficient manner on cloud environments. Moreover it is helpful in the medical field for the diagnosis aids. In future, the performance of image retrieval mechanism will be increased by some other model for saving the image in the mapper.

References

[1] C.Bai, L. Huang, X. Pan, J. Zheng, S. Chen, "Optimization of deep convolutional neural network for large scale image retrieval," *Neurocomputing*, vol. 303, pp. 60-67, 2018.https://doi.org/10.1016/j.neucom.2018.04.034

[2] N.A. Sakr, A.I. Eldesouky, H. Arafat, "An efficient fast-response content-based image retrieval framework for big data," *Computers & Electrical Engineering*, vol. 54, pp.522-538, 2016. https://doi.org/10.1016/j.compeleceng.2016.04.015

[3] X. Gao, X. Shi, G. Zhang,J. Lin, M. Liao, K.C. Li, C. Li, "Progressive Image Retrieval With Quality Guarantee Under MapReduce Framework," *IEEE Access*, vol. 6, pp.44685-44697, 2018.https://doi.org/10.1109/ACCESS.2018.2842796

[4] J. Cao, M. Wang, H. Shi, G. Hu, Y. Tian, "A new approach for large-scale scene image retrieval based on improved parallel-means algorithm in mapreduce environment," *Mathematical Problems in Engineering*, 2016. https://doi.org/10.1155/2016/3593975

[5] H. Wang, Z. Li, Y. Li, B.B. Gupta, C. Choi, "Visual saliency guided complex image retrieval," *Pattern Recognition Letters*, Volume 130, pp64-72, 2020. https://doi.org/10.1016/j.patrec.2018.08.010

[6] M. A. Alazeez, H. B. Abdalla, G. Li, J.Lin, "NoSQL Injection: Data Security on Web Vulnerability," *International Journal of Security and Its Applications*, vol. 10, no. 9, pp. 55-64, 2016. http://dx.doi.org/10.14257/ijsia.2016.10.9.07

[7] H. B. Abdalla, J. Lin, G. Li, and M. Gilani, "NoSQL: Confidential on Data Security and Data Management by using a Mobile Application," *International Journal of Information and Electronics Engineering*, vol. 6, no. 2, pp. 84-88, 2016.https://doi.org/10.18178/IJIEE.2016.6.2.600

[8] C. Stergiou, K. E. Psannis, B. B.Gupta, "IoT-based Big Data secure management in the Fog over a 6G Wireless Network,"*IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5164 - 5171, 2021. https://doi.org/10.1109/JIOT.2020.3033131

[9] C. Yu, J. Li, X. Li, X. Ren, B. B. Gupta, "Four-image encryption scheme based on quaternion Fresnel transform, chaos and computer generated hologram,"*Multimedia Tools and Applications*, vol. 77, no. 19, pp. 4585–4608, February 2018.https://doi.org/10.1007/s11042-017-4637-6

[10] C. Yu, X. Li, S. Xu, and J. Li"Computer Generated Hologram-Based Image Cryptosystem with Multiple Chaotic Systems,"*Wireless Networks*, vol. 27, pp. 3507–3521, 2021.https://doi.org/10.1007/s11276-019-02223-z

[11] A.A.E. Latif, B.A.E.Atty, M.S.Hossain, A. Rahman, A. Alamri, and B. B. Gupta, "Efficient Quantum Information Hiding for Remote Medical Image Sharing,"*IEEE Access*, vol. 6, pp. 21075 – 21083,2018.https://doi.org/10.1109/ACCESS.2018.2820603

[12] W. Guo, N.K. Alham, Y. Liu, M. Li, M. Qi, "A resource aware MapReduce based parallel SVM for large scale image classifications," *Neural Processing Letters*, vol. 44, no. 1, pp.161-184, 2016.https://doi.org/10.1007/s11063-015-9472-z.

[13] V. Srivastava, S. Gupta, G. Chaudhary, A. Balodi, M. Khari, and V. G. Díaz, "An Enhanced Texture-Based Feature Extraction Approach for Classification of Biomedical Images of CT-Scan of Lungs," *International Journal of Interactive Multimedia and Artificial Intelligence* (Special Issue on Current Trends in Intelligent Multimedia Processing Systems), vol. 6, no. 7, pp. 18-25, 2021. http://doi.org/10.9781/ijimai.2020.11.003

[14] N. Lohar, D. Chavan, S. Arade, A. Jadhav, D. Chikmurge, "Content Based Image Retrieval System over Hadoop Using MapReduce,"*International Journal of Scientific Research in Science, Engineering and Technology*, Vol.2, no. 1, pp. 123-125, 2016.

[15] A.W.Smeulders, M. Worring, S. Santini, A. Gupta, R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 22, no. 12, pp.1349-1380, 2000.https://doi.org/10.1109/34.895972

[16] M. Alsmirat, F. Alalem, M. A.Ayyoub, Y. Jararweh, B. Gupta, "Impact of digital fingerprint image quality on the fingerprint recognition accuracy,"*Multimedia Tools and Applications*, vol.78(4), pp. 3649–3688, February 2019.https://doi.org/10.1007/s11042-017-5537-5

[17] M. Lagiewka, M. Korytkowski, R.Scherer, R, "Distributed image retrieval with colour and keypoint features," *Journal of Information and Telecommunication*, vol. 3, no. 4, pp.430-445, 2019.https://doi.org/10.1080/24751839.2019.1620023

[18] S. Mezzoudj, R. Seghir, Y. Saadna,"A Parallel Content-Based Image Retrieval System Using Spark and Tachyon Frameworks," *Journal of King Saud University-Computer and Information Sciences*, vol. 33, no. 2, pp. 141-149 2021https://doi.org/10.1016/j.jksuci.2019.01.003

[19] C. Yan, H. Xie, S. Liu, J. Yin, Y. Zhang, Q. Dai, "Effective Uyghur language text detection in complex background images for traffic prompt identification," *IEEE transactions on intelligent transportation systems*, vol. 19, no. 1, pp.220-229, 2018.https://doi.org/10.1109/TITS.2017.2749977

[20] C. Yan, Y. Zhang, J. Xu, F. Dai, J. Zhang, Q. Dai, F. Wu, "Efficient parallel framework for HEVC motion estimation on many-core processors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 12, pp.2077-2089, 2014.https://doi.org/10.1109/TCSVT.2014.2335852

[21] L. Deng, "A tutorial survey of architectures, algorithms, and applications for deep learning," *APSIPA Transactions on Signal and Information Processing*, vol. 3, 2014. https://doi.org/10.1017/atsip.2013.9

[22] G.E. Hinton, S. Osindero, Y.W.The, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp.1527-1554, 2006.https://doi.org/10.1162/neco.2006.18.7.1527

[23] M. Norouzi, M. Ranjbar, G. Mori, "Stacks of convolutional restricted boltzmann machines for shift-invariant feature learning," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2735-2742, June 2009. https://doi.org/10.1109/CVPRW.2009.5206577

[24] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu,F.E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp.11-26, 2017.https://doi.org/10.1016/j.neucom.2016.12.038

[25] D. Li, L. Deng, B.B. Gupta, H. Wang, C. Choi, "A Novel CNN based Security Guaranteed Image Watermarking Generation Scenario for Smart City Applications,"*Information Sciences*, vol. 479, pp. 432-447, 2019. https://doi.org/10.1016/j.ins.2018.02.060

[26] A. Sharif Razavian, H.Azizpour, J. Sullivan, S. Carlsson, "CNN features off-the-shelf: an astounding baseline for recognition," *In Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 806-813, 2014. https://doi.org/10.1109/CVPRW.2014.131

[27] R. Girshick, J. Donahue, T. Darrell, J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580-587, 2014.https://doi.org/10.1109/CVPR.2014.81

[28] S. Xibing, W. Rong, Y. Yi, "MapReduce Based Remote Sensing Image Retrieval Algorithm," *International Journal of Database Theory and Application*, vol. 9, no. 8, pp.1-12, 2016.http://dx.doi.org/10.14257/ijdta.2016.9.8.01

[29] Z. Meng, "Remote Sensing Image Retrieval Algorithm based on MapReduce and Characteristic Information," *International Journal of Simulation, Systems, Science and Technology*, vol. 17, no. 3, 2016. https://doi.org/10.5013/IJSSST.a.17.03.07

[30] X. Li, D. He, J.Y. Li, "Parallel image search application based on online hashing hierarchical ranking," *Cluster Computing*, vol. 22, no. 1, pp.1469-1478, 2019.https://doi.org/10.1007/s10586-018-1922-8

[31] C. Schmid, R. Mohr, "Local grayvalue invariants for image retrieval," *IEEE transactions on pattern analysis and machine intelligence*, vol. 19, no. 5, pp.530-535, 1997.https://doi.org/10.1109/34.589215

[32] U. Hany, L. Akter, "Speeded-Up Robust Feature extraction and matching for fingerprint recognition," IEEE International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), pp. 1-7, May 2015.https://doi.org/10.1109/ICEEICT.2015.7307439

[33] C. Yang, O. Tiebe, K. Shirahama, M. Grzegorzek, "Object matching with

hierarchical skeletons," *Pattern Recognition*, vol. 55, pp.183-197, 2016. https://doi.org/10.1016/j.patcog.2016.01.022

[34] B. Mathan Kumar, R. PushpaLakshmi, "Multiple kernel scale invariant feature transform and cross indexing for image search and retrieval," *The Imaging Science Journal*, vol.66, no.2, pp.84-97, 2018.https://doi.org/10.1080/13682199.2017.1378285

[35] Nister and Stewenius, "UK bench dataset", "https://archive.org/details/ukbench," accessed on September 10, 2019.

Zhong Qianwen

Zhong Qianwen is doing Electronic and Communication Engineering in Shangdong University, China, Shandong, Weihai, Huancui District, Shandong University. His area of interest is image processing, big data classification, Artificial Intelligence and Network and security. He has published many papers in various reputed journals.