IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 9, 2019, accepted January 18, 2020, date of publication January 28, 2020, date of current version February 6, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2970123

Stereoscopic Image Feature Indexing Based on Hybrid Grid Multiple Suffix Tree and Hierarchical Clustering

FENGFENG DUAN[1,2] AND QICONG ZHANG[3]

[1]School of Journalism and Communication, Hunan Normal University, Changsha 410081, China
[2]Hunan Social Public Opinion Monitoring and Network Public Opinion Research Center, Changsha 410081, China
[3]School of Computer Science, Communication University of China, Beijing 100024, China

Corresponding author: Fengfeng Duan (dffeng2010@126.com)

This work was supported in part by the National Social Science Fund Project of China "Research on intelligent acquisition, analysis, and processing technology and application of cross media network public opinion big data" (No. 18BXW109).

ABSTRACT In order to achieve content-based binocular stereoscopic image or video retrieval efficiently, a feature indexing algorithm based on hybrid grid multiple suffix tree and hierarchical clustering is proposed. With the RGB-D image model, the shape features of depth map obtained from the matching of binocular stereoscopic left image and right image and the color features of left image are extracted respectively. The features are quantified and hashed, and the optimized underlying features are sorted as leaf nodes of hierarchical indexing. Then the shape and color feature values of leaf nodes are mapped to the two-dimensional coordinates, and the 2D feature points are put into different grid hash areas respectively by clustering and labeled with multiple suffix tree. Furthermore, to construct the global index, a pointer to an array of clustering grid center point is defined according to the computation of grid area feature values. The experimental results show that compared to the double grid suffix tree and typical stereoscopic image feature indexing structure, the proposed algorithm can effectively reduce the indexing construction complexity While maintaining high recall, it can also greatly improve the query efficiency, which can better realize the feature indexing of binocular stereoscopic images or videos.

INDEX TERMS Feature indexing, hierarchical clustering, hybrid grid multiple suffix tree, stereoscopic image.

I. INTRODUCTION

With the development and application of new media technologies and network platforms, the analysis and acquisition of multimedia information become an important demand area, which also makes it become a key application and research field of multimedia database. The big data and continuous increasing of unstructured data in network database require constantly optimization of data acquisition in accuracy and speed. It is necessary to consider the characteristics of data content and organization form to improve the application efficiency. As an important part of network multimedia, image and video retrieval and acquisition have been attracting much attention [1]. Content-based image/video retrieval can help users obtain similar content more accurately [2]. In recent years, stereoscopic vision technologies have been widely concerned and developed rapidly. At the same time, the construction and application of network stereoscopic vision resources also increase rapidly, especially for the binocular stereoscopic image/video resources. However, the characteristics, structure and form of them are usually different from two-dimensional vision resources. It is a problem how to achieve content-based binocular stereoscopic image/video retrieval to meet the users' needs for resources accurately and rapidly. It requires feature extraction accurately and indexing construction efficiently for extracted high-dimensional features, so as to achieve fast matching and effective retrieval.

The raise of the amount data of videos and images will lead to complexity increasing, speed and accuracy reducing in retrieval. For matching and retrieval, each image or video will be extracted a large number of dimensional features [3]. If matching one by one, it will take a lot of time. This cannot meet the needs of users for fast and efficient

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

retrieval, which will result in curse of dimensionality in high-dimensional index. Fortunately, in research and practice, it is found that many of the features usually do not contribute much to matching and retrieval, and removing them has little impact on the accuracy. If these features can be skillfully removed, the efficiency of matching and retrieval will be greatly improved. But usually the dimension of feature is still large after dimensionality reduction and the complexity is still high. So, if we can avoid some irrelevant objects and reduce the number of feature matches, the retrieval efficiency can also be greatly improved, which is the advantage of indexing structure. Therefore, when we construct an efficient indexing structure, it will greatly improve the efficiency of retrieval, and support users to obtain the required resources accurately and quickly [4]. However, the construction of index will bring some related problems: 1) the construction of index needs certain storage space, which will lead to waste of resources; 2) the construction and maintenance of index will increase the time complexity of algorithm execution, which will reduce the efficiency; 3) related work shows that the construction of index often reduces the recall of matching and retrieval, and the higher the efficiency, the more the recall will be reduced [2], [4]. Therefore, an excellent indexing algorithm can reduce the storage space which has been occupied, have lower time complexity in execution, and achieve a balance between recall and efficiency.

II. RELATED WORK AND PROPOSED FLOW CHART

Index usually refers to the data structure arranged in a certain order according to the location and distribution characteristics of the objects, and the data structure contains the content information of the objects so as to facilitate the query and utilization. Indexing structure design is one of the key technologies of large-scale data file query and retrieval. An excellent indexing structure can support users to search for required files in massive data quickly. Low dimensional data structure is relatively simple, so the design of indexing structure algorithm is mainly for high dimensional data. Stereoscopic image or video indexing technology is usually for the content-based feature indexing. So, one of the keys of indexing lies in the type and method of feature extraction [5]–[9]. Since stereoscopic image features are still multi-dimensional vectors, which are similar to 2D image, so the related indexing algorithms are often used for stereoscopic vision feature. In some studies, indexing methods are grouped into four classes, including space partitioning, clustering, hashing, and product quantization [10]. For the current researches, the main focus is to use some related indexing methods on stereoscopic image or video features, in order to improve the efficiency of retrieval. So, we can give a brief overview in three dimensions for the related work.

A. HIERARCHICAL INDEXING STRATEGY

The principle and algorithm of hierarchical indexing structure, which is an important dimensionality reduction method of high dimensional indexing, have been paid great attention. Hierarchical indexing usually has two categories, the one is hierarchical structure based on classification, that is, different feature points are distributed at different levels, and features with similar attributes or classification are at the same level. Yang *et al*. [11] propose an image indexing structure algorithm based on parallel hierarchical K-means clustering, in which multi-processor parallel computing is used for node clustering. In the algorithm, the processing speed can be improved, but the complexity is high and the utilization efficiency is usually low. The other category is hierarchical-based index, in which each lower layer is indexed by corresponding high level, and the high level is dimensionality reduction or indexing pointer of the lower. Chiang *et al*. [12] propose a hierarchical indexing method of image features based on grid, in which the rough matching and accurate matching of features are executed in different levels. The method is innovative in framework design, but the number of grids is difficult to determine in implementation. Kiranyaz *et al*. [13] propose a high dimensional indexing method based on hierarchical unit tree structure to realize content-based multimedia data retrieval. The method can achieve retrieval quickly and the performance decreases less when the amount of data increases. However, it is difficult to set the model threshold, and the classification rule function is complex and difficult to implement. Song *et al*. [8] propose three-scale space indexing structure, which represents the global patterns, local details and fixed-length vectors respectively. These hierarchical indexing methods have some advantages, and also can be used for feature indexing of stereoscopic image/video in some way, but there are still some shortcomings that lead to poor efficiency.

B. TREE INDEXING ALGORITHM

In recent years, tree index which supports similarity retrieval are studied and applied widely. The popular tree index includes the methods of spatial division and data partition [14]. The index based on spatial division usually includes k-d-tree, adaptive k-d-tree and k-d-B-tree. The operation of the structures often inconvenient, especially the deletion is more complex, and the utilization rate of memory space is relatively low. The index based on data partition usually has balanced hierarchical trees, such as B-tree, B+-tree, and R-tree formed from B-tree extended to space [15]. The improved tree structures also have R∗-tree, X-tree, SS-tree, SR-tree and so on based on classic R-tree [16], and classic suffix tree [17], [18]. The disadvantage of the structures is the problem of overlap while dividing node region boundaries with rectangular or circular. In addition, for the extended application of SR-tree, A-tree with virtual boundary region and other indexing structures based on documents or in combination with trees are also proposed. In these related studies, there are some about indexing on stereoscopic object features. Liu *et al*. [19] propose the motion tree indexing algorithm with hierarchical motion description based on hierarchical tree of balance structure according to the stereo motion characteristics in

content-based 3D motion retrieval. The algorithm improves the retrieval efficiency, but the construction of indexing is complex. Feng *et al*. [20] construct randomized tree index in a supervised training for approximate nearest neighbor search of binary features. The tree index has uniform leaf sizes and low error rates. Deng *et al*. [21] propose the G-ML-Octree index for 3D moving objects, which improves the performance of indexing construction. These related methods tend to deteriorate rapidly in retrieval performance with increasing of dimension, that is so-called curse of dimensionality. For example, when the dimension is increasing, the retrieval time complexity of R-tree index will quickly tend to O (n).

C. CLUSTER INDEXING APPROACH

In order to better implement the construction of indexing, especially for the high dimensional indexing structure, clustering is introduced as an important analysis method. Clustering analysis is usually a statistical technology and process that divides a group of research objects into relatively homogeneous categories respectively and it is an unsupervised learning way. Clustering is of great significance for building efficient indexing structure. It is useful in classifying and gathering for the data with similar features to achieve better dimensionality reduction and similar query. The clustering methods usually include four categories. The first one is the clustering based on dividing method, such as K-means and improved K-modes, K-prototype, K-medoids algorithms. The second one is hierarchical clustering, including condensed and split clustering algorithms. The third one is neural network clustering based on model method, such as SOM (Self-organizing Mapping) algorithm. The fourth one is fuzzy clustering, such as FCM (Fuzzy C-means) algorithm [22]. The methods commonly used in practice include K-means clustering and related improved algorithms, hierarchical clustering, and so on. According to the characteristics of stereo vision, some clustering-based indexing structures and related methods are applied to indexing on stereoscopic image/video features to improve retrieval efficiency. Xu *et al*. [23] propose the approximate nearest neighbor (ANN) indexing structure algorithm based on cluster locality sensitive hashing for stereoscopic vision image features. It is an improved typical cluster indexing algorithm for stereoscopic vision image features, which improves the retrieval efficiency effectively. However, it is difficult to extend and apply in practice and the indexing reconstruction complexity is high when features are increasing. Ahmed *et al*. [24] propose the indexing multidimensional feature vectors based on locality sensitive hashing by partitioning the data space into zones and blocks, and the indexing locations are divided into nine sub-locations in maximum to store the data. But the fixed number of classifications will greatly increase the complexity when the number of images increases.

Many indexing structure algorithms can better realize the indexing of high dimensional features and improve the efficiency of retrieval. The traditional indexing structures are usually suitable for the original data, such as the structured or semi-structured data. However, with the development and application of network and new media technologies, the data presents new characteristics, such as complexity, diversity, large volume and personalization. Traditional indexing structures are often lack of analysis, processing and application to the characteristics of data content, which results in the inaccuracy and inefficiency of retrieval. Moreover, at present there are few methods for stereoscopic image/video feature indexing. Generally, there are two ways to improve the efficiency of high dimensional indexing, which are the dimensionality reduction and the indexing algorithm improvement [25]. In this paper, according to the principle and characteristics of binocular stereoscopic vision, the shape features of depth map obtained from the matching of binocular stereoscopic vision left image and right image and the color features of left image are extracted respectively. With the extracted features, we propose an indexing structure algorithm based on hybrid grid multiple suffix tree and hierarchical clustering (HGMST-HC) for binocular stereoscopic images. Compared with the existing typical indexing algorithms, the proposed algorithm in this paper can better achieve feature indexing of binocular stereoscopic images/videos and improve the efficiency of matching and retrieval. The contributions are as follows: 1) the memory consumption of indexing construction is smaller and the space complexity is lower; 2) the time required for indexing construction is less, and the maintenance of increasing, deleting and modifying is simpler by using hierarchical structure, which reduces the time complexity; 3) the efficiency of matching and retrieval can be improved greatly, and the construction as well as application of index has little impact on the retrieval accuracy, which can obtain a higher recall. The flow chart of the proposed indexing algorithm in this paper is shown in Fig. 1.

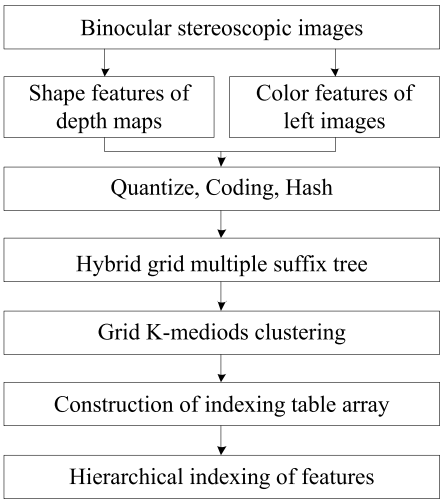

FIGURE 1. The flow chart of the proposed indexing algorithm.

III. FEATURE EXTRACTION OF BINOCULAR STEREOSCOPIC IMAGE

A. SHAPE FEATURE EXTRACTION

According to the matching between left and right images of binocular stereoscopic vision, disparity and depth values

can be calculated to obtain the depth maps and then the temporal-spatial consistency optimization is executed [26]. The depth map can be viewed as a two-dimensional image, but it is different in the way of formation. The 2D image is a projection of light reflection, while the depth map of 3D data is the projection and resample of depth value, which contains more intrinsic information of 3D. In the depth map, the gray value of pixel corresponds to the depth value of the scene. Usually, gray image has various changes in scene and its texture feature is obvious and complicated, while the depth map has different characteristics, including less changing in scene, simpler in texture and clearer in outline. At the same time, depth map is independent of color. So, compared with color image, the depth map is seldom affected by interference of light, shadow, and changes of environment. A binocular stereoscopic image and its depth map are shown in Fig. 2. Feature extraction of depth map based on shape can obtain accurate descriptors. They can not only effectively describe the shape of object, but also can better express the change information of depth direction. So, the features of depth map can be used as an important part of stereoscopic vision resource features with properties of translation, rotation and scale invariant. In this paper, the PCA-HODG algorithm is used to extract the shape features of depth maps [27].

FIGURE 2. A binocular stereoscopic image and its depth map.

Assume that the binocular stereoscopic left and right images are $m \times n$ in resolution respectively. According to the PCA-HODG algorithm, the selected window size is 64×128 in feature extraction of depth map. In addition, the method of blocks overlap is used in the algorithm. In order to obtain feature blocks as much as possible and accurate, sliding window detection over a depth map is performed to extract the features. At the same time, the overlap of windows is also performed to realize the window detection of full features. Considering the resolution of the depth map and the size of selected window, the depth map is divided into M windows and the number of M is defined as:

$$M = \psi \times \eta, \tag{1}$$

where $\psi = \lceil \frac{m}{64} \rceil$, represents the number of windows in horizon; $\eta = \lceil \frac{n}{128} \rceil$, represents the number of windows in vertical. In the sliding detection of windows, the jump values of window in horizontal and vertical direction are $\frac{m-64}{\psi-1}, \frac{n-128}{\eta-1}$ in each time respectively. So, the features of M windows can represent a full depth map through division and detection of windows. In this way, M feature sequences are constructed for each depth map. For example, in experiment,

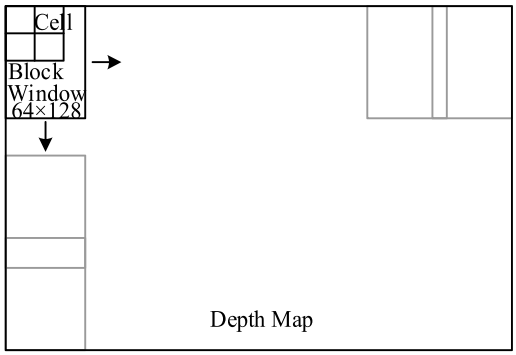

FIGURE 3. The example of sliding window in feature detection.

if the resolution of depth map is 640×480, then M is 40. The example of sliding window in feature detection is demonstrated in Fig. 3.

For the number of M feature sequences, each sequence has d-dimensional feature vectors and it will form $M \times d$ dimensional matrices. The method of PCA is used to reduce the dimensions of the data. Then the first d principal components are selected as the features of the full depth map. In experiment of the paper, the value of d is 20 [28].

B. COLOR FEATURE EXTRACTION

The left image of binocular stereoscopic vision is used for feature extraction. Color features can be obtained without high complexity and large amount of calculation. Moreover, they are often not sensitive to rotation, scaling, fuzzy and other physical transformation. It has great advantages to measure and represent the global difference of two images by color features based on histogram. So, color features can be selected as an important part for similarity matching and retrieval.

RGB is the most common color space in video and most of the digital images are also expressed with the RGB color space. However, the spatial structure does not meet the people in subjective judgment of color similarity. So, it is necessary to convert it into HSV space which is closest with the subjective perception of human eyes [29]. Each component of HSV color space is quantified as non-equal interval for 8 segments, 3 segments and 3 segments. At the same time, they are synthesized into one-dimensional feature vector [30]. The HSV color space can be quantified for 72 segments. Then the histogram distribution is calculated and is defined by:

$$H(k) = \frac{n_k}{L} \quad (k = 0, 1, 2, 3, \ldots, l-1), \tag{2}$$

where l represents the number of colors contained in the image, n_k represents the number of pixels whose quantified color value is k, L represents the total number of pixels within the image.

For color histogram, pixels with high frequency are selected as main colors, which is called the main color histogram. The pixels of low frequency can be regarded as noise. So, the main color histograms can represent features

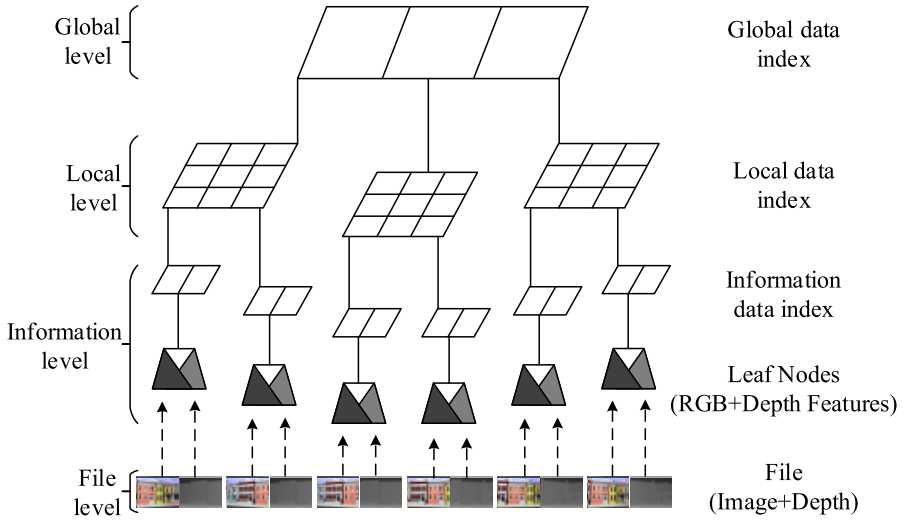

FIGURE 4. The indexing structure diagram of the proposed algorithm.

of image. In order to represent the image features more comprehensively, the method of cluster is used to acquire the main color histograms and the center of each cluster is considered as the main color in this paper. Based on the idea of K-means, the d dimensional feature values of main color histogram are calculated and the value of d is chosen according to the dimension of shape features. The algorithm can be described as flows:

Step 1: Initialization and the number of d elements $h_1, h_2, \ldots, h_d$ are selected arbitrarily as the cluster centers, then m cluster spaces $K_1, K_2, \ldots, K_D$ are established.

Step 2: For a sample x in the sample set X, it can be ajusted into cluster K_J which is corresponded by h_i according to the rule of minimum distance calculation $j = \arg\min_{1 \le i \le d} \delta(x, h_i)$, which is $x \in K_J$, where $1 \le i \le d, \quad 1 \le J \le D$.

Step 3: Calculate the mean of the elements in each cluster, which is $h_i = \frac{1}{n_i} \sum_{x \in K_J} x$, where n_i is the number of elements in the cluster space K_J, and then update each cluster center.

Step 4: If the cluster center is no longer changing or the value of E is the minimum, where $E = \sum_{i=1}^{d} \sum_{x \in K_J} \|x - h_i\|^2$, then the clustering is over, or turn to *Step 2*.

According to the algorithm above, d dimensional main color histogram features can be obtained, which are the final center h_i of each cluster, where $h_i \in \{h_1, h_2, \ldots, h_d\}$. In this paper, according to the feature dimension of depth map, the value of d is also 20 [28].

IV. THE INDEXING ALGORITHM OF HYBRID GRID MULTIPLE SUFFIX TREE AND HIERARCHICAL CLUSTERING

In order to overcome the disadvantages of the existing algorithms, and realize the feature indexing of binocular stereoscopic image/video pertinently to improve the query and retrieval efficiency, we propose an indexing structure algorithm based on hybrid grid multiple suffix tree and hierarchical clustering (HGMST-HC) by considering the characteristics and complexity of stereo vision. The indexing structure diagram of the proposed algorithm is shown in Fig. 4.

The overall architecture of the proposed indexing structure adopts the idea of hierarchical clustering. It is generally believed that the hierarchical clustering is a hierarchical decomposition to the given data set. According to the strategy, hierarchical clustering usually includes the methods of agglomerative and divisive. In this paper, the proposed indexing structure combines the above two hierarchical clustering methods. For indexing construction, it is divisive hierarchical clustering from the top-level to the low-level, while is agglomerative hierarchical clustering from the bottom to the top.

A. FEATURE OPTIMIZATION OF INFORMATION LEVEL

The shape and color features of binocular stereoscopic vision depth map and left image are optimized to form low-level leaf nodes of hierarchical indexing. The feature optimization process is shown in Fig. 5.

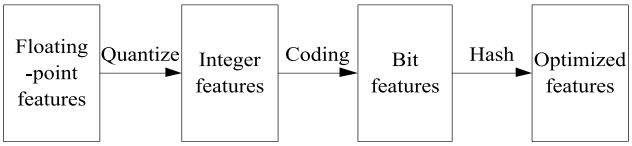

FIGURE 5. The feature optimization process.

1) FEATURE QUANTIZATION

The floating-point values of extracted shape and color features are quantized respectively. The quantized function is defined as:

$$T = \frac{F_{\max} - F_{\min}}{I}, \tag{3}$$

where $F_{\max}, F_{\min}$ are the maximum and minimum of the feature vectors in each image feature dimension, respectively, that is $F_{\max} = \max\{F[u]\}$, $F_{\min} = \min\{F[u]\}$, I is the feature dimension of the image, and $u = 0, 1, 2, \ldots, I$-1. The

quantized feature vector values can be expressed as:

$$F'[u] = int\left(\frac{F[u]}{T}\right). \tag{4}$$

The feature values can be quantized as a sequence of I integers in the range of [0, I-1], and in the experiment of the paper I is 20.

2) THE CODING OF BINARY FEATURES

Binary encoding is adopted to encode the quantized values of shape and color features extracted from depth maps and left images respectively. After the coding, the shape and color features are expressed as f_d, f_c respectively. The number of encoding figures is E, which is defined as:

$$E = \lceil \log_2 I \rceil \times I. \tag{5}$$

when the feature dimension is 20, each dimension is represented by 5 bits, that is, the encoding number of the features is 100 bits.

3) THE NORMALIZATION OF FEATURES

Considering the characteristics of depth map, color image as well as quantified features, on the one hand, the binary bit encoding formed by shape and color features is relatively complex and the number of encoding figures is large. On the other hand, the encoding sequence of depth map shape features is sparse, and there are many zero values after quantization. In order to achieve better classification and indexing of features, hash processing for shape and color feature coding is carried out. To maintain the greater similarity of the original adjacent feature data in the hash processing, some problems should be solved as far as possible. The high dimensional feature vectors can be mapped into a low dimensional hash feature space using efficient hash function, and the closer points in the original space should still in neighborhood after hashing [24]. At the same time, the collision probability of short distance points should be greater than those long distance points.

For the number determination of hash bit, to achieve uniform hash and reduce collision, the proposed algorithm takes into account the big data in the TB level. When the shape and color features of depth maps and left images are represented by two-dimensional coordinates, the order of magnitude for one dementional feature data is MB. So, according to the idea of hashing, the number of hash figures can be set to 3 bytes, which is 24 bits.

The feature values are hashed with the method of model calculation. The hash functions of shape and color features are defined respectively as:

$$H_d(f_d) = f_d \bmod 2^{24}, \tag{6}$$
$$H_c(f_c) = f_c \bmod 2^{24}. \tag{7}$$

The main purpose of hash processing lies in two aspects. The one is to transform the features into fixed length and dense binary short bit code data which can optimize the feature descriptors and facilitate matching calculation. The other is to achieve the first aggregation of feature values which is good for upper level indexing and can achieve query quickly.

B. FEATURE CLUSTERING OF LOCAL LEVEL

Cluster is widely used in many areas and the purpose is to divide a data set into a number of non-cross groups with the same characteristics based on the similarity. One of the advantages of clustering is to improve the effectiveness and efficiency of information retrieval. Usually, the retrieval only needs to match with specific clusters, not to traverse all the data in dataset [31]. In local level of the proposed indexing structure, the grid K-medoids clustering method is used to divide the feature points.

The values of shape feature H_d and color feature H_c are mapped to two dimensional coordinates which can be expressed as $P(x,y)$. The coordinate point similarity measure is defined as:

$$d(P_i, P_j) = \left[(x_i - x_j)^2 + (y_i - y_j)^2\right]^{\frac{1}{2}}, \tag{8}$$

where $P_i(x_i, y_i)$, $P_j(x_j, y_j)$ are feature points in grid region. By similarity calculation, the feature points are assigned to the nearest cluster centers, and the objective function is reduced. The objective function is defined as:

$$W = \sum_{i=1}^{w} \min_{t\in\{1,2,\ldots,K\}} \|P_i - C_t\|^2, \tag{9}$$

where w, K are the number of feature points and clustering subsets in each grid, P_i is the feature point, C_t is the cluster center point which is represented by the grid central point of clustering subset.

The evaluation criteria function of clustering performance is defined as:

$$E = \sum_{t=1}^{K} \sum_{p\in X_t} \|p - C_t\|^2, \tag{10}$$

where X_t is a clustering subset in the grid. The clusters are formed when the value of evaluation criterion function is the minimum.

In the grid K-medoids clustering, the number of grids is N^2 and the feature points are mapped into each grid according to the values [12]. The clustering is executed in every grid separately and each grid is divided into 9 cells, which is 9 sub-clusters and the value of K is 9 [17]. There are several advantages of the clustering method. Firstly, the value of K can be determined effectively and the clustering is more optimal and efficient. Furthermore, the errors caused by noise and outlier data can be avoided. Besides, there is no need to be re-clustered for a large number of points while the original nodes deleting and the new nodes inserting, and the upper indexing is also not need to change again, which effectively reducing the amount of calculation and the complexity of the algorithm. So, the construction of indexing is simple and

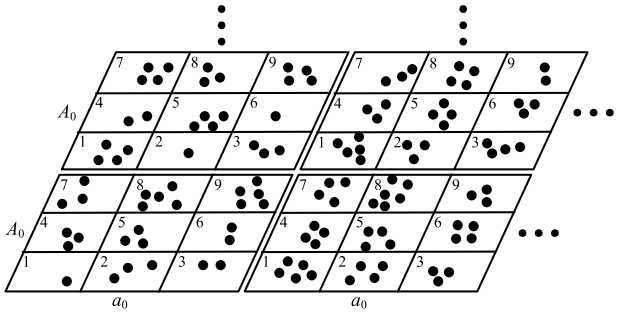

FIGURE 6. The feature clustering of local level.

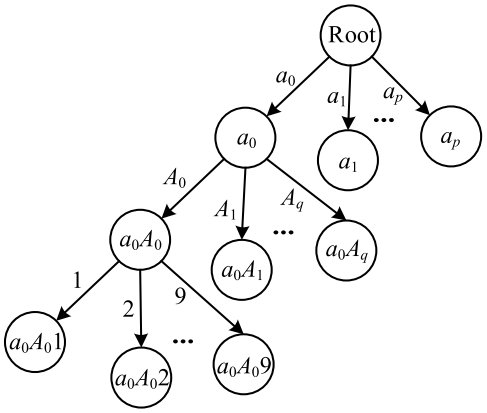

FIGURE 7. The suffix tree structure of clustering points.

efficient. In addition, the size of sub-cluster is reduced and the number is increased, which reduces the amount of calculation to retrieval and upper level indexing. The feature clustering of local level is shown in Fig. 6.

As a linear pattern matching algorithm, the suffix tree was first proposed by Weiner in 1973. It has been applied and developed in many algorithms, and it can achieve string matching and query effectively. In this paper, according to the principles of suffix tree and double suffix tree [17], the improved multiple suffix tree is constructed, in which the location area of feature point in the grid is tagged [18]. The grids in row and column are represented as a_0,a_1,a_2,...,a_p and A_0,A_1,A_2,...,A_q respectively. Each grid is divided into 9 sub-cluster regions, and each cell region is marked as 1,2,...,9 respectively. In the query, according to the top-level indexing, the tags in row and column are selected to get the adjacent grid, and then match the similar cell region to find matching points. For example, one of the adjacent cell regions can be represented as a_0A_18, and the similar matching point is in the clustering cell region, or in the other cell regions. The multiple suffix tree structure of clustering points is shown in Fig. 7.

In the proposed algorithm, there are two key problems to be solved.

1) THE SIZE AND NUMBER OF GRIDS

The maximum number of grids is relatively fixed, but in practice the grid is dynamically established or deleted according to the indexing distribution of feature points. Considering the existence or not of the grid region which belongs to the quantized two-dimensional feature points, and whether the grid region is empty, the grid is dynamically added or deleted. The advantage of this method is to avoid empty grid and reduce the waste of storage space. Of course, although there may be some sparse grids, which may increase memory consumption, it is still superior to the complex operation.

For the setting of grid size, to achieve query and operation simply, the scale of data represented by the grid is set to MB. The step of cell is $\phi = 1024$ and the size is 1024×1024, so the step of grid is $\varphi = 3\phi = 3072$. The adding of new grids and deleting of original grids may follow some ways. For a quantized 2D mapping feature point $P(x,y)$ to be inserted, the belonged grid of it is $a_{int\left(x/\varphi\right)}A_{int\left(y/\varphi\right)}$. Detecting the horizontal and vertical coordinates, if the grid is existing, then the new feature point is inserted and the clustering is executed. Otherwise, building a new grid and inserting the feature point. When a feature point is deleted, if there is nothing else in the grid, then deleting the grid and releasing the memory. If the figure is λ bits after hashing and encoding, and the maximum number of grid is N^2, so there can be the following definition:

$$N = \begin{cases} \left(\dfrac{2^\lambda}{\varphi}\right)^{\frac{1}{2}}, & \text{while integer} \\ int\left(\dfrac{2^\lambda}{\varphi}\right)^{\frac{1}{2}} + 1, & \text{else.} \end{cases} \tag{11}$$

2) COLLISION PROCESSING

The clusters of grid can be storaged in a dynamic array, the points are arranged in ascending order with distance from the cluster center. The most similar point is the first element of the array and the conflict points are stored after the first conflict node. The binary search is used in operation of querying, inserting or deleting and bi-directional detection is executed untile encounter a non-conflict node or the similarity is less than the threshold. Firstly, the method can avoid conflict effectively and realize better storage and operation of the feature points. Morever, as clustering implement is in cell regions, the number of clusters is increased while the feature poins reducing in each class, which improves the efficiency of ranking, querying, inserting and deleting. In addition, the clusters and grids can be indexed by the upper level, which won't reduce the efficiency of entire indexing operation as cluster increasing.

C. INDEXING TABLE CONSTRUCTION OF GLOBAL LEVEL

By establishing one dimension array indexing table, the grid index can be constructed in global level. The grid index needs to be associated with the feature points, and the grid position is calculated by using the feature points to get the center point. The upper index is established with the center point to generate indexing table which is expressed and stored through a pointer array. The purpose of the establishment of global indexing array is to reorganize the grid effectively and

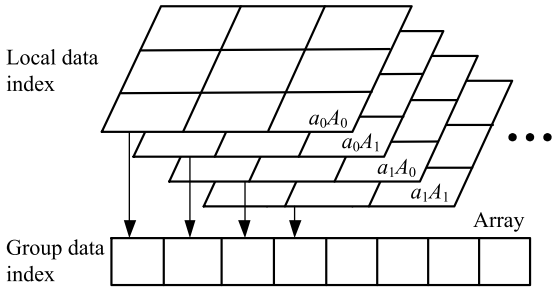

FIGURE 8. The global index by array.

improve the efficiency of query as the pointer points to each cluster center. At the same time, the clustering of features in upper level can be realized. As an example, for a grid a_pA_q, the indexing value is:

$$S[v] = \left(x_t^2 + y_t^2\right)^{1/2}, \quad v = 0, 1, 2, \ldots, N^2 - 1, \quad (12)$$

where $x_t = \frac{p+1}{2}\varphi$, $y_t = \frac{q+1}{2}\varphi$, $x_t, y_t \in Z_t(x_t, y_t)$, and Z_t is the center of the grid.

Each element of the array points to the center of the grids respectively. So, the array data sequence can be constructed to index the grids, and the global index by array is shown in Fig. 8.

The operations of global index are described as follows.

1) FOR INDEXING CONSTRUCTION

when inserting a point, the clustering level may need to add a new grid, and it is necessary to modify the global level index. So, a new indexing point will be put in global level, which forms the array element.

2) FOR INDEXING MODIFICATION

when a point is deleted, a grid in clustering level may become empty because of the deletion. So, it is necessary to remove the empty grid. Then the global level index should be modified and the indexing point should also be deleted. The array element can be deleted, which frees the memory.

3) FOR DATA QUERY

in global level, with group data index the query is executed on global values, then matching with the horizontal and vertical coordinates respectively. The horizontal and vertical encoding in grid can be acquired according to similarity matching to query the neighbor grid.

V. EXPERIMENTAL RESULTS AND ANALYSIS

A. EXPERIMENTAL ENVIRONMENT AND DATASET

We adopt Windows 7 with dual-core 32-bit processor as experimental environment, and the CPU clock speed is 3.3GHz. C++ programming language is used to simulate the proposed algorithm as well as the comparative algorithms, which are implemented in Visual Studio 2010. At present, there is no available benchmark dataset that is suitable for stereoscopic image feature indexing. So, we use the dataset composed of 5200 binocular stereoscopic images, which are key frames extracted from 500 binocular stereoscopic videos produced and established by China Art Science and Technology Institute and Communication University of China. These images are mainly about Chinese traditional cultures, such as drama opera, folk arts and dance, handicraft production, customs, cultural and physical landscape, cultural heritage, and so on. There are 10 categories, including Yang Opera 'Duan Taihou', cultural site 'Memorial Archway', national costume 'Cheongsam', Mongolian Drama 'Eternal Horse Eulogy', folk song 'Jingning Songs and Dance', local opera 'Tale of White Serpent', Kun Opera 'Fifteen Strings of Cash', drama 'Teahouse', folk custom 'Shengfang Folkways' and Xi Opera 'Pearl Tower'. The sample binocular stereoscopic images of the 10 categories in dataset are shown in Fig. 9. The algorithm HGMST-HC proposed in this paper is executed for feature indexing of these images. The experimental results are compared with the situation of without indexing structure, the adaptive suffix tree (AST) indexing structure algorithm [18] and the approximate nearest neighbor (ANN) indexing structure algorithm based on cluster locality sensitive hashing [23] to verify the effectiveness of the proposed algorithm.

B. MEMORY CONSUMPTION OF INDEXING CONSTRUCTION

For the indexing structure construction of stereoscopic image features with different algorithms, it will require the memory consumption in execution, which indicates the complexity of indexing construction. When the number of images is 5200, the memory consumption of indexing construction with different algorithms is shown in Fig. 10. As the index is built, most of the dataset files are leaf nodes located in bottom level in the proposed algorithm, so there is less memory consumption. We can see from the Fig. 10, the memory consumption is 0MB when there is no index to be built, while compared with AST algorithm and ANN algorithm, the proposed algorithm HGMST-HC in this paper has less memory consumption, and decreasing by 28.82% and 45.02% respectively.

C. TIME CONSUMPTION OF INDEXING CONSTRUCTION

For the images of dataset in the experiment, different algorithms are executed to construct indexing structure respectively. The time consumption of each algorithm in feature indexing construction is shown in TABLE 1. We can see from the TABLE 1, except for the situation of without index, the proposed algorithm HGMST-HC in this paper has less time consumption compared to the AST algorithm and ANN algorithm, reducing by 58.34% and 50.91% respectively. The advantage of time efficiency is mainly due to the fact that it does not need to be re-clustered when an indexing data node is inserted.

D. THE INDEXING PERFORMANCE AND QUERY RESULTS

The query efficiency is usually used to verify the indexing performance, which can be measured with recall and

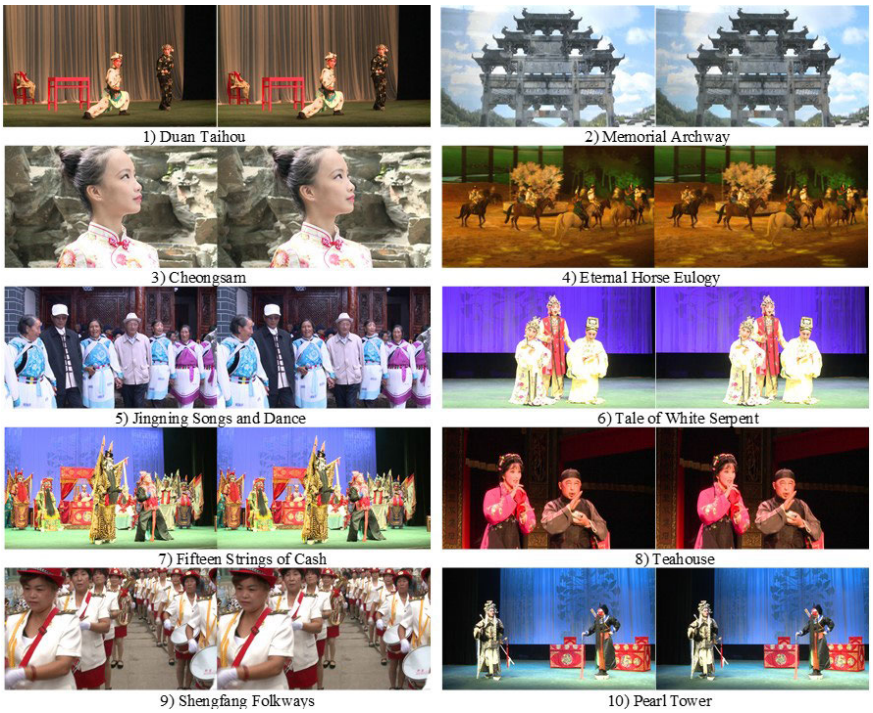

FIGURE 9. The sample binocular stereoscopic images of 10 categories.

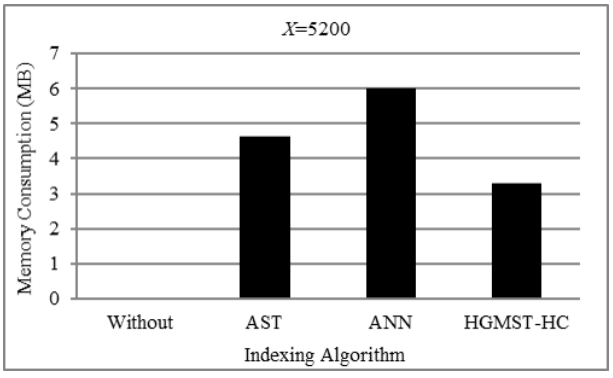

FIGURE 10. The memory consumption of indexing construction with different algorithms.

TABLE 1. The time consumption of each algorithm in feature indexing construction.

Algorithm	Time Consumption (s)
Without	0
AST	7.73
ANN	6.56
HGMST-HC	3.22

matching time. The higher recall and the less query time, the query efficiency is better. Generally, the construction of indexing structure will reduce the recall. So, if the recall decreases less for an algorithm, then the indexing structure is more superior. As the images indexed in experiment are extracted from 500 stereoscopic videos, we verify the query accuracy and efficiency for binocular stereoscopic videos. For stereoscopic video clips of each category, each one is selected as query sample for matching [32], and then

FIGURE 11. A query example of binocular stereoscopic video image.

calculating the average recall and average matching time. The average recall and average matching time can be considered as the mean average recall (*MAR*) and mean average matching time (*MAMT*) for all videos in database. The *MAR* and *MAMT* are defined respectively as:

$$MAR = \frac{1}{Q}\sum_{i=1}^{Q} \mathrm{Re}call\,(i), \tag{13}$$

$$MAMT = \frac{1}{Q}\sum_{i=1}^{Q} Time\,(i), \tag{14}$$

where Q is the number of binocular stereoscopic videos in matching database, $\mathrm{Re}call(i)$ and $Time(i)$ are the recall and matching time of the $i-$ th video. For each category in query, the $\mathrm{Re}call(i)$ can be defined as:

$$\mathrm{Re}call(i) = \frac{Correct}{Correct + Missing} \times 100\%. \tag{15}$$

The *MAR* and *MAMT* of the indexing structures for each algorithm are shown in TABLE 2. From TABLE 2, we can see all the indexing structure algorithms of AST, ANN and HGMST-HC can greatly reduce the matching and retrieval time to improve the query efficiency in overall perspective.

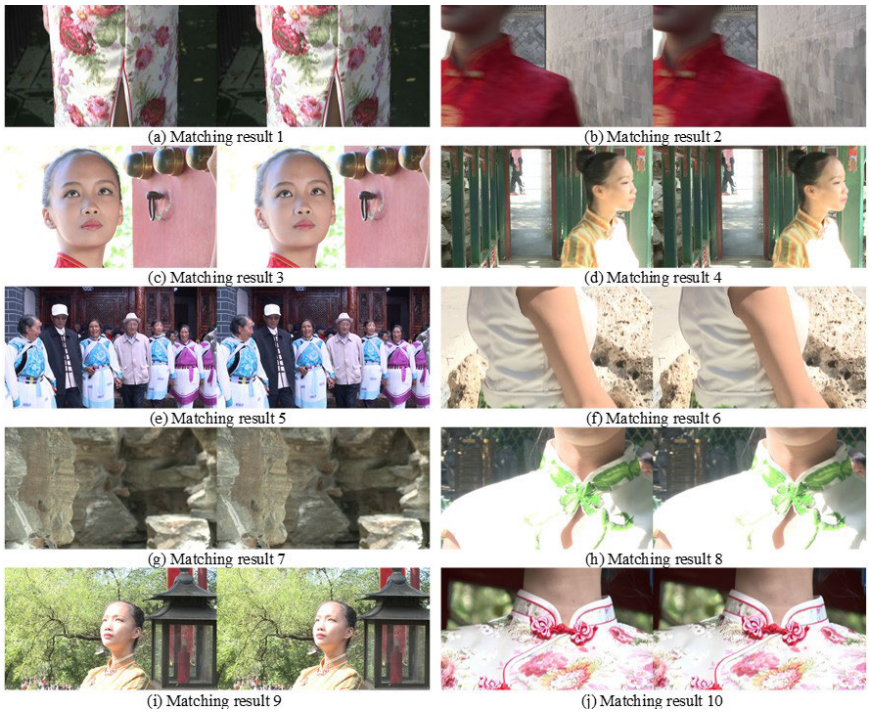

FIGURE 12. The matching results of binocular stereoscopic video image with HGMST-HC indexing structure algorithm.

TABLE 2. The *MAR* and *MAMT* of the indexing structures for each algorithm.

Algorithm	*MAR* (%)	*MAMT* (s)
Without	93.83	4.04
AST	80.65	1.85
ANN	82.71	1.69
HGMST-HC	91.09	0.98

The query time of the algorithm AST is reduced by 54.21% and the recall decreases 13.18%. For the ANN algorithm, the time is reduced by 58.17% and the recall decreases 11.12%. While for the HGMST-HC algorithm proposed in this paper, the time is reduced by 75.74%, but the recall only decreases 2.74%. So, it indicates that the indexing structure of the proposed algorithm HGMST-HC is obviously better than the others.

The algorithm proposed in this paper is implemented to the matching and retrieval for binocular stereoscopic videos in database. The first key frame images of query video and the matching results with HGMST-HC indexing structure algorithm are shown in Fig. 11 and Fig. 12 respectively. By checking the matching results and the contents in the database, all the returned binocular stereoscopic videos belong to the same category in content or semantics with the query video except for the 5th. It is less relevant to the category of query video and does not belong to it formally. But with regard to the analysis on visual content, the main reason of the 5th being returned as matching result is that it has some similarity to the query video on visual shape of person and background. Therefore, the application of proposed indexing structure algorithm HGMST-HC can ensure higher query accuracy.

VI. CONCLUSION

In order to achieve retrieval of binocular stereoscopic image or video efficiently, a feature indexing structure algorithm based on hybrid grid multiple suffix tree and hierarchical clustering is proposed in consideration of the structural characteristics. The binocular stereoscopic image indexing is realized efficiently with feature dimensional reduction. The experimental results show that the algorithm has obvious advantages. The query efficiency is improved effectively when the index is constructed according to the characteristics of binocular stereoscopic image. With the advantages of clustering and hierarchical indexing, the computational complexity is low for node deletion and insertion, which improves the scalability of indexing structure and achieves optimization. In case of the volume of big data, the proposed algorithm of indexing structure is better than the other existing typical algorithms. And the efficiency decreases less with the amount of data increases. It can also provide some references and supports for multi-view stereoscopic feature indexing and content-based retrieval. Nevertheless, we will make further efforts to improve the algorithm in the future. On the one hand, the feature extraction and indexing for large-scale database files will be optimized to reduce complexity. On the other hand, the parallel processing for indexing construction and query will be applied to improve the retrieval speed

and efficiency. In addition, we will consider using GPU to accelerate the process of image feature indexing.

REFERENCES

[1] N. Zeng, H. Hao, and H. Zheng, "Efficient stereo index technology for fast combination query of electric power big data," in *Proc. 1st IEEE Int. Conf. Comput. Commun. Internet (ICCCI)*, Oct. 2016, pp. 329–333.

[2] E. Tiakas, D. Rafailidis, A. Dimou, and P. Daras, "MSIDX: Multi-sort indexing for efficient content-based image search and retrieval," *IEEE Trans. Multimedia*, vol. 15, no. 6, pp. 1415–1430, Oct. 2013.

[3] B. Luo, S. Jiang, and L. Zhang, "Indexing of remote sensing images with different resolutions by multiple features," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 4, pp. 1899–1912, Aug. 2013.

[4] P. Schuch, "Survey on features for fingerprint indexing," *IET Biometrics*, vol. 8, no. 1, pp. 1–13, Jan. 2019.

[5] B. Miller and S. McCloskey, "Metric feature indexing for interactive multimedia search," in *Proc. IEEE 13th Conf. Comput. Robot Vis. (CRV)*, Jun. 2016, pp. 109–115.

[6] Z. Xu, J. Du, L. Ye, and D. Fan, "Multi-feature indexing for image retrieval based on hypergraph," in *Proc. 4th Int. Conf. Cloud Comput. Intell. Syst. (CCIS)*, Aug. 2016, pp. 494–500.

[7] K. Kikuchi, K. Ueki, T. Ogawa, and T. Kobayashi, "Video semantic indexing using object detection-derived features," in *Proc. IEEE 24th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2016, pp. 1288–1292.

[8] D. Song and J. Feng, "Fingerprint indexing based on pyramid deep convolutional feature," in *Proc. IEEE Int. Joint Conf. Biometrics (IJCB)*, Oct. 2017, pp. 200–207.

[9] K. Liao, H. Lei, Y. Zheng, G. Lin, C. Cao, M. Zhang, and J. Ding, "IR feature embedded BOF indexing method for near-duplicate video retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 12, pp. 3743–3753, Dec. 2019.

[10] T. A. Pham, V. H. Le, and D. N. Le, "A review of feature indexing methods for fast approximate nearest neighbor search," in *Proc. IEEE 5th NAFOSTED Conf. Inf. Comput. Sci. (NICS)*, Nov. 2018, pp. 372–377.

[11] Y. Yang, J. Wu, J. Fang, and Z. Cui, "Parallel hierarchical k-means clustering-based image index construction method," in *Proc. IEEE 11th Int. Symp. Distrib. Comput. Appl. Bus., Eng. Sci.*, Oct. 2012, pp. 424–428.

[12] T. W. Chiang, T. Tsai, and M. J. Hsiao, "A hierarchical grid-based indexing method for content-based image retrieval," in *Proc. IEEE 3rd Int. Conf. Intell. Inf. Hiding Multimedia Signal Process. (IIH-MSP)*, vol. 1, Nov. 2007, pp. 206–209.

[13] S. Kiranyaz and M. Gabbouj, "Hierarchical cellular tree: An efficient indexing scheme for content-based retrieval on multimedia databases," *IEEE Trans. Multimedia*, vol. 9, no. 1, pp. 102–119, Jan. 2007.

[14] H. Xu, D. Xu, and E. Lin, "An approach of hierarchical image index based on subspace cluster," *J. Image Graph.*, vol. 14, no. 1, pp. 142–147, Jan. 2009.

[15] E. T. Khalaf, M. N. Mohammad, and K. Moorthy, "Robust partitioning and indexing for iris biometric database based on local features," *IET Biometrics*, vol. 7, no. 6, pp. 589–597, Nov. 2018.

[16] N. Chen, X. Zhong, and L. Li, "Research on optimized R-tree high-dimensional indexing method based on video features," in *Proc. IEEE 2nd Int. Conf. Cloud Comput. Big Data Anal. (ICCCBDA)*, Apr. 2017, pp. 93–97.

[17] Y. Lo and C. Wang, "Hybrid multi-feature indexing for music data retrieval," in *Proc. 6th IEEE/ACIS Int. Conf. Comput. Inf. Sci. (ICIS)*, Jul. 2007, pp. 543–548.

[18] U. Gunasinghe and D. Alahakoon, "The adaptive suffix tree: A space efficient sequence learning algorithm," in *Proc. IEEE Int. Joint Conf. Neural Netw. (IJCNN)*, Aug. 2013, pp. 1–8.

[19] F. Liu, Y. Zhuang, F. Wu, and Y. Pan, "3D motion retrieval with motion index tree," *Comput. Vis. Image Understand.*, vol. 92, nos. 2–3, pp. 265–284, Nov. 2003.

[20] Y. Feng, L. Fan, and Y. Wu, "Fast localization in large-scale environments using supervised indexing of binary features," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 343–358, Jan. 2016.

[21] Z. Deng, L. Wang, W. Han, R. Ranjan, and A. Zomaya, "G-ML-Octree: An update-efficient index structure for simulating 3D moving objects across GPUs," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 5, pp. 1075–1088, May 2018.

[22] R. Xu and D. C. Wunsch, "Survey of clustering algorithms," *IEEE Trans. Neural Netw.*, vol. 16, no. 3, pp. 645–678, Jun. 2005.

[23] X. Xu, W. Geng, R. Ju, and Y. Yang, "OBSIR: Object-based stereo image retrieval," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2014, pp. 1–6.

[24] T. Ahmed and M. Sarma, "Locality sensitive hashing based space partitioning approach for indexing multidimensional feature vectors of fingerprint image data," *IET Image Process.*, vol. 12, no. 6, pp. 1056–1064, Jun. 2018.

[25] S. Ramaswamy and K. Rose, "Adaptive cluster distance bounding for high-dimensional indexing," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 6, pp. 815–830, Jun. 2011.

[26] F. Duan, "Consistent depth maps estimation from binocular stereo video sequence," *J. Shanghai Jiaotong Univ., Sci.*, vol. 21, no. 2, pp. 184–191, Apr. 2016.

[27] F. Duan, Y. Wang, L. Yang, and S. Pan, "Feature extraction for stereoscopic vision depth map based on principal component analysis and histogram of oriented depth gradient," *J. Comput. Appl.*, vol. 36, no. 1, pp. 222–226, Jan. 2016.

[28] F. Duan, "An automatic extraction method for binocular stereo colour vision image," *Electrotehn., Electron., Autom.*, vol. 65, no. 4, pp. 168–176, Dec. 2017.

[29] X.-N. Zhang, J. Jiang, Z.-H. Liang, and C.-L. Liu, "Skin color enhancement based on favorite skin color in HSV color space," *IEEE Trans. Consum. Electron.*, vol. 56, no. 3, pp. 1789–1793, Aug. 2010.

[30] L. Jiang, G. Shen, and G. Zhang, "An image retrieval algorithm based on HSV color segment histograms," *Mech. Elect. Eng. Mag.*, vol. 26, no. 11, pp. 54–57, Nov. 2009.

[31] P. Gupta and A. K. Sharma, "A framework for hierarchical clustering based indexing in search engines," in *Proc. IEEE 1st Int. Conf. Parallel, Distrib. Grid Comput. (PDGC)*, Oct. 2010, pp. 372–377.

[32] F. Duan and S. Duan, "Stereoscopic video clip matching algorithm based on incidence matrix of similar key frames," *3D Res.*, vol. 9, no. 2, pp. 1–12, Jun. 2018.

FENGFENG DUAN received the B.S. degree in computer science from Anhui Normal University, in 2007, the M.S. degree in computer science from Huazhong Normal University, in 2010, and the Ph.D. degree in computer science and technology from the Communication University of China, in 2016. He is currently an Associate Professor with the School of Journalism and Communication, Hunan Normal University, Changsha, China. He is also a Researcher with the Hunan Social Public Opinion Monitoring and Network Public Opinion Research Center, Changsha. His research interests include computer vision, image processing, and content-based retrieval.

QICONG ZHANG received the B.E. degree in computer science from the Zhangjiakou Communication College of PLA, in 2002, and the M.E. degree in computer science from Shandong University, in 2011. He is currently pursuing the Ph.D. degree with the School of Computer Science, Communication University of China. He is also an Associate Professor with Shandong Management University, Jinan, China. His research interests are mainly in image processing and storage, and content-based retrieval.