The Design of target image retrieval algorithm based on Hadoop cloud platform

YuanYong  Chen
Research Institute of Electronic Science and Technology
University of Electronic Science Technology of China
Chengdu, China
Email: wgchenyuanyong@163.com

Abstract - With the rapid development of computer technology, information and network technology, the traditional retrieval model based on keywords and text description can not meet the needs of mass image retrieval. Therefore, content-based image retrieval has become a hot research topic at home and abroad in recent years. Content-based image retrieval CBIR (Content Based Image Retrieval) is a database to retrieve the target image with similar images from the image itself, the content of manpower. Therefore, with the help of Hadoop and LBP algorithm of texture based image retrieval, image retrieval from large image target with similar characteristics is the focus of current research.

Keywords- Image retrieval, texture, Hadoop, LBP

I. INTRODUCTION

The texture of the concept has its origins in human touch -- "of a thing or object on the surface of the touch", though most of us recognize texture is no problem, but today, is still difficult to give texture of a unified and accurate can be recognized by the definition. Just roughly the distribution pattern of pixel gray level image texture definition pairs in the space of the description. The texture can be considered as a pattern that is produced by the change of the space according to a certain rule. It is one of the inherent characteristics of the real image area.

Texture is an important feature in content-based image retrieval, and the effective use of this feature is of great significance for image retrieval.

II. SYSTEM FRAMEWORK AND MODULE DESIGN

Hadoop based image retrieval system is designed to achieve massive, heterogeneous, distributed image resources for the rapid retrieval and timely response. System uses a distributed architecture, which is composed of the upper and lower layers, the business logic layer, data and data processing layer, the overall framework as shown below[1].

A. HDFS module design

HDFS cluster has two types of nodes, the use of Master/Slave architecture, to managers - worker mode operation, that is, a NameNode (Management) and a number of DataNode (workers) components. When the user sends a request to the file to read and write operation, the cluster through the NameNode and DataNode interaction to achieve the file read and write operation. HDFS frame structure as shown below[4]:

Figure 1. Architecture of object image retrieval system based on Hadoop

Figure 2. HDFS framework design

The HDFS namenode is the core, user management data node and client access to files, file system namespace, maintenance the entire file system data structures that record and preservation system files and metadata. These information to backup files stored in the set of nodes on the namenode, Hadoop on multi copy of the data block is automatically maintained, when the clusters in a node data loss caused by the failure of the mission. Namenode will automatically re deploy computing tasks. When Hadoop starts, collecting data blocks of information through DataNode.

Datanode is file system node, is responsible for the storage and retrieval of data block according to needs, the storage location of the data block adjustment with Hadoop system change, datanode cycle aptitude namenode reported heartbeat. When a client sends a request, the namenode monitoring program is open, when to detect a client sends a request, the namenode monitoring program is opened, when the detection to the client request. Namenode
will HDFS distributed system directory information such as data, using Hadoop program instructions for local data processing, HDFS data import and export, the corresponding operation.

B. MapReduce module design

MapReduce module design is mainly used for large-scale data set of parallel computing, in this system is mainly responsible for calculation in the process of the image matching and similarity measure in image retrieval, and the matching results in accordance with the similarity from big to small order returned to the user. MapReduce module to achieve the work of parallel computing block diagram as shown below[6]:

![MapReduce work block diagram](image)

Figure3. Implementation of parallel computing MapReduce work block diagram

First, MapReduce programs running, on HDFS image feature database were data segmentation, image feature database sharding, then by the datanode node will each data transmission to each tasktracker node operation, by the map read data slice, the data into the key / value.Form of data characteristics, including key key said image features in the data points of the offset distance, the key value for image feature values.

Then (map) function calls these key / value value with a radius of N. The value of 3 pixels around the two * 3 neighborhood is multiplied by the value of the corresponding value of the weight matrix. The 8 values are summed up and the decimal number is obtained for the LBP eigenvalue of the 3 * 3 neighborhood.

<table>
<thead>
<tr>
<th>$g_1$</th>
<th>$g_2$</th>
<th>$g_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$g_4$</td>
<td>$g_5$</td>
<td>$g_6$</td>
</tr>
<tr>
<td>$g_7$</td>
<td>$g_8$</td>
<td></td>
</tr>
</tbody>
</table>

Figure5. 3x3 Matrix

In order to improve the expression ability of basic LBP features, the former is extended to the basic characteristics of LBP, calculation of LBP features is no longer confined to the 3 * 3 neighborhood, but set a sampling radius R and the sampling points in P, and in the center as a center point sampling P and ulnar radius circumference at intervals. The gray P points the value of binarization and gray value by comparing the size of the center point, the commonly used R and the corresponding P as shown in Figure 5. The P-8, R=1 as an example, to illustrate the computational process of the extended LBP features. The center pixel gray hypothesis value is $g_c$, the gray 8 sampling the pixel values are respectively $g_1, g_2, ..., g_8$, the surrounding area is characteristic of the center pixel calculation formula is as follows:

![Image](imagedata)

Figure6: Improved LBP

$s(g_c,g_i) = \begin{cases} 1, & g_i \geq g_c \quad (1 \leq i \leq 8) \\ 0, & g_i < g_c \end{cases}$  \hspace{1cm} (1)

Two values of 8 neighboring pixels, in accordance with the order of the number of pixels to the center pixel LBP value, calculation method:

$LBP = \sum_{i=1}^{8} s(g_c,g_i)2^{i-1}$  \hspace{1cm} (2)

The LBP value is only related to the difference between the center and the adjacent points, and the description of the texture feature is shift invariant. There are 256 possible values of LBP value between 0-255, which reflects the different texture pattern information.

IV. LBP FEATURE EXTRACTION

A. LBP Function

Function prototype is: $J=$LBP ([I, R, N, MAPPING, MODE,]).

This function returns the local two value pattern of the image I encoding the image or the local two value mode intensity histogram. This value is calculated by using the MAPPING table defined by the R table to calculate the LBP encoding by sampling the circle with a radius of N. Parameter MODE has two forms:

- ‘h’:Return the number of different encoding modes, i.e., the image of the local two value pattern.
- ‘nh’:Return the percentage of each encoding mode, i.e., the local two value mode intensity histogram.
B. GETMAPPING mapping function

Function prototype is: \( \text{MAPPING} = \text{GETMAPPING} (\text{SAMPLES}, \text{MAPPINGTYPE}) \).
Return a LBP encoding mapping table in the neighborhood of the SAMPLES sampling point.
Parameter MAPPINGTYPE has four forms:
- 'u2': Return to the MAPPING table is mapped to the uniform LBP, after a total of 59 model mapping.
- 'ri': The returned MAPPING table is mapped to the invariant LBP rotation, which is a total of 36 models.
- 'riu2': The returned table is mapped to the LBP, and a total of 10 models are mapped
- '0': Parameter 0 indicates that the basic LBP encoding is not mapped, that is, to maintain the basic LBP code, a total of 256 modes.

C. Image block histogram

The images are divided into different blocks under the same dimension and divided into different blocks in different dimensions[3].
Function prototype for: \( T = \text{fenkuai} (S, N_1, N_2) \)
The matrix \( S \) is divided into \( n_1 \times n_2 \) and other molecular matrix, using the output \( j \), \( i \) is the first \( j \). \( T[i \text{ sub matrix} \)

\[
\text{function } T = \text{fenkuai}(S, n_1, n_2) \\
[M, N] = \text{size}(S); \\
M_1 = M/n_1; \\
N_1 = N/n_2; \\
T = \text{cell}(n_1, n_2); \\
\text{for } i = 0:(n_1-1) \\
\text{for } j = 0:(n_2-1) \\
\quad T{i+1,j+1} = S((i*M_1+1):(i+1)*M_1),(j*N_1+1):(j+1)*N_1); \\
\text{end} \\
\text{end} \\
\]

By dividing an image into a number of small blocks, in each block to extract features and then generate a histogram of the image.

D. Generated histogram function

\( \text{HH} = \text{hist} (h_1,0:15) / (kk*ff) \) this function is to calculate the percentage of each feature is calculated out of the following graphics[5].

![Generated histogram](image)

V. MATCHING ALGORITHM

Through statistics in different image LBP probability. Through this can get the image texture spectrum histogram (i.e., LBP descriptor), take different images of the mode probability difference absolute value and as the image similarity measure, this value is more and more small image similarity is high, whereas the known image of; considering the influence of blocks of the image retrieval performance. Therefore, the image is divided into 2 x 2 block, 3 x 3 block, 4 x 4 block used by specific redo the above experiments.[7]

![Texture image retrieval system](image)

VI. EXPERIMENTAL RESULT

We choose a fixed image as the source of the database, retrieve similar images in the database, and achieve the results in order to retrieve the similarity of the size of the array. In the search results display interface, the first line of the left image is to retrieve the image, the other images by the similarity of the source and the retrieval, from the most similar to the order of the order of similarity[8].

a) Non block search results

![Non block search results](image)

b) 2 x 2 block search results

![2 x 2 block search results](image)

c) 4 x 4 block search results

![4 x 4 block search results](image)
Ⅶ. EXPERIMENTAL RESULT ANALYSIS

In the experiment when the 8 * 8 block is retrieved, the experimental results have been very accurate, from a large number of experimental results can be obtained in a certain range block is more refined search results more accurate. From the above several LBP descriptor retrieval methods, on the same block LBP basic retrieval more accurate, but basic LBP descriptor dimension higher retrieval efficiency is low. Other LBP descriptors reduce feature dimension, but also the loss of feature information retrieval. In the results of the experimental results can be found in all the images in line with the results of the image, the overall experimental results are good.

Ⅷ. CONCLUSION

Through the research and analysis of image retrieval system model, based on the Hadoop cloud platform, the image retrieval system based on texture feature is designed and developed. The system realizes the classical algorithm LBP.

REFERENCE