Eng Int Syst (2022) 1: 23–33

Engineering
Intelligent Systems

Lossless Compression Algorithm of Multimedia Data Based on Artificial Intelligence

Quanpeng Ji*

College of Artificial Intelligence, Chongqing University of Arts and Sciences, Yongchuan 402160, China

In order to improve the storage efficiency of multimedia data in a multi-thread rule linked list, efficient data compression is needed. A lossless compression algorithm for multimedia data based on artificial intelligence is proposed. A similar cache structure model of multi-thread rule linked list multimedia data is constructed, a distributed cloud computing method is adopted to mine the autocorrelation characteristics of the multi-thread rule linked list multimedia data, high-dimensional statistical information of the multi-thread rule linked list multimedia data is extracted, a phase space reconstruction method is combined to reconstruct the characteristics of the multi-thread rule linked list multimedia data, a knowledge map technology is used to realize the integration and fusion of main components of the multi-thread rule linked list multimedia data, and an artificial intelligence learning method is adopted to realize the lossless compression optimization of the multimedia data. The simulation results show that the load of lossless compression of multimedia data in the multi-threaded rule linked list is large, the storage capacity is improved, and the real-time ability to access data is good. It has good application value in multi-threaded rule linked list multimedia data mining and storage structure optimization.

Keywords: Artificial Intelligence, Multimedia Data, Lossless Compression, Storage.

1. INTRODUCTION

In the information society, a large volume of multimedia data needs to be intelligently stored and scheduled through the network space, multimedia data lossless compression and storage models of multi-threaded rule linked lists are designed, cloud storage databases are established, and the intelligent storage capacity of multimedia data of multi-threaded rule linked lists is improved (Hashemi and Yang, 2009). The related data storage methods have received great attention. The storage of multi-thread rule linked list multimedia data mainly utilizes a local database and a network database. In the process of database storage and scheduling, intelligent selection and distributed network planning and design are required. An intelligent storage structure model of multi-thread rule linked list multimedia data is established to realize intelligent storage and transmission control of data. Intelligent selection and storage method of multi-thread rule linked list multimedia data is of great significance in optimizing the storage structure and improving space capacity (Ju and Zou, 2015).

Lossless compression of multi-threaded rule linked lists for multimedia data is realized through the extraction and mining of optimized features of data, combined with an embedded storage space planning method, and the optimized storage of the database. In traditional methods, lossless compression of multi-threaded rule linked lists for multimedia data mainly uses the decision tree planning method (Lyu et al., 2016), the PSO algorithm and genetic algorithm, combined with the intelligent optimization control method. To improve these methods, a lossless compression optimization model of multi-threaded rule linked list based on distributed structure adaptive screening control is proposed in this document. The distributed structure adaptive screening method is used to schedule the extracted data information flow and allocate storage space to improve the balance of data storage.

*Corresponding Author e-mail: jqpeng1978@126.com

However, this method is computationally expensive and has poor real-time performance (Chen and Li, 2017; Qiao and Sun, 2008; Huang and Liu, 2016). This document proposes a lossless compression method for multimedia data based on multi-threaded rule linked lists based on correlation dimension feature analysis and maximum Lyapunov exponential spectrum decomposition, which improves the throughput of lossless compression for multimedia data of multi-threaded rule linked lists, but this algorithm has poor balance and anti-interference (Sun et al., 2014). To solve the above problems, this paper proposes a lossless compression method of multimedia data based on artificial intelligence and multi-thread rule linked list. Firstly, a similar cache structure model of multi-thread rule linked list multimedia data is constructed (Zhou et al., 2018). Distributed cloud computing method is then used to mine the autocorrelation features of the multi-thread rule linked list multimedia data, and high-dimensional statistical information of the multi-thread rule linked list multimedia data is extracted. Phase space reconstruction method is then combined to reconstruct the features of multi-thread rule linked list multimedia data. Knowledge map technology is used to realize the integration and fusion of the main components of the multi-thread rule linked list multimedia data, thus realizing lossless compression optimization. Finally, a simulation experiment is carried out to demonstrate the superior performance of the method in improving the lossless compression capability of multi-threaded regular linked lists for multimedia data (Syed and Mamun, 2019; Anand and Sundararajan, 2020; Tan et al., 2019).

2. BASIC DEFINITIONS

2.1 The Compression Type of Multimedia Data

Compression algorithms are designed for various redundancies in multimedia data, that is, using certain coding methods to eliminate redundant information, to achieve undistorted compression, or based on the physiological characteristics of human vision and hearing, distortion compression within the allowable distortion limit to obtain a higher compression ratio. These redundancies can be summarized into the following categories.

2.1.1 Time Redundancy

Time redundancy refers to the fact that the processor utilization is less than a certain limit when the real-time system is running normally, which makes the processor have enough idle time. When the real-time task is running in error, it can use these idle time to realize fault-tolerant operation. As there is an obvious spatiotemporal correlation in the image sequence, most video encoders use 2D DCT to obtain good compression efficiency. The first stage utilizes the method of inter-frame time redundancy, and the second stage utilizes the coding method of intra-frame space redundancy.

2.1.2 Space Redundancy

Spatial redundancy refers to the fact that the same object or background (or parts of it) in the objective world generally has approximately the same surface physical properties, which is reflected in the image as having a certain gray correlation (redundant bits). For the same bit stream in the sampled image, audio, or video data, using the coding mechanism to replace the repeated bit sequence with repeated values can save considerable data capacity. For example, a melanin is followed by 20 white pixels, so all 20 white pixels don't need to be stored, only the number of white pixels. Considering the hardware and software implementation, choosing 8×8 block sizes does not generate significant memory requirements, and the computational complexity of 8×8 DCT is manageable on most computing platforms.

In addition, the size of 8×8 blocks can provide remarkable good compression efficiency. Since the spatial correlation is significantly reduced when there are more than 8 pixels near one pixel, reducing the compression efficiency, blocks larger than 8×8 cannot provide acceptable compression efficiency. As DCT is an orthogonal transformation, the orthogonality of the transformation can greatly reduce the complexity of coding and decoding. For highly correlated image data DCT will provide high compression efficiency.

Spatial redundancy and time redundancy are the statistical characteristics reflected when the image signal is regarded as a probabilistic signal, which is also called statistical redundancy.

2.1.3 Structural Redundancy

Some images have very strong texture structures from the region they are taken from, such as grass mat images, ocean wave images and so on. Some small local image patterns in these images appear repeatedly in the whole image, and even have certain arrangement rules.

2.1.4 Visual Redundancy

A camera can capture more information than a human eye can differentiate, resulting in "invisible" information in the multimedia object. This information can be removed without affecting the quality of the image as the observer is unable to perceive it.

2.1.5 Knowledge Redundancy

Most of the images are two-dimensional images of objects that exist objectively in nature, and their regional structure, gray level change and other attributes are subordinate to the laws of nature, and also accord with the knowledge structure of human beings. For example, when a face image is seen, it is already known that the human face must have symmetrically distributed mouth, eyes, mouth, nose and ears and their respective approximate position and shape.

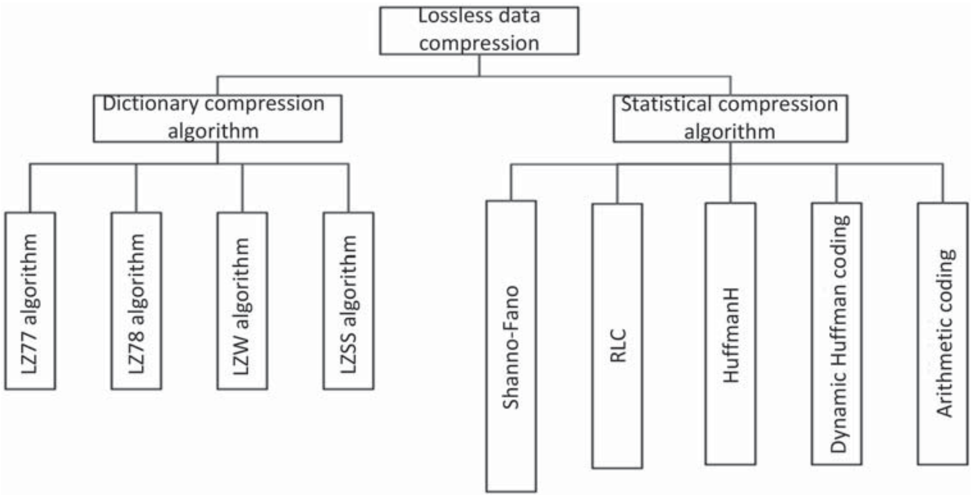

Figure 1 Classification of commonly used lossless data compression.

Table 1 Source symbol and its probability.

Source Symbol	**A**	**B**	**C**	**D**	**E**	**F**	**G**
Probability	0.20	0.19	0.18	0.16	0.14	0.10	0.01

2.2 Classification of Lossless Data Compression

Lossless compression technology is commonly referred to as general-purpose compression technology, it is also called information-keeping coding, entropy coding, distortion-free coding and so on, that is, according to the certain method of coding of a large volume of data to achieve information compression stored process, in the process of data compression that does not allow the loss of accuracy, the compressed data should be able to be decoded back to the original state of compression. Mainly used for the compression of text files, databases, program data, and image data (such as fingerprint images, medical images, etc.) for special applications. The compression ratio of such algorithms is low, generally from 1/2 to 1/5.

Non-destructive compression is a necessary choice when a compressed object requires accurate data such as text or numbers. The lossless compression model can be divided into statistical compression algorithm and dictionary-based compression algorithm and the specific classification map is shown in Figure 1.

2.2.1 Statistical Compression Algorithm

Based on the early origins of the statistical compression algorithm, it is essentially the occurrence frequency of statistical characters being used to re-encode the characters themselves. This method belongs to the entropy encoding class, which is independent of the order of the original data and is related to its occurrence frequency. The main statistical compression algorithms are Shanno-Fano coding, run-length coding (RLC), Huffman coding and arithmetic coding. Taking Hoffman coding as an example, the basic idea of this method is to arrange the source symbols in probability descending order, combine the two node probabilities with the least probability into one new probability, rearrange and merge the new probability and the remaining other probabilities in descending order until it reaches one node, and the probability value of that node is 1. The symbol with the largest probability is represented by 0, and the symbol with smallest probability is marked by 1 until the source symbol is reached. The marked "0" and "1" are then sorted from the root node to the source symbol to get the symbol sequence of each symbol. From the above coding ideas, we can see that the smaller the probability, the longer the symbol code is, otherwise, the shorter code total length must be less than the length of the source file coding. The operation steps of Huffman coding are illustrated with examples. The source symbols and probabilities are shown in Table 1.

2.2.2 Dictionary Based Compression Algorithm

Dictionary coding methods are similar to dictionary coding. Its basic principle is to form the various entries in the dictionary with a longer string or a frequent combination of letters, and the method of expressing them with a relatively short number or symbol. The compression effect is related to the appearance of repeated data and the size of the dictionary. The methods mainly include: LZ77 algorithm, LZSS algorithm, LZ78 algorithm, LZW algorithm and other basic algorithms. For example, LZW coding, is a dictionary-based compression algorithm that produces a string table while compressing the data, representing the actual data through symbols or code in the string table to reduce the amount of data. Dictionary compression algorithm is used when the data in the source has the characteristic of repeatability.

According to the principles described above, the specific steps of LZW coding are as follows:

Step 1: The dictionary is initialized to contain all possible monocharacters and the current prefix P is null.

Table 2 String table initialization.

Alphabetic string	Index
a	0H
b	1H
LZW_CLEAR	2H
LZW_EOI	3H

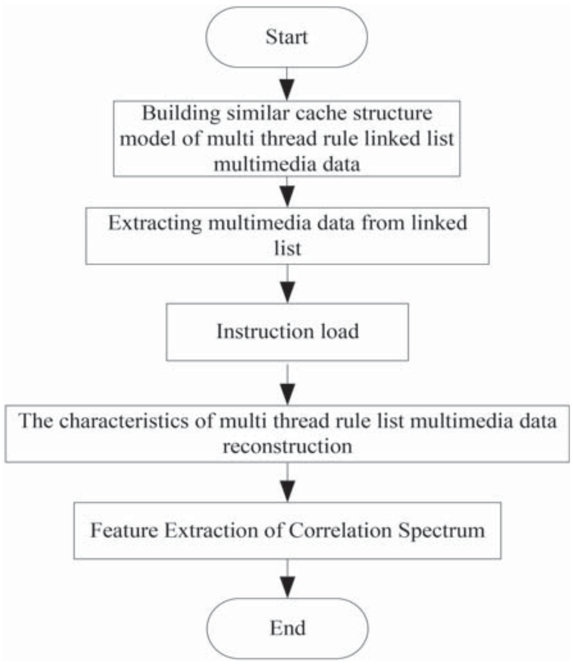

Figure 2 Compression algorithm flow.

Step 2: Read the character as the current character S.

Step 3: Determine if the string PS is in the dictionary:

1) If "Yes", use S to extend P, that is, P = P + S.
2) If "No", add P + S to the dictionary with a codeword corresponding to the current prefix P, update P to P = S, and now P contains only one character S.

Step 4: Determine whether there are any codewords to be translated in the character stream:

1) If "Yes", go back to step 2.
2) If "No", go back to step 5.

Step 5: Output the codeword representing the current prefix P to the codeword stream and the encoding ends.

The source data character stream aabbbabb is coded below according to the above operation steps. According to the source data, a string table is initialized, each character in the string table corresponds to an index, and the string table is coded by LZW. The dictionary is initialized as Table 2.

The LZW_CLEAR in Table 2 indicates initialization of the string table, and LZW_EOI means the end of the encoding.

2.3 Data Storage Structure Analysis and Location Node Deployment

Based on the above analysis, combined with the advantages of dictionary compression algorithm and statistical compression algorithm, a multimedia data compression algorithm based on artificial intelligence is proposed. The general flow of the algorithm is as follows:

In order to realize the lossless compression optimization design of multimedia data based on the artificial intelligence multi-thread rule linked list, it is necessary to first construct the data storage structure analysis and the node location model, realize the lossless compression optimization of multimedia data of the multi-thread rule linked list, and then set $A \subset V$, $B \subset V$ and $A \cap B = \varphi$, design the node distribution link structure of multimedia data of the multi-thread rule linked list under the distributed cloud computing environment, assume the multimedia data set of the multi-thread rule linked list, calculate the storage cost of the central node, use ontology models A and B to perform statistical feature analysis and adaptive matching of the multimedia data of the multi-thread rule linked list (Zhang et al., 2019), set the corresponding popularity occupied space as $G(A)$ and $G(B)$ constructing a similar cache community storage area, which is expressed as a pair of route hops of a and b, deploy nodes in the directed graph $G = (V, E)$ table for lossless compression of the multimedia data of the multi-thread rule linked list, and obtain an optimized node deployment model as shown in Figure 3.

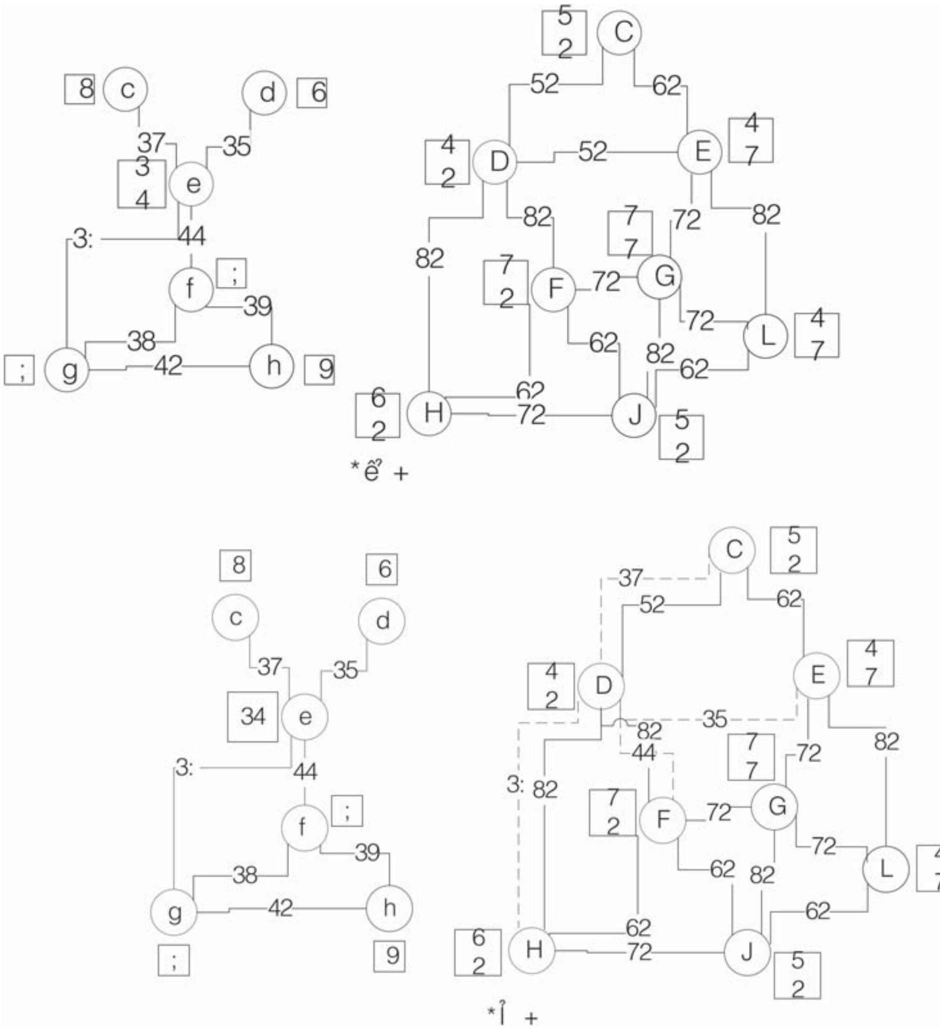

Figure 3 Node deployment model for lossless compression of multimedia data in multithreaded rule linked lists.

In the multimedia data lossless compression node model of the multi-thread rule linked list shown in Figure 3, a directed graph model is established according to the route hop number reduction distribution to obtain the edge vector of the storage structure model. The multimedia data lossless compression array nodes of the distributed multi-thread rule linked list all have the same storage medium characteristics, which are denoted as r, the popular content threshold of the nodes is analyzed (Zhang et al., 2012; Mao et al., 2016; Lin et al., 2016), and the HAN similarity between adjacent nodes is calculated as:

$$\begin{aligned} x_{id}(t+1) &= wx_{id}(t) + c_1 r_1 [r_3^{t_0 > T_0} p_{id} - x_{id}(t)] \\ &\quad + c_2 r_2 [r_4^{t_g > T_g} p_{gd} - x_{id}(t)] \end{aligned} \tag{1}$$

In the formula, t_0 and t_g respectively represent the link distribution set and storage cost function deployed by the site selection node; T_0 and T_g are respectively the response information and threshold of adjacent nodes. According to the above analysis, a similar cache structure model of the multi-thread rule linked list multimedia data is constructed, and the distributed cloud computing method is adopted to mine the autocorrelation characteristics of the multi-thread rule linked list multimedia data.

2.4 Autocorrelation Feature Mining of Multimedia Data Based on Multi-Thread Rule Linked List

A distributed cloud computing method is adopted to mine autocorrelation characteristics of the multimedia data of a multi-thread rule linked list, cache state characteristic values of part of peripheral nodes are calculated, and transmission capacities of storage equipment and switches are obtained as follows:

$$\begin{aligned} &\min \quad F(x) = (f_1(x), f_2(x), \ldots, f_m(x))^T \\ &s.t. \quad g_i \le 0, \qquad i = 1, 2, \ldots, q \\ &\qquad\quad h_j = 0, \qquad j = 1, 2, \ldots, p \end{aligned} \tag{2}$$

In the formula, in the time period T of a high frequency query, the internal cache state type is r_k, and the statistical average value of multimedia data lossless compression of multi-threaded rule linked list is a random variable (denoted as X), which represents the random probability distribution density of multimedia data lossless compression of multi-threaded rule linked list, and A represents the time delay function. Mining the quantitative statistical average value of the multimedia data of a multi-thread rule linked list (Zhou

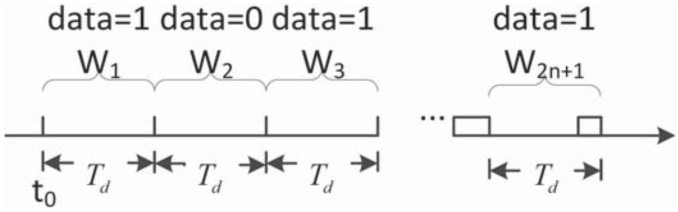

Figure 4 Priority attribute list for the lossless compression of multimedia data in a multi-thread rule linked list.

et al., 2014; Qi et al., 2014; Mohan and Govardhan, 2013), adopting the method of sub-graph sequence reconstruction, the frame function of lossless compression of multimedia data of a multi-thread rule linked list is formed by m time slots, the attribute set of the category of intelligent scheduling of multimedia data of the *i* network multi-thread rule linked list is $delay(v)$, the cache states of some peripheral nodes are classified and analyzed, assuming that each storage node $v_i = \{v_{i1}, v_{i2}, \cdots, v_{ip}\}$ has a delay attribute *delay(v)*, the fuzzy constraint cost function of lossless compression of multimedia data of the multi-thread rule linked list is obtained as follows:

$$\mathrm{AT}(v) = \max_{u \in FI(v)} [AT(u) + delay(v)] \tag{3}$$

$$\mathrm{RT}(v) = \min_{w \in FO(v)} [RT(w) - delay(v)] \tag{4}$$

Finally, according to the signaling between switches, it is returned to the IC Compiler for lossless compression of the multi-threaded rule linked list multimedia data. According to the timing information of each node, the interactive delay of lossless compression of multi-threaded rule linked list multimedia data is obtained as follows:

$$bnr_\beta(X) = R_\beta X - R_\beta X_1 \tag{5}$$

In order to cooperatively utilize the storage resources of each node, according to the feature size of lossless compression of multimedia data in multi-thread rule linked list, the stack statistical feature value distribution is obtained to meet the following requirements:

$$\sum_{m=1}^{n} x_G^m \le E_G, \sum_{m=1}^{n} x_T^m \le E_T, \sum_{m=1}^{n} x_W^m \le E_W, \sum_{m=1}^{n} x_L^m \le E_L \tag{6}$$

Updating the scheduling efficiency of data transmission link nodes to obtain the optimized sum of multimedia data lossless compression efficiency of multi-thread rule linked list as $\sum_{\sigma} \mu^{m\omega} T_\sigma^\omega$, where $\omega \subseteq \{G, T, W, L\}$, $m \in [1, n]$. In the limited subspace, the number of lossless compression of multimedia data in the multi-threaded rule linked list exceeds T. From t_0 onwards, lossless compression scheduling is carried out in T_d units to realize the autocorrelation feature mining of the multimedia data in the multi-threaded rule linked list (Chen et al., 2018; Yeohee and Verdú, 2017).

3. LOSSLESS COMPRESSION OPTIMIZATION

3.1 Feature Reorganization of Multimedia Data in Multithread Rule Linked List

Based on the above-mentioned similar cache structure model of multi-thread rule linked list multimedia data, and using the distributed cloud computing method to mine the autocorrelation characteristics of multi-thread rule linked list multimedia data, the optimization design of multi-thread rule linked list multimedia data lossless compression is carried out. This paper proposes a multi-thread rule linked list multimedia data lossless compression method based on artificial intelligence. By extracting high-dimensional statistical information of multimedia data of a multi-threaded rule linked list to obtain a storage space rule set $S = \{s_i | i = 1, 2, \cdots, N_S\} (S \subset C)$, scheduling cache data by utilizing the capacity of a switch, the energy in the domain of the controller end can be obtained as follows:

$$E = [E_G, E_T, E_W, E_L] \tag{7}$$

Combining this with the phase space reconstruction method, the feature reorganization of multimedia data of the multi-thread regular linked list is carried out to obtain the stored state space vector feature $C \subset S$, and the F phase space reconstruction model is as follows:

$$S_p(u) = \{F^p[s(t)]\}(u) = \int_{-\infty}^{\infty} K_p(t, u) s(t) dt \tag{8}$$

$s(t) \to s(t)$, $s(t) \to S(f)$, $s(t) \to s(-t)$, $s(t) \to S(-f)$ are regarded as the upper limit of storage capacity of network nodes respectively, and an association rule mapping method is adopted to construct a priority attribute list for the lossless compression of multimedia data in a multi-threaded rule linked list, as shown in Figure 4.

According to the priority attribute list of multimedia data lossless compression in the multi-threading rule linked list of Figure 4, lossless compression and feature reorganization are carried out to improve the balance of data storage. In the multimedia data link layer (Chen et al., 2017), the node distribution model of the multi-thread rule linked list multimedia data location is recorded as $x_1, x_2, \cdots, x_{m+1}$, and the state equation of the multimedia data link layer cooperative storage can be obtained:

$$x_1 + x_2 + \cdots + x_{m+1} = T + t - m \times t \tag{9}$$

The optimal allocation of storage space by piecewise interpolation method is as follows:

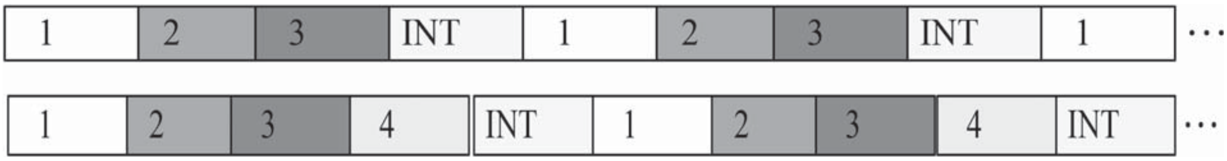

Figure 5 Multi-thread rule linked list multimedia data lossless compression storage structure model.

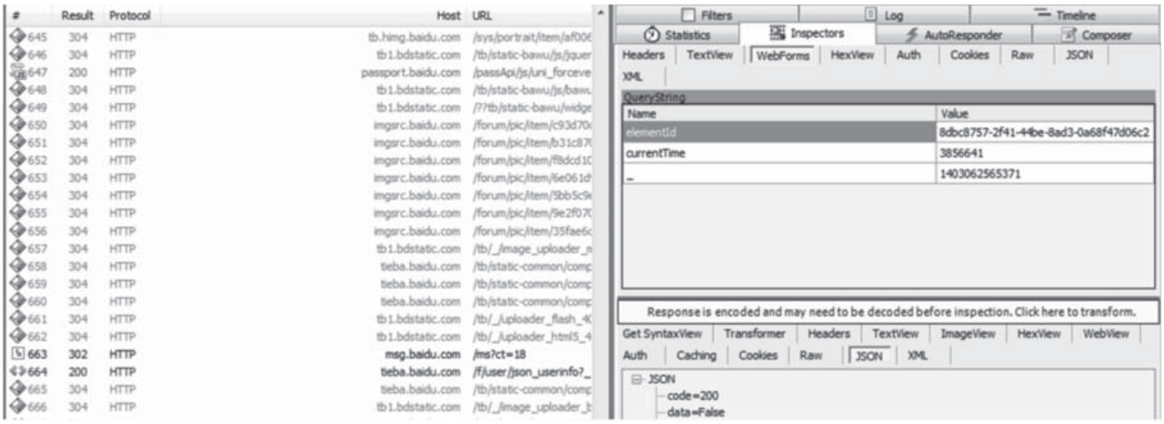

Figure 6 Simulation operation interface.

$$H_i(x) = \sum_{k=1}^{K} p_k \ln \frac{1}{p_k} = -\sum_{k=1}^{K} p_k \ln p_k \tag{10}$$

Wherein, K represents the data transmitting and receiving node for multimedia cooperative storage of the multi-thread rule linked list, and p_k is the spatial capacity distribution of the multimedia data link layer, thus realizing the feature reorganization of the multimedia data.

3.2 Lossless Compression Optimization of Multimedia Data Based on a Multi-Threaded Regular Linked List

The knowledge map technology is used to realize the integration and fusion of the main components of the multi-thread rule linked list multimedia data, and the data storage scheduling model for collaborative storage is expressed as follows:

$$\begin{cases} x = (x_1, x_2, \ldots, x_n) \\ y = F(x) = (f_1(x), f_1(x), \ldots, f_m(x))^T \end{cases} \tag{11}$$

Wherein, $x = (x_1, x_2, \ldots, x_n)$ is a multimedia data set of a multi-thread rule linked list; $y = F(x)$ represents the allocation node set of storage space (Ravanmehr et al., 2017), and reads the frame set $P(n_i) = \{p_k | pr_{kj} = 1, k = 1, 2, \cdots, m\}$ of data to obtain the distribution set of corresponding frame types as follows:

$$RTT_s = (1 - \alpha) \times RTT_s + \alpha \times RTT \tag{12}$$

For single node storage capacity, $Color(u) = Uncolor$, $Sch(u)$, where:

$$Sch(u) = NY \tag{13}$$

The cache of each peripheral node is replaced by:

$$flow_k = \{n_1, n_2, \cdots, n_q\}, q \in N \tag{14}$$

In the formula, q represents the load of lossless compression of multimedia data in a multi-threaded regular linked list, n_q represents the feature distribution sequence of cache replacement information, and when $Sch(u) = RY$, the quantized feature set output in the cache state module is:

$$x_n = [x(0), x(1), \cdots, x(N-1)]^T \tag{15}$$

To sum up, combined with cloud computing technology, the main component integration and lossless compression optimization of the multi-thread rule linked list multimedia data are realized (Zhu et al., 2015), and the optimized storage structure model is shown in Figure 5.

4. SIMULATION EXPERIMENT AND RESULT ANALYSIS

In order to test the application performance of the method in realizing lossless compression of multi-threaded rule linked lists of multimedia data, experimental analysis is carried out, loading and intelligent scheduling of multi-threaded rule linked lists of multimedia data are completed in the class file MinePressureCollectionC.nc, and cloud computing integrated processing of data is carried out using a cross compiler. In the NS-2.27 cloud computing platform, multi-thread regular linked list multimedia data is reorganized into a distributed structure. The operating interface is as follows:

The dynamic transmission rate of multimedia data is set to 32 Mbyte/s, the storage scale set of multimedia data is 2000, the attribute category of multimedia data is 12, and the multimedia data simulates 300 real-time storage tasks. The distribution of data to be stored is shown in Figure 7.

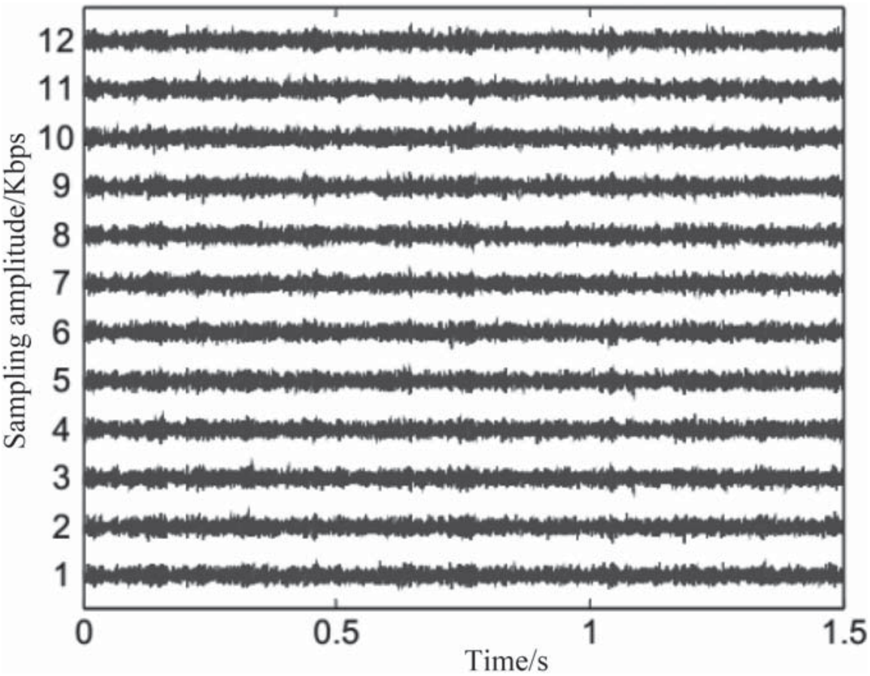

Figure 7 Multimedia data of multi-thread rule linked list.

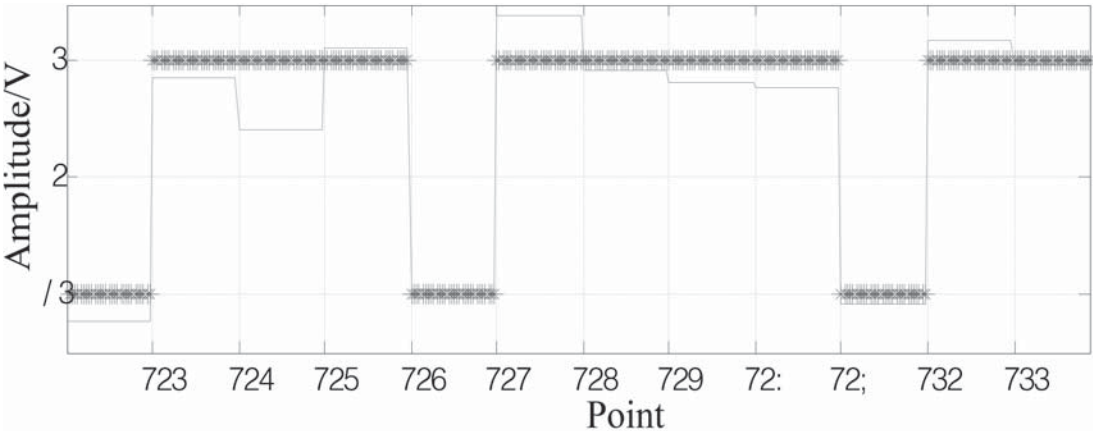

Figure 8 Optimized storage distribution.

Taking the multimedia data of the multi-thread rule linked list of Figure 7 as the research object, lossless compression design is carried out, and the optimized storage distribution is shown in Figure 8.

According to the analysis of Figure 8, the multimedia data lossless compression design using the multi-thread regular linked list method has better dynamic allocation capability and better balance. The throughput capacity of data storage was tested and the comparison result is shown in Figure 9.

The research shows that the multimedia data lossless compression using the multi-thread rule linked list method in this paper has higher load, higher storage capacity and better real-time access to data.

Five groups of data were randomly selected to do histogram statistics on the multimedia data after the solution-related process processing, as shown in the original data histogram in Figure 10 as a1, a2, a3, a4, a5, and the data distribution is messy. The statistical histograms of a histogram after correlation are corresponding b and c histograms, histograms b1, b2, b3, b4, b5 are statistical histograms of adjacent data corresponding to a graph (1, 2, 3, 4, 5), histograms c1, c2, c3, c4, c5 are statistical histograms of the corresponding data processed between the data of the same eigenvalue using the least squares method on the basis of the histograms of b (1, 2, 3, 4, 5).

Where, (a1–a5) raw data; (b1–b5) neighboring data; (c1–c5) data of the same eigenvalue. The vertical axis (Probability) in Figure 10 represents the probability of the data value appearing after processing the data, and the horizontal axis (Pixel) represents the data value after processing the multimedia data. To reduce the correlation of adjacent data, the results shown in the histograms of b1, b2, b3, b4 and b5 are obtained by using the method of doing poor processing of adjacent data. The processed data are mainly concentrated between -50 and 50, and the fluctuation range decreases, showing a certain regularity. The above results are converted into data of specific size, and the statistical results are shown in Table 3.

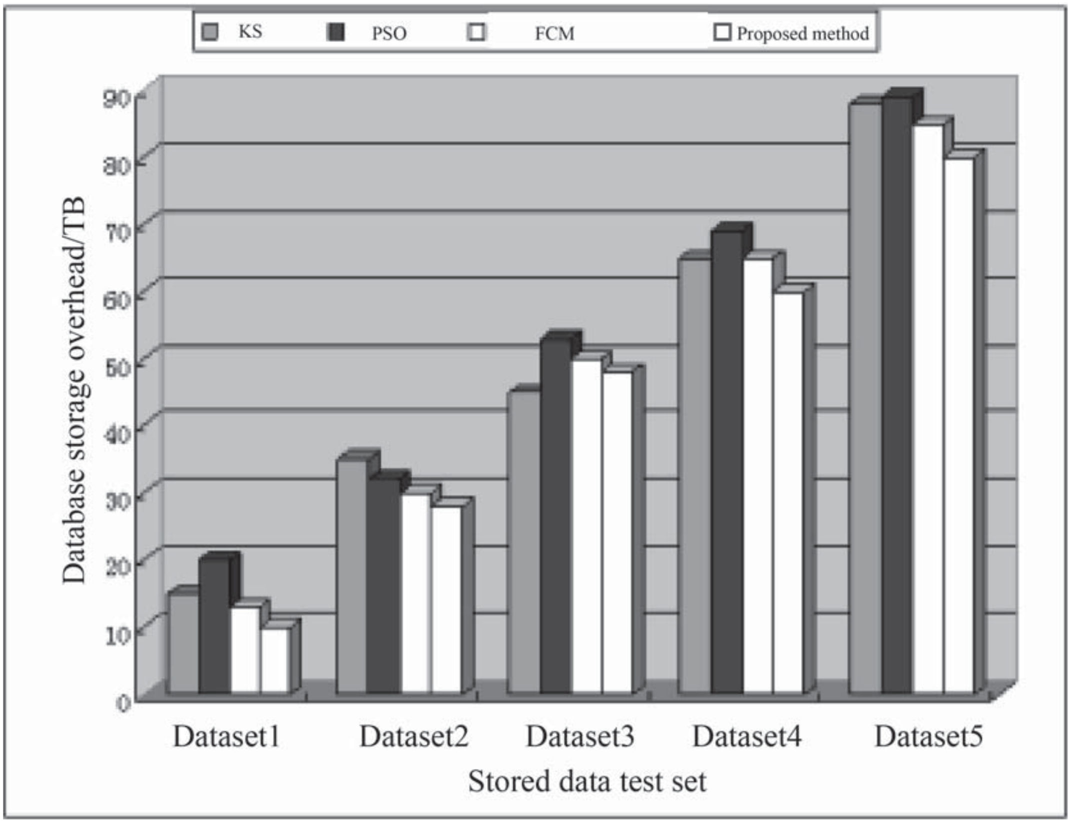

Figure 9 Capacity comparison of data storage.

Table 3 Difference processing results of adjacent data.

Multimedia Data Name	Raw Data Size /Kb	Size of Adjacent Processing Data /Kb	D-Value /Kb
A	3452	3425	27
B	2435	2325	110
C	654	606	48
D	2547	2432	115
E	1320	1330	10
F	1299	1187	112
G	1658	1786	−128
H	749	720	29
I	563	463	100

After the first step operation to solve the correlation, from the difference between the original data of Table 3 and the adjacent data processing, the overall correlation of multimedia data can still be reduced. Although the difference between multimedia data can reduce the correlation of some data, it can also increase the space occupied by some data. However, this kind of processing of multimedia data provides convenience for the compression processing of subsequent multimedia data.

5. CONCLUSION

Intelligent selection and distributed network planning and design were carried out, and an intelligent storage structure model of multi-thread rule linked list for multimedia data was established to realize the intelligent storage and transmission control of data. This paper proposed a lossless compression method of multi-thread rule linked list for multimedia data based on artificial intelligence. A similar cache structure model of multi-thread rule linked list multimedia data is constructed, a distributed cloud computing method is adopted to mine the autocorrelation characteristics of the multi-thread rule linked list multimedia data, high-dimensional statistical information of the multi-thread rule linked list multimedia data is extracted, a phase space reconstruction method is combined to reconstruct the characteristics of the multi-thread rule linked list multimedia data, and knowledge map technology is used to realize the integration and fusion of the main components of the multi-thread rule linked list multimedia data, thus realizing lossless compression optimization. The multi-thread regular linked list lossless compression design of multimedia data using the method in this paper has better dynamic allocation capacity, better balance and improved storage capacity.

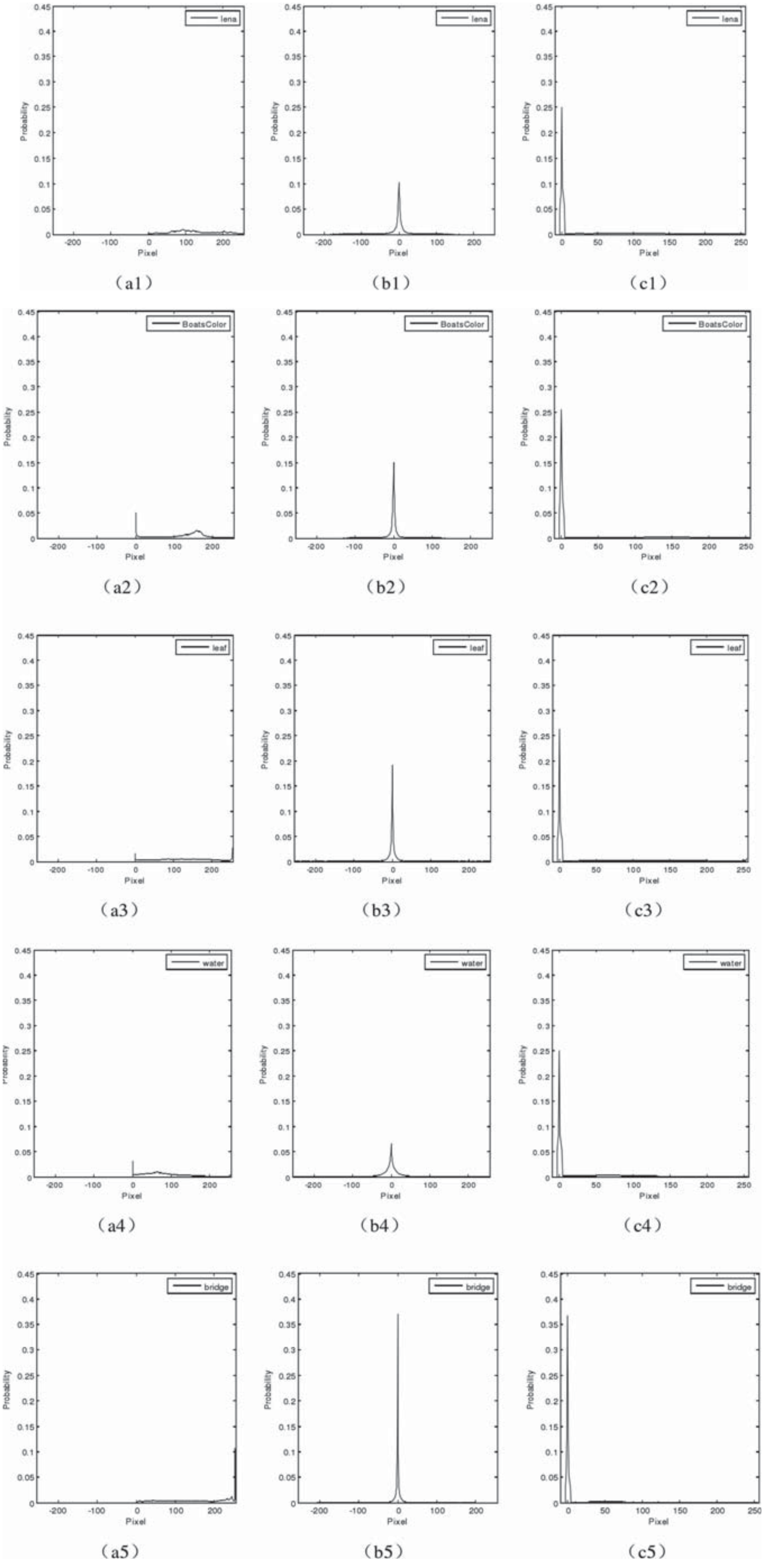

Figure 10 Histogram.

REFERENCES

1. Anand M., M Sundararajan (2020). Fractal Antenna – Wireless Communication New Beginning Breakthrough in Digital Era. *Acta Informatica Malaysia*, 4(1): 01–06.
2. Chen, J., Wu, W.H., Ding, Z. LTE multimedia broadcast multicast service provisioning based on robust header compression. *IEEE Transactions on Wireless Communications*, 2017, 22(9): 53–60.
3. Chen, L.Q., Yan, L.X., Sang, H.S., et al. High-throughput architecture for both lossless and near-lossless compression modes of LOCO-I algorithm. *IEEE Transactions on Circuits and Systems for Video Technology*, 2018, 17(32): 23–30.
4. Chen, Y., Li, L.J. Very fast decision tree classification algorithm based on red-black tree for data stream with continuous attributes. *Journal of Nanjing University of Posts and Telecommunications (Natural Science Edition)*, 2017, 37(2): 86–90.
5. Hashemi, S., Yang, Y. Flexible decision tree for data stream classification in the presence of concept change, noise and missing values. *Data Mining and Knowledge Discovery*, 2009, 19(1): 95–131.
6. Huang, S.C., Liu, Y. Classification algorithm for noisy and dynamic data stream. *Journal of Jiangsu University of Science and Technology (Natural Science Edition)*, 2016, 30(3): 281–285.
7. Ju, C.H., Zou, J.B. An incremental classification algorithm for data stream based on information entropy diversity measure. *Telecommunications Science*, 2015, 31(2): 86–96.
8. Lin, J.M., Ban, W.J., Wang, J.Y., et al. Query optimization for distributed database based on parallel genetic algorithm and max-min ant system. *Journal of Computer Applications*, 2016, 36(3): 675–680.
9. Lyu, Y.X., Wang, C.Y., Wang, C., et al. Online classification algorithm for uncertain data stream in big data. *Journal of Northeastern University (Natural Science Edition)*, 2016, 37(9): 1245–1249
10. Mao, W.T., Tian, Y.Y., Wang, J.W., et al. Granular extreme learning machine for sequential imbalanced data. *Control and Decision*, 2016, 31(12): 2147–2154.
11. Mohan, B., Govardhan, A. Online aggregation using MapReduce in MongoDB. *International Journal of Advanced Research in Computer Science and Software Engineering*, 2013, 3(9): 1157–1165.
12. Qi, C.Y., Li, Z.H., Zhang, X., et al. The research of cloud storage system performance evaluation. *Journal of Computer Research and Development*, 2014, 51(S1): 223–228.
13. Qiao, Z.W., Sun, W.X. Two improvements to C4.5 algorithm. *Journal of Jiangsu Polytechnic University*, 2008, 20(4): 56–59.
14. Ravanmehr, V., Kim, M., Wang, Z.Y., et al. ChIPWig: A random access-enabling lossless and lossy compression method for ChIP-seq data. *Bioinformatics, 2017*, 34(6): 911–919.
15. Sun, B., Wang, J.D., Chen, H.Y., et al. Diversity measures in ensemble learning. *Control and Decision*, 2014, 29(3): 385–395.
16. Syed Mamun R.R., Md. Belayet H., Md. E.H., Md. Ariful A.A., and Md. Sarikat A.S. (2019). Modeling and Control of a Magnetic Levitation System Using an Analog Controller. *Acta Electronica Malaysia*, 3(2): 41–44.
17. Tan M., Yuan J.S., Wei M., Zhang Y.Q. (2019). Commercial Complex Intelligence and Program Research. *Information Management and Computer Science*, 2(1): 04–06
18. Yeohee, I.M., Verdú, S. Optimal universal lossless compression with side information. *IEEE Transactions on Information Theory*, 2017, 11(8): 99–103.
19. Zhang, H.L., Li, X.F., Yang, S.B., Xu, J.Y., Li, H.Y., Wang, Q. Dual closed-loop fuzzy PID depth control for deep-sea self-holding intelligent buoy. *Information and control*, 2019, 48(2): 202–208, 216.
20. Zhang, Y., Li, Z.R., Liu, X.D. Active learning SMOTE based imbalanced data classification. *Computer Applications and Software*, 2012, 29(3): 91–93.
21. Zhou, X.P., Zhang, X.F., Zhao, X.N. Cloud storage performance evaluation research. *Computer Science*, 2014, 41(4): 190–194.
22. Zhou, Y.H., Zhang, H.L., Li, F.F., Qi, P. Local focus support vector machine algorithm. *Journal of Computer Applications*, 2018, 38(4): 945–948.
23. Zhu, X.B., Liu, Y.Q., Cai, Q., et al. A survey on content-based image retrieval technology. *Computer Simulation*, 2015, 32(5): 1–4, 85.