International Journal of Database Theory and Application
Vol.9, No.8 (2016), pp.1-12
http://dx.doi.org/10.14257/ijdta.2016.9.8.01

MapReduce Based Remote Sensing Image Retrieval Algorithm

Shen Xibing[1], Wei Rong[2] and Yang Yi[3]

[1]*College of resources and environment, Qinzhou University, Qinzhou, 535000, China*
[2]*College of foreign language, Qinzhou University, Qinzhou, 535000, China*
[3]*College of software, Guangxi University of Science and Technology, Liuzhou, 545006, China*

Abstract

The remote sensing images are massively stored, so it is difficult for the traditional single-node mode to meet the real-time requirement for remote sensing image retrieval. In order to improve remote sensing image retrieval efficiency and accuracy, a kind of feature information MapReduce based remote sensing image retrieval algorithm is proposed in this article. Specifically, the color features and the texture features of the remote sensing image are firstly extracted, and then Map function is adopted to calculate the similarity between the remote sensing image to be retrieved and the image in the feature library according to the color features and the texture features, and finally Reduce function is adopted to collect the intermediate results of various node tasks and the remote sensing images are ranked by a descending order according to the similarity in order to obtain the remote sensing image retrieval result. The test result shows that the proposed algorithm can rapidly and accurately retrieve the remote sensing image, thus not only improving the remote sensing image retrieval efficiency, but also improving the remote sensing image retrieval accuracy.

Keywords: *Remote Sensing Image; Feature Extraction; Cloud Computing; Retrieval Algorithm*

1. Introduction

Along with the development of the satellite remote sensing technology, the remote sensing image data are gradually increased, but the traditional manual retrieval methods have such defects as large workload and low efficiency and cannot meet the remote sensing image application requirement. The computer based automatic retrieval of the remote sensing image can improve the remote sensing image retrieval efficiency and effect, so the design of the remote sensing image retrieval algorithm with high efficiency and high retrieval accuracy becomes an important task of the present research [1].

In allusion to problems regarding the automatic retrieval of the remote sensing image, the scholars at home and abroad have carried out a lot of researches. Specifically, the content based image retrieval (CBIR) has such advantages as rapid retrieval speed and high accuracy, thus becoming the main retrieval algorithm for the remote sensing image retrieval, wherein such remote sensing image features as color, shape and texture are firstly extracted to describe the remote sensing image content, and then these features are compared with the remote sensing image feature library in order to obtain the retrieval result [2-4]; the traditional single-node mode cannot meet the real-time requirement for the remote sensing image retrieval [5,6]; the distributed processing technology can be used to allocate the tasks to various working nodes for parallel processing in order to jointly finish the tasks through the mutual coordination of the nodes, so the distributed processing technology becomes a new solution for the remote sensing image retrieval[7].

ISSN: 2005-4270 IJDTA

At present, the distributed processing technology mainly includes grid computing and cloud computing, wherein Hadoop is the basic distributed system architecture, and the users can develop MapReduce distributed program without the need to know the bottom layer information. As a result, Hadoop which can be used for large-scale data analysis has become a mainstream parallel processing model in the cloud computing field, and has been widely applied in such fields as virtual database, large-scale data processing, biomedicine and patent image classification [8].

In order to improve remote sensing image retrieval efficiency and accuracy, MapReduce based remote sensing image retrieval algorithm is proposed in this article. Specifically, the color features and the texture features of the remote sensing image are firstly extracted, and then Map function is adopted to match the remote sensing image according to the color features and the texture features and calculate the similarity between the remote sensing image to be retrieved and the image in the feature library, and finally Reduce function is adopted to collect the intermediate results of various computing node tasks and the remote sensing images are ranked by a descending order according to the similarity in order to obtain the remote sensing image retrieval result. The test result shows that the proposed algorithm can rapidly and accurately retrieve the remote sensing image, thus not only improving the remote sensing image retrieval efficiency, but also improving the remote sensing image retrieval accuracy.

2. Remote Sensing Image Feature and Similarity Matching

In CBIR based remote sensing image retrieval, the features of the remote sensing image to be retrieved are firstly extracted, and then the similarity between the remote sensing image to be retrieved and the image in the remote sensing image library is calculated, and finally the image retrieval is realized according to the similarity.

2.1. Remote Sensing Image Extraction

Color is an important feature for remote sensing image classification, so it is necessary to extract the color features of the remote sensing image in RGB color space to obtain such color features as RGB mean value, R mean value, G mean value and B mean value.

Texture feature can be used to describe the space variation of the remote sensing image. In this article, Gabor filter is adopted to extract the texture features of the remote sensing image, wherein Gabor filter h(x,y) and Fourier transformation H(u,v) are as follows:

$$\begin{cases} h(x,y) = g(x+y)\exp\left(2\pi\sigma f x\right) \\ H(u,v) = \exp\left[\dfrac{(u-f)^2 + \hat{v}^2}{2a^2}\right] \end{cases} \tag{1}$$

$$\text{Therein, } \begin{cases} g(x,y) = \dfrac{1}{2\pi\sigma^2}\exp\left(-\dfrac{x^2+y^2}{2\sigma^2}\right) \\ (x,y) = (x\cos\theta + y\sin\theta, -x\sin\theta + y\cos\theta) \\ (u,\hat{v}) = (u\cos\theta + v\sin\theta, -u\sin\theta + v\cos\theta) \\ f\times\sigma = \lambda\dfrac{(2^B+1)}{(2^B-1)} \end{cases} \tag{2}$$

$$\begin{cases} \lambda = \dfrac{\sqrt{2\ln 2}}{\pi} \\ a = \dfrac{1}{2\pi\sigma} \end{cases} \tag{3}$$

In the above formula, f is the center frequency of the filter band-pass area, B is the width of the filter, θ is the direction angle of the principal axis of the filter, and σ is variance.

Gabor filter parameter is determined according to Formulae (1) ~ (3), and then the energy values of the convolution between various filters and the image are calculated, and finally the mean value and the mean square error of the image filtering energy value are taken as the texture features of the remote sensing image, namely:

$$F_{texture} = \{\mu_{0,0}^{texture}, \sigma_{0,0}^{texture}, \cdots, \mu_{k-1,l-1}^{texture}, \sigma_{k-1,l-1}^{texture}\} \tag{4}$$

In the above formula, K is the number of the center frequencies and L is number of the direction angles.

The energy mean value μ and the mean square error σ of the sub-image can be calculated as follows:

$$\begin{cases} \mu_{k,l}^{texture} = \dfrac{\sum_x \sum_y E_{k,l}(x,y)}{n \times n} \\ \sigma_{k,l}^{texture} = \sqrt{\dfrac{\sum_x \sum_y E_{k,l}(x,y) - \mu_{k,l}^{texture}}{n \times n}} \end{cases} \tag{5}$$

In this way, 24 texture features of the remote sensing image can be obtained. Therefore, the remote sensing image totally includes 28 features composed of color features and texture features.

2.2. Similarity Matching

The remote sensing image to be retrieved is $p0$, the remote sensing image library totally includes pi images (i=1, 2,…, n), the color features thereof are expressed as $ci \in R$m, the texture features thereof are expressed as $ti \in R$k, the dimensionalities of the color features and the texture features are respectively M and K, and the similarity between $p0$ and pi(i=1,2,…,n) are calculated according to Formula (6).

$$R_{0i} = w_1 D_{t_i} + w_2 D_{c_i} \tag{6}$$

In the above formula, w_1 and w_2 are the weight values ($w_1 + w_2 = 1$), D_{t_i} and D_{c_i} respectively represent the color feature similarity and the texture feature similarity, and the calculation formulae are as follows:

$$\begin{cases} D_{t_i} = 1 - \dfrac{\left(\sum_{m=1}^{M} (t_0^m - t_i^m)^2\right)^{1/2}}{\max_i \left(\sum_{m=1}^{M} (t_0^m - t_i^m)^2\right)^{1/2}} \\ D_{c_i} = 1 - \dfrac{\left(\sum_{k=1}^{K} (t_0^k - t_i^k)^2\right)^{1/2}}{\max_i \left(\sum_{k=1}^{K} (c_0^k - c_i^k)^2\right)^{1/2}} \end{cases} \tag{7}$$

For Roi (i=1,2,…,n), the images in the remote sensing image library are ranked by a descending order, and the first m images are selected as the retrieval results.

3. Mapreduce Based Remote Sensing Image Retrieval

3.1. Mapreduce Based Image Storage

As the basis for the automatic retrieval of the remote sensing images, image storage is a data-intensive calculation process. MapReduce distributed processing method is adopted in this article to upload the images to HDFS. The specific process is as follows:

(1) Map stage: Map function is adopted to read one remote sensing image at each time and meanwhile extract the color features and the texture features of the image.

(2) Reduce stage: the extracted feature data of the remote sensing image are stored in HDFS. HBase is a column oriented distributed database, so HBase table is adopted for the remote sensing image storage of HDFS, wherein the specific design of HBase table is as shown in Table 1.

Table 1. Hbase Table Design of Remote Sensing Image

Remote Sensing Image id	Original Image File	Color Feature	Texture Feature
001	file001	*c*1	t1
002	file001	*c*2	t2
…	…	…	…
00n	file00n	*c*n	tn

MapReduce based image storage process is as shown in Figure 2.

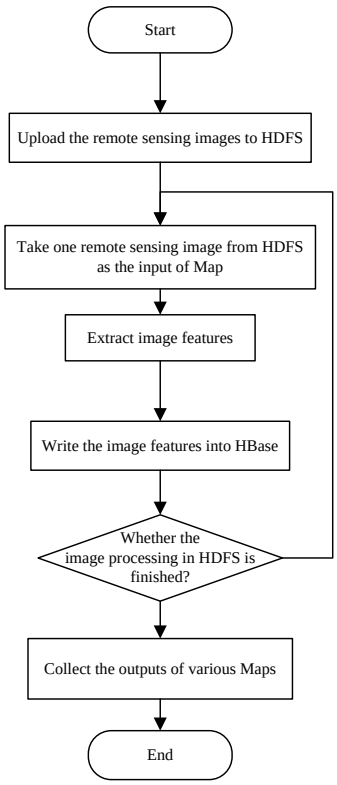

Figure 2. Remote Sensing Image Storage Process

3.2. Mapreduce Based Remote Sensing Image Retrieval

Since the remote sensing images and the features thereof are stored in HBase, when the data set of HBase is very large, it will take long time to scan the whole table. In order to reduce image retrieval time and improve the retrieval efficiency, MapReduce calculation model is adopted for the parallel computation of the remote sensing image retrieval. The specific frame diagram is as shown in Figure 3, and the specific execution process is as shown in Figure 4.

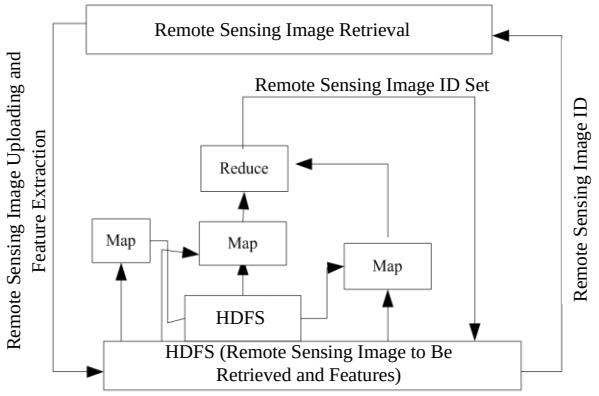

Figure 3. Remote Sensing Image Retrieval Frame Diagram

MapReduce based remote sensing image retrieval steps are as follows:

Step 1: Map stage: read the remote sensing image to be retrieved from HDFS cache, then extract the color features and the texture features, then match the similarity between the extracted features and the image features in HBase, and then output the key value <Similarity, Image ID> from Map.

Step 2: according to the similarity values, rank and redistribute all key values <Similarity, Image ID> output by Map, and then input them into Reducer.

Step 3: Reduce stage: collect all key value pairs < Similarity, Image ID>, then rank the similarities of these key values, and then write the first N key values into HDFS.

Step 5: Output the IDs of the images that are most similar to the remote sensing image to be retrieved.

```
map(key,value)
Begin
Csearch=ReadSearchCharact( ); //
Cdatabase=value; //
Path = GetPicturePath( value) ; / /
SimByColor=CompareByColor(Csearch, Cdatabase) ; / /
SimByTexture = CompareByTexture(Csearch, Cdatabase); //
Sim=SimByColor*w1 + SimByTexture*w2; //
Commit(Sim,Path);
End
```

Map function is defined as follows:

```
map(key,value)
Begin
Csearch=ReadSearchCharact( ); //read the features of the remote sensing image to be retrieved
Cdatabase=value; //read the data in the remote sensing feature library
Path = GetPicturePath( value) ; / /read the image path in the remote sensing image library
```

SimByColor=CompareByColor(Csearch, Cdatabase) ; / /calculate the remote sensing image color similarity

SimByTexture = CompareByTexture(Csearch, Cdatabase); //calculate the remote sensing image texture similarity

Sim=SimByColor*w1 + SimByTexture*w2; //calculate and match similarity

Commit(Sim,Path);

End

reduce(key,value):

Begin

Sort(key,value); //

Commit(key,value); //

End

Reduce function is defined as follows:

reduce(key,value):

Begin

Sort(key,value); //rank the remote sensing images according to the similarity values

Commit(key,value); //key value refers to similarity value, and value refers to the path of the similar remote sensing image

End

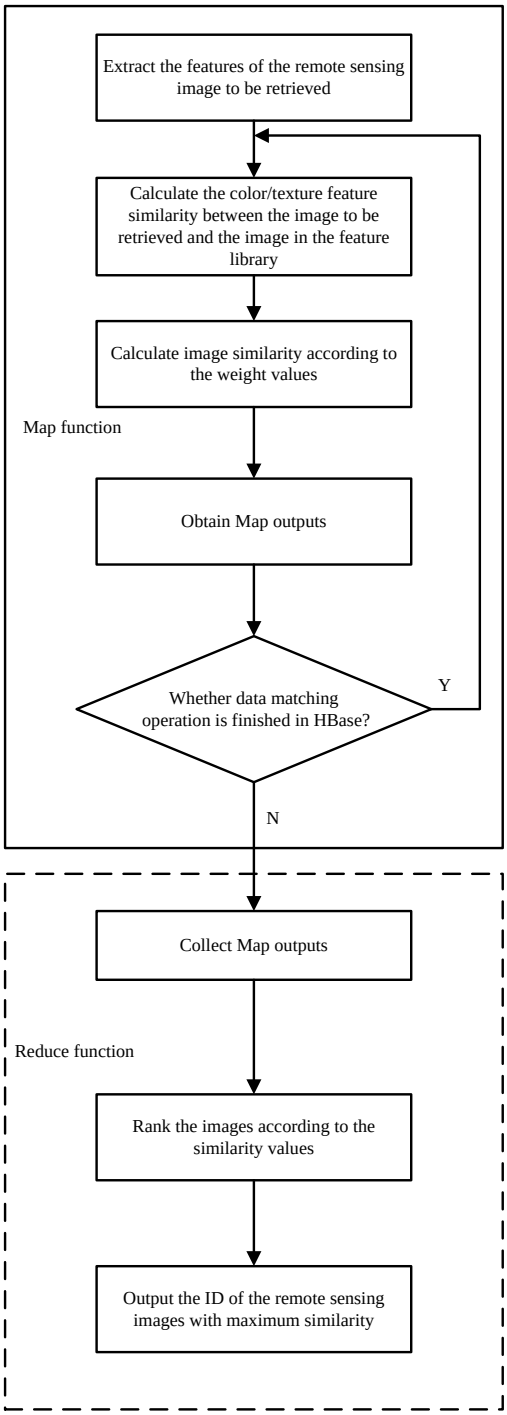

Figure 4. Mapreduce Based Remote Sensing Image Retrieval Process

4. System Test and Analysis

4.1. Experiment Environment

The Hadoop distributed system composed of one host computer and three common computers through Linux environment is adopted in this article, and the specific configurations are as shown in Table 2. 20,000 remote sensing images are collected. In

order to more persuasively explain the result of the proposed remote sensing image retrieval algorithm, B/S single-node system is adopted for the comparison experiment.

Table 2. Node Configuration

Node	Operating System	IP	CPU	RAM
Host Computer	Linux	192.168.0.101	Conroe i7 3960X 3.3GHz	4G
Common Computer 1	Linux	192.168.0.102	Conroe i3 2120 3.3GHz	2G
Common Computer 2	Linux	192.168.0.103	Conroe i3 2120 3.3GHz	2G
Common Computer 3	Linux	192.168.0.104	Conroe i3 2120 3.3GHz	2G

4.2. Storage Performance Test and Analysis

The storage time of different quantities of remote sensing images under different node conditions is as shown in Figure 5. According to Figure 5, when the quantity of the remote sensing images is less than 500, the storage time in B/S single-node system is slightly different from that in Hadoop distributed system, and the advantage is not obvious. When the quantity of the remote sensing images is more than 500, the storage time in B/S single-node system is significantly increased, but the storage time in Hadoop distributed system is slowly increased, thus indicating that the adoption of MapReduce mode for uploading the remote sensing images to HDFS can improve the storage efficiency. When the quantity of the remote sensing images is more than 2,000, the storage time in 2-node and 3-node distributed systems is exponentially increased, thus indicating that there are more than 3 Map tasks and multiple tasks are assigned to some nodes at the same moment, but one node can only execute one Map task at one time, so increasing the number of the nodes in Hadoop distributed system can improve the execution efficiency of the remote sensing image retrieval system.

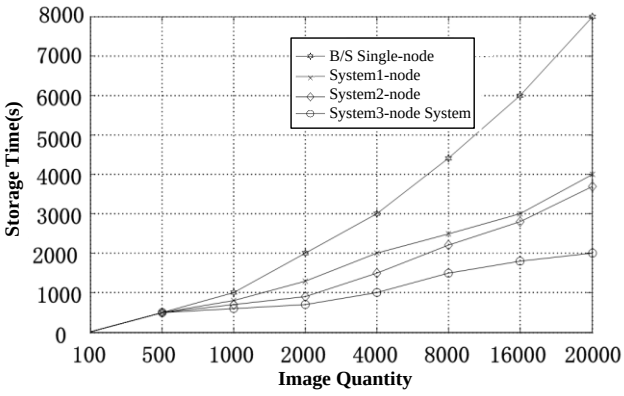

Figure 5. Remote Sensing Image Storage Time Change Curve

4.3. Remote Sensing Image Retrieval Performance Test and Analysis

The remote sensing image retrieval time of different remote sensing image libraries under different node conditions is as shown in Figure 6. According to Figure 6, when there are a small amount of images in the remote sensing image library, the retrieval time in Hadoop multi-node distributed system is longer than that in B/S single-node system and 1-node system, mainly because the adoption of multi-node for parallel computation

increases the computation workload and accordingly prolongs the retrieval time. When the quantity of the images is more than 10,000, the image retrieval time in the multi-node distributed system is obviously less than that in single-node system, mainly because the adoption of MapReduce for parallel computation has the advantage of assigning the image sensing retrieval tasks to multiple nodes in order to improve the remote sensing image retrieval efficiency.

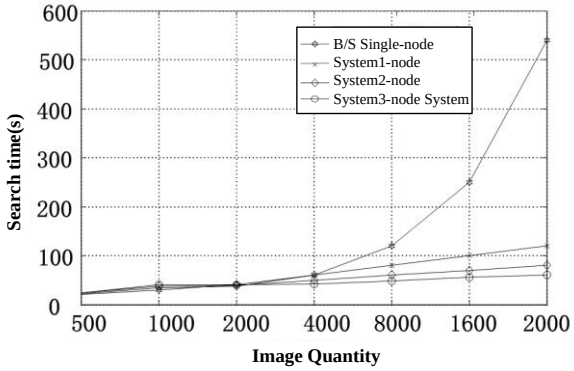

Figure 6. Remote Sensing Image Retrieval Efficiency Comparison

4.4. System Load Test

The remote sensing image retrieval task is submitted to MapReduce distributed system with three nodes. Specifically, the loads of various nodes at different time and under different data volumes are tested, and CPU utilization rates of various nodes are recorded as shown in Figures 7 and 8.

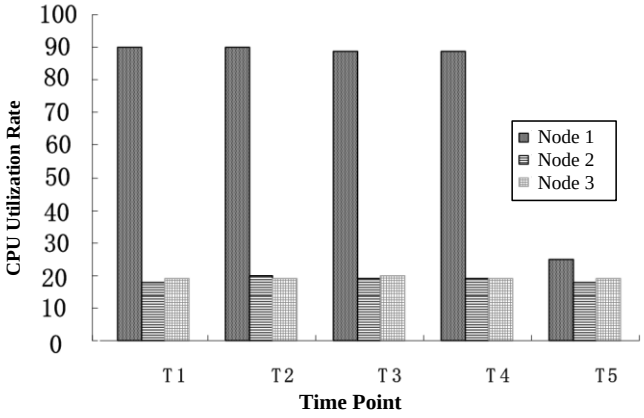

Figure 7. CPU Utilization Rate for Processing 200 Remote Sensing Images

According to Figure 7, when processing a small quantity of remote sensing images (200), due to the small quantity of images, only one Map tasks is assigned to node 1, and node 1 starts to execute Reduce task after finishing the previous task at time t5.

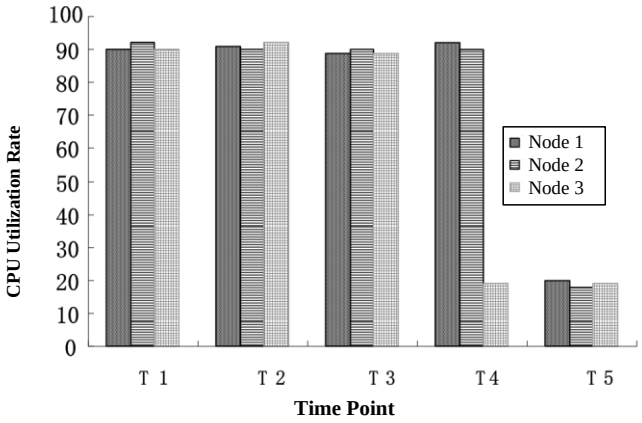

Figure 8. CPU Utilization Rate for Processing 2,000 Remote Sensing Images

According to Figure 8, when processing a large quantity of remote sensing images (2,000), since there are multiple Map tasks to be executed, the three nodes fail to finish Map tasks during the time quantum from T1 to T3; since node 3 finishes Map task and enters in idle state at the time T4, node 3 starts to execute Reduce task so that the task of the heavy-load node can be automatically transferred to the idle node for execution to balance the system load. Meanwhile, due to the mutual coordination of Map and Reduce tasks, the data processing capacity of each node can be fully used to improve the data efficiency of each node.

4.5. Remote Sensing Image Retrieval Result Comparison

Hadoop distributed system and B/S single-node system are adopted to retrieve various kinds of remote sensing images, and the mean retrieval result is as shown in Table 3. According to the following table, Hadoop distributed system is superior to B/S single-node system in the aspects of precision rate and recall ratio, thus indicating that Hadoop distributed system can improve remote sensing image retrieval quality.

Table 3. Remote Sensing Image Retrieval Result Comparison

Different Types	Hadoop Distributed System		B/S Single-node System	
	Precision Rate (%)	Recall Ratio (%)	Precision Rate (%)	Recall Ratio (%)
Vegetations	93.36	77.96	91.50	76.92
Wastelands	87.61	79.99	86.44	77.18
Houses	81.96	70.89	79.52	69.30
Lakes	84.37	67.86	82.33	66.59
Rivers	75.80	65.31	74.97	64.24
Roads and Squares	81.05	60.53	79.41	58.74

5. Conclusion

In allusion to the low retrieval efficiency of the traditional methods for retrieving massive remote sensing images, MapReduce based remote sensing image retrieval algorithm is proposed in this article according to the advantages of Hadoop distributed system. The test result shows that the proposed algorithm can rapidly and accurately retrieve the remote sensing images, thus not only improving the remote sensing image

retrieval efficiency, but also improving the remote sensing retrieval accuracy. Therefore, the proposed algorithm has wide application prospect in the automatic retrieval of remote sensing images.

References

[1] J. Hu and Z. Gao, "Modules identification in gene positive networks of hepatocellular carcinoma using Pearson agglomerative method and Pearson cohesion coupling modularity", Journal of Applied Mathematics, vol. 2012, **(2012)**.
[2] Z. Lv, A. Tek and F. Da Silva, "Game on, science-how video game technology may help biologists tackle visualization challenges", PloS one, vol. 8, no. 3, **(2013)**, pp. 57990.
[3] T. Su, W. Wang and Z. Lv, "Rapid Delaunay triangulation for randomly distributed point cloud data using adaptive Hilbert curve", Computers & Graphics, vol. 54, **(2016)**, pp. 65-74.
[4] D. Zeng and Y. Geng, "Content distribution mechanism in mobile P2P network", Journal of Networks, vol. 9, no. 5, **(2014)**, pp. 1229-1236.
[5] W. Gu, Z. Lv and M. Hao, "Change detection method for remote sensing images based on an improved Markov random field", Multimedia Tools and Applications, **(2015)**, pp. 1-16.
[6] Z. Chen, W. Huang and Z. Lv, "Towards a face recognition method based on uncorrelated discriminant sparse preserving projection", Multimedia Tools and Applications, **(2015)**, pp. 1-15.
[7] X. Song and Y. Geng, "Distributed community detection optimization algorithm for complex networks", Journal of Networks, vol. 9, no. 10, **(2014)**, pp. 2758-2765.
[8] D. Jiang, X. Ying and Y. Han, "Collaborative multi-hop routing in cognitive wireless networks", Wireless Personal Communications, **(2015)**, pp. 1-23.
[9] Z. Lv, A. Halawani and S. Feng, "Multimodal hand and foot gesture interaction for handheld devices", ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), vol. 11, no. 1, **(2014)**, pp. 10.
[10] G. Liu, Y. Geng and K. Pahlavan, "Effects of calibration RFID tags on performance of inertial navigation in indoor environment", 2015 International Conference on Computing, Networking and Communications (ICNC), **(2015)**.
[11] J. He, Y. Geng, Y. Wan, S. Li and K. Pahlavan, "A cyber physical test-bed for virtualization of RF access environment for body sensor network", IEEE Sensor Journal, vol. 13, no. 10, **(2013)**, pp. 3826-3836.
[12] W. Huang and Y. Geng, "Identification Method of Attack Path Based on Immune Intrusion Detection", Journal of Networks, vol. 9, no. 4, pp. 964-971, **(2014)**.
[13] X. Li, Z. Lv and J. Hu, "XEarth: A 3D GIS Platform for managing massive city information", Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 2015 IEEE International Conference on. IEEE, **(2015)**, pp. 1-6.
[14] J. He, Y. Geng, F. Liu and C. Xu, "CC-KF: Enhanced TOA Performance in Multipath and NLOS Indoor Extreme Environment", IEEE Sensor Journal, vol. 14, no. 11, **(2014)**, pp. 3766-3774.
[15] N. Lu, C. Lu, Z. Yang and Y. Geng, "Modeling Framework for Mining Lifecycle Management", Journal of Networks, vol. 9, no. 3, **(2014)**, pp. 719-725.
[16] M. Zhou, G. Bao, Y. Geng, B. Alkandari and X. Li, "Polyp detection and radius measurement in small intestine using video capsule endoscopy", 2014 7th International Conference on Biomedical Engineering and Informatics (BMEI), **(2014)**.
[17] G. Yan, Y. Lv, Q. Wang and Y. Geng, "Routing algorithm based on delay rate in wireless cognitive radio network", Journal of Networks, vol. 9, no. 4, **(2014)**, pp. 948-955.
[18] Y. Lin, J. Yang and Z. Lv, "A Self-Assessment Stereo Capture Model Applicable to the Internet of Things", Sensors, vol. 15, no. 8, **(2015)**, pp. 20925-20944.
[19] K. Wang, X. Zhou and T. Li, "Optimizing load balancing and data-locality with data-aware scheduling", Big Data (Big Data), 2014 IEEE International Conference on. IEEE, **(2014)**, pp. 119-128.
[20] L. Zhang, B. He and J. Sun, "Double Image Multi-Encryption Algorithm Based on Fractional Chaotic Time Series", Journal of Computational and Theoretical Nanoscience, vol. 12, **(2015)**, pp. 1-7.
[21] T. Su, Z. Lv and S. Gao, "3d seabed: 3d modeling and visualization platform for the seabed", Multimedia and Expo Workshops (ICMEW), 2014 IEEE International Conference on. IEEE, **(2014)**, pp. 1-6.
[22] Y. Geng, J. Chen, R. Fu, G. Bao and K. Pahlavan, "Enlighten wearable physiological monitoring systems: On-body rf characteristics based human motion classification using a support vector machine", IEEE transactions on mobile computing, vol. 1, no. 1, **(2015)**.
[23] S. Zhou, L. Mi, H. Chen and Y. Geng, "Building detection in Digital surface model", 2013 IEEE International Conference on Imaging Systems and Techniques (IST), **(2012)**.
[24] J. He, Y. Geng and K. Pahlavan, "Toward Accurate Human Tracking: Modeling Time-of-Arrival for Wireless Wearable Sensors in Multipath Environment", IEEE Sensor Journal, vol. 14, no. 11, **(2014)**, pp. 3996-4006.

[25] Z. Lv, A. Halawani and S. Fen, "Touch-less Interactive Augmented Reality Game on Vision Based Wearable Device", Personal and Ubiquitous Computing, vol. 19, no. 3, **(2015)**, pp. 551-567.
[26] G. Bao, L. Mi, Y. Geng, M. Zhou and K. Pahlavan, "A video-based speed estimation technique for localizing the wireless capsule endoscope inside gastrointestinal tract", 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), **(2014)**.

Authors

Shen Xibing, He graduated from Southwest University, Physical geography professional, received master degree in science degree. Now he is a lecturer at Qinzhou University in Guangxi. His main research direction is the regional development and urban geographic information system. He has published academic papers.

