Modeling for Web-based Image Processing and JImaging System Implemented Using Medium Model

Ma Weifeng[1],[*] and Li Jun[2]

[1]*School of Information and Electronic Engineering, Zhejiang University of Science and Technology, Hangzhou, 310023, P.R. China;* [2]*Zhejiang Changzheng Professional and Technical College, Hangzhou, 310023, P.R. China*

Abstract: With the increase of image data and the continuous improvement of application requirements, image processing is facing some key problems such as off-site treatment, rapid processing, data sharing, etc. Web computing as an important branch of distributed computing, is the effective way to solve these key problems. This paper proposes three kinds of Web computing model for image processing, which is on the basis of in-depth analysis of the related research work at home and abroad, analyzes the architecture in detail from the aspects of system construction method, application characteristics, etc. Based on the "medium model", the web-based image processing system-JImaging is implemented. Moreover, the key implementation technology is analyzed in detail, such as the cross-platform client, SOAP/XML communication, and security, etc. Lastly, it can be used for the remote sensing image processing, and to obtain certain positive results in accessibility, sharing, and cross-platform.

1. INTRODUCTION

With the advancement of digital image acquisition technology and the rapidly increasing demand for image applications, promote the development of digital image processing technology. On the one hand, the accessed image data become more than before, which needs timely and effective treatment; on the other hand, the area of application continues to expand, which needs new image process method and technique, therefore, higher and newer requirements need to be proposed on image processing system.

Lots of famous image processing software systems are stand-alone or centralized such as Photoshop of Adobe company, ACDSee of ACDSystem company, PhotoStyler of Taiwan Ulead company and ERDAS of remote sensing image processing software. Although, these software systems are with efficient design, high stability, powerful, etc., but these systems also have their own shortcomings: a) the stand-alone software must be purchased and downloaded, then installed manually. The users cannot get the easy, remote, online services of image processing; 2) these software systems are developed in special operating system, so that these systems have the disability to cross-platform and poor scalability. Distributed image processing technology has become a hotspot of current research. Web Computing, as a new kind of network computing method, is based on Client/Server mode and uses the Web browser as user interface, and can combine a variety of distributed computing generics. Web computing is an extension of distributed computing, its

emergence will eventually extend distributed computing to the Internet, and increasingly becoming an important platform for distributed computing. As the Internet technology is advancing, Web computing is becoming widely popular. The related implementation technique of Web computing makes the digital images Web computing possible. At present, at home and abroad, the image Web computing mostly concentrated in areas such as telemedicine, but there are very few systemic research in the architecture, implementation technology, etc. This paper proposes the image Web computing architecture and related three models, and explores specific implementation technology based on the practical research work.

2. THREE MODELS OF WEB-BASED IMAGE COMPUTING ARCHITECTURE

Under the theoretical guidance of the Web computing application framework, according to the characteristics of image processing applications, we propose the distributed image processing architecture based on Web.

In the face of the requirement of image processing applications, image processing functions can be deployed on the client or distributed on the server; therefore, according to the difference of the deployment location of the image processing functions and the client implementation technology, we propose three kinds of distributed image processing models. The following will elaborate and analyze these three kinds of models in detail.

2.1. Lightweight Model

It can be observed from Fig. (**1**) that in the lightweight architecture, all image processing functions are deployed on
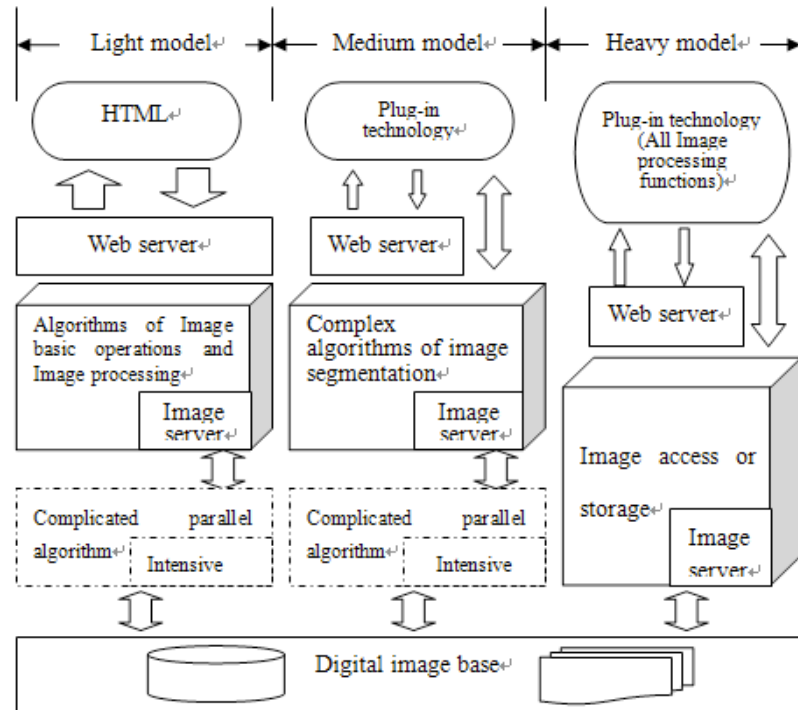
Fig. (1). Three models of the image web computing.

the server, the client does not require any image processing logic, it can be achieved by using simple HTML page technology, that is why it is called "lightweight model".

In this architecture, clients need to achieve simple functions, just show the information to user about which image processing functions and which image data resources provided by the server, and provide interface to user, allow users to submit the image data and associated processing algorithms parameters which they need to deal with, then send processing business request to the server through HTTP and other data transport protocol. Finally, the result that is accessed from image processing server can be previewed in the web page or available for download.

When the Web server receives the user's request, it will analyze the protocol and send the request to the image processing server. Image processing server deploys image processing functions, it provide the corresponding image processing services on a user's request. For the part of the complex processing algorithms which server cannot content computing resources, we can build intensive computing environment, such as cluster computing environment, etc, to implement algorithm's parallel processing and improve the speed. At the same time, the image processing server obtains the operations of the image data resources and associated metadata through the data access interface.

As this lightweight architecture's processing logic is concentrated on the server side, therefore, it is easy to develop the client. What we need to do is to focus on the business logic development of the serve side, and then it is very simple and quick to construct image processing applications. However, precisely because of this over-concentration mechanism, the communication between client side and server side greatly increases, for example, if users want to make a simple geometric operation in image, it also needs a

complete communication. Therefore, this architecture is not suitable for real-time image processing applications which always need interaction. It is suitable for the application areas which require coarse-grained images processing services.

2.2. Medium Model

In the face of the problem which exists in the lightweight architecture, we propose the medium model architecture, try to shift the over-concentration image processing functions from the server to the client, and minimize their communication, provide an efficient, real-time distributed image processing application for reference.

As the client has some functions of the image processing, therefore, the implementation technologies cannot choose the form of HTML, Web pictures, etc, which is just a simple information display. Instead, we should choose the shadow of desktop programs which have a little function of logic processing. At present, the main implementation method is using the network plug-in technology.

Through using the network plug-in technology, develop a common data model of client-side image processing, and integrate the basic algorithm of reading and writing image formats and simple image processing. When accessing image processing applications through the browser, the client will automatically load the data model plug-in, also, it can communicate and exchange data with the server through some high-level protocol such as SOCKET, SOAP (Simple Object Access Protocol), or distributed object computing technology.

With this architecture, the image processing server can focus on some image processing algorithms which are more advanced and associated with specific applications. And the same is the case with the lightweight architecture, when
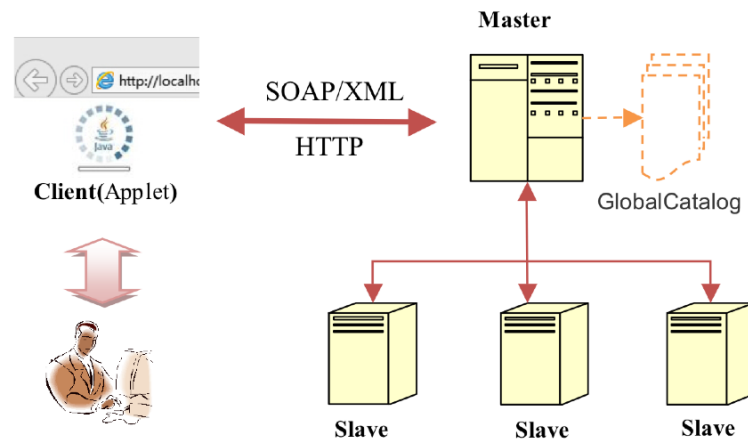
Fig. (2). The architecture of JImaging system.

there is a very complex image processing algorithm, the intensive calculation method can be used to parallel compute through deploying more than a few nodes [1, 2].

The biggest advantage of this architecture is that there is a common data model, when you want to build different image processing applications, only need to make few adjustments of image processing server. Therefore, it has versatility, and the basic image processing functions provided by the client-side data model can content the basic service needs better, which significantly reduce the frequency communication problem between client and server, also it is suitable for all kinds of image processing applications. However, this architecture also has problems that it is difficult to design and develop.

2.3. Heavyweight Model

Contrast to the lightweight architecture, the heavyweight architecture deploys all the functions of the image processing to the client, but the server-side only to realize the operations of image resources and meta-data resources.

Under this architecture, because the image processing functions are integrated in the client-side data model, the amount of communication between the client and server can be minimized. However, it is more difficult than the medium model to design and develop client-side data model with using the network plug-in technology, for example, the multi-threaded computing to solve complex algorithms, the multiple cache technology of image processing, etc. At the same time, for the distributed computing applications which make the browser as a user interface, its client's computing power is lower than desktop application program. Therefore, this architecture is suitable for the image processing applications which do not require a large number of complex calculations. Viewing from the function, this image application system is "lightweight applications".

3. JIMAGING-A WEB-BASED IMAGE PROCESSING SYSTEM

JImaging is a web-based image processing system which is based on the "medium model", and its structural design is shown in Fig. (2).

From Fig. (2), JImaging system adopts the architecture implementation of JavaEE. The client adopts the java applet.

Combining with the JAI image manipulation package, the client provides with the image data management and organization, basic operation of image, and some basic image processing functions. The users can run the client of applet through web explorer and Java Web Start to obtain simple and basic image processing service. For the complex image processing service, the client will submit the request, and interact with the remote server cluster through the SOAP. The server cluster consists of master node and slave node. The master node is responsible for the dispatching of the request, file and image processing service, catalog mirroring, etc. While the slave node takes charge of the image processing algorithm and provides the complex and professional image processing algorithm, and the working process receives the dynamic scheduling of master node, implements the requested image processing service, and returns the processing results to the master node. Meanwhile, different kinds of slave server nodes can be implemented and deployed to satisfy the practical need of engineering, according to the application requirements of different images.

4. JIMAGING SYSTEM IMPLEMENTATION

4.1. Cross-platform Client

The key point in the process of the implementation technology of the image Web calculation is to support the Web cross-platform image processing technology. At present, most image processing systems are based on the C/C++ implementation, so they are hard to be upgraded to Web computing system. Luckily, SUN company always pays much attention to the image and image processing technology, and its Java language supports the image processing as it is officially launched, and it goes through three stages: Java AWT; Java 2D API; and Java JAI (Java Advanced Imaging) [3].

Java AWT and Java 2 DAPI have magnitude limits in image processing, for example, they cannot meet the particular users' requirements of image processing. Therefore, in conjunction with several companies, SUN company puts forward that the JAI advanced image processing is regarded as an extension pack of JDK based on AWT and 2D API. JAI technology has very high performance with the features of cross-platform, distribution, device independence, robustness, etc. JAI provides very strong image processing API, including image operation, pre-processing and image segmentation, etc. Its programming development is totally based

```
PlanarImage src, dst;

Integer [] low, high, map;

……

pb = new ParameterBlock();

pb.addSource(src);

pb.add(low);

pb.add(high);

pb.add(map);

RenderedImage dst = JAI.create("threshold", pb);
```

Fig. (3). JAI codes.

on the object-oriented implementation, for instance, implementing the remote sensing image processing for an overpass by using the threshold slicing algorithm, and the codes are shown in Fig. (**3**).

The client of JImaging adopts the Java Applet, utilizing JAI lib to develop the data model and the basic image processing arithmetic, and its operation effect is shown in Fig. (**4**).

4.2. Data Communication

JImaging system has two forms in data transmission: binary system (image data) and XML text form (parameter files, control information and so on) [4, 5]. XML text form is supported by SOAP, so it can be directly embedded in the part section of the SOAP pack. With the development of the standard of SOAP, there is SOAP pack based on attachment currently, which makes it possible for transmission of the multimedia file. On the Internet, the system passes the data through the SOAP by using the SOAP SDK with attachment of SAAJ provided by SOAP. The JImaging system for data delivery strategy is described in Fig. (**5**).

From the Fig. (**5**), client compresses the several data files (image data, parameter file text, etc.) which are required to be transferred, and packs into a file. Here, our compression algorithm sample is based on the jar algorithmic compression [6-8]. Then, add to the SOAP packet as an attachment, and transfer by using SOAP packet to bind the HTTP protocol. When the data receiver gets the data file and unpacks the packet, then gets the file needed and to process. At the same time, the returning of the results is an opposite processing.
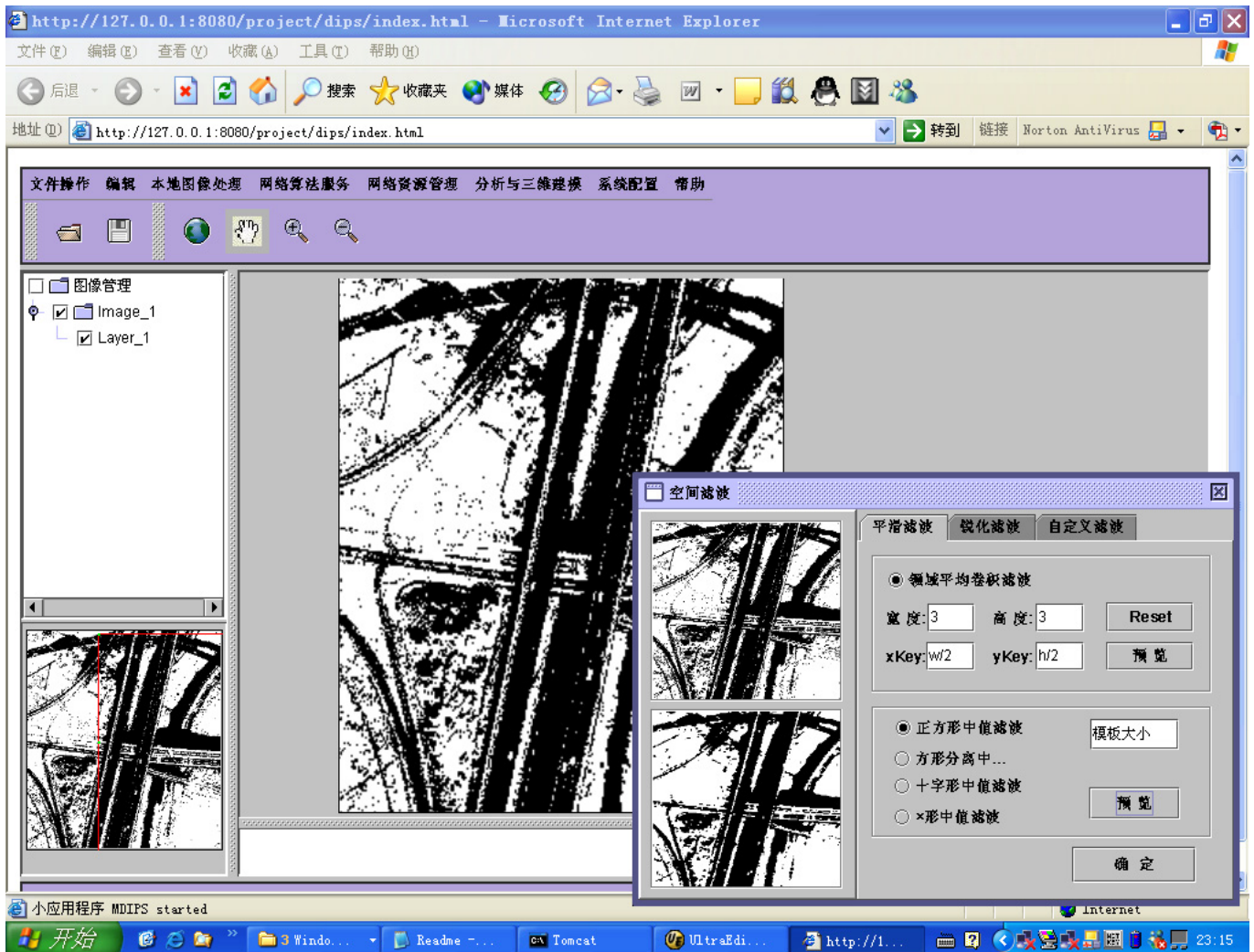

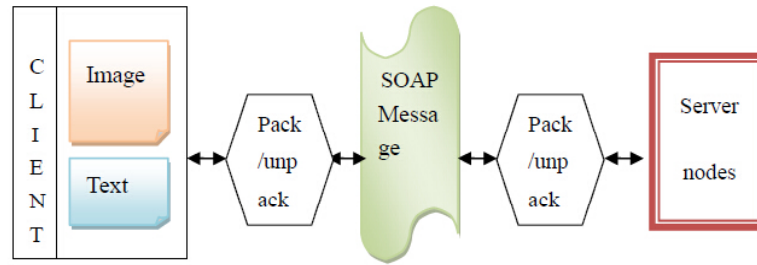**Fig. (4).** The interface of cross-platform client.

Fig. (5). Data transmission process.

Table 1.   Image transmission with SOAP.

	Image Size (k)	Transmission Time (ms)		Speed (k/s)
1	65k	59	64	1015.63
		70		
		63		
2	165k	89	85	1941.18
		81		
		85		
3	2080k	551	601	3460.90
		621		
		631		
4	4098k	1176	1152	3557.29
		1132		
		1148		
5	6150k	1672	1602	3838.95
		1604		
		1530		

4.3. System Security

In JImaging system, the java applet adopted by the client is required to be downloaded to the network client dynamically. In order to ensure the security of the network environment, java adopts the sandbox security mechanism. As installing the program, this security mechanism can govern the security of all levels through reading the configuration information of security policy file POLICY. The programs based on the java applet cannot access the local file and data and other resources by default. Therefore, we configure the POLICY security policy file, and use the digital signature technology to authorize the users to access the local resources, realizing the safe distributed image processing [9].

5. EXPERIMENTAL DATA ANALYSIS

In order to detect the responsibility of the whole system network, the speed of the network communication with SOAP protocol has been tested. Local Area Network with 10M/S has been constructed and adopted in this experiment. It has 5 groups of different size of the image file in the ex-periment, and each group has been tested three times, finally the result takes the average.

When the image data is transmitted based on SOAP protocol, the total time includes SOAP packet processing time (the packing time and the unpacking time) and the actual network transmission time. The SOAP packet processing time is relatively stable, but the actual network transmission time is influenced largely by the size of image data. From Table **1** and Fig. (**6**), when the image data is small, the SOAP packet processing time has larger proportion, which makes the whole transmission rate relatively low. When the image data is large, the SOAP packet processing time has smaller proportion, so that the whole transmission rate is high. The experiment result proved that the network transmission rate of the system can be maintained on 3000-4000k/s, which is adequately fast to response to client.

CONCLUSION

Web computing is an important branch of distributed computing, with the development of the network infrastruc-
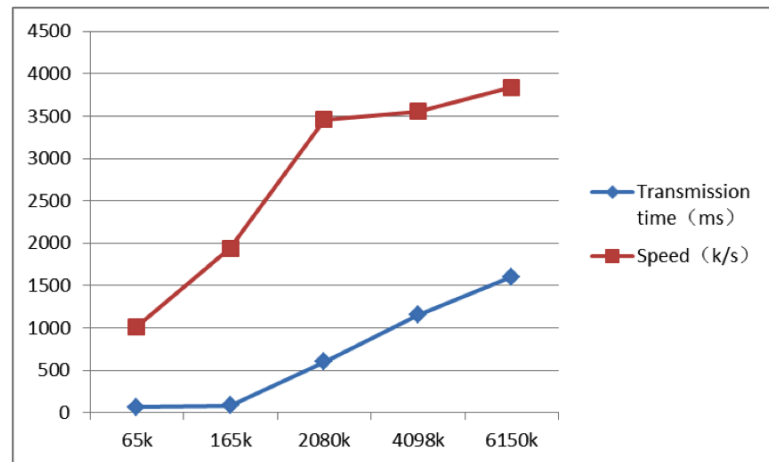
Fig. (6). The curves of transmission time and speed.

ture and Web computing technology, very large number of applications will be extended to the Internet in the future, and the combination of Digital image processing and Web computing will become a trend. This article makes a systematic and detailed analysis on architecture of the image Web computing, and proposes its own model. We have achieved the analysis of the key technologies of image Web computing, and done a preliminary exploration of the image Web computing through programming practice, developed a prototype system. In the future, we will continue to study the relevant content of the image Web computing and improve the system.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1]   M. Weifeng, G. Cen, J. Li, and Z. Shen, "Analysis on high-powered remote sensing image processing and spatial information grid modeling," *Computer Engineering*, vol. 32, no. 5, p. 3, 2006.

[2]   Z. Shen, J. Luo, G. Huang, D. Ming, W. Ma, and S. Hao, "Distributed computing model for processing remotely sensed images based on grid computing," *Information Sciences*, vol. 177, no. 2, pp. 504-518, 15 January 2007.

[3]   SUN. Programming in Java Advanced Imaging Release 1.0.1[DB/OL], November 1999

[4]   R. Santos, "Java advanced imaging API: A tutorial. RITA", Vol. XI Number 1, 2004, Available from: http://www.inf.ufrgs.br/~revista/docs/rita11/rita_v11_n1_p93a124.pdf

[5]   M. Weifeng, J. Li, Z. Xu, and Z. Shen, "Research on distributed geocomputation environment adapting to remotely sensed image processing," In: *Proceedings of the Second International Conference on Computer Science & Education. Wuhan, China,* 2007.

[6]   S. Silva, L. Alcada-Almeida, and L.C. Dias, "Development of a web-based multi-criteria spatial decision support system for the assessment of environmental sustainability of dairy farms," *Computers and Electronics in Agriculture*, vol. 108, pp. 46-57, October 2014.

[7]   A. Maroosi, R. C. Muniyandi, E. Sundararajan, and A. M. Zin, "Parallel and distributed computing models on a graphics processing unit to accelerate simulation of membrane systems," *Simulation Modelling Practice and Theory*, vol. 47, pp. 60-78, September 2014.

[8]   R. de Beer, D. G. Demilly, S. Nastase, and D. van Ormondt, "A distributed computing system for magnetic resonance imaging: Java-based processing and binding of XML," *Computer Methods and Programs in Biomedicine*, vol. 73, no. 3, pp. 221-231, March 2004.

[9]   P. Wang, J. Wang, Y. Chen, and G. Ni, "Rapid processing of remote sensing images based on cloud computing," *Future Generation Computer Systems*, vol. 29, no. 8, pp. 1963-1968, October 2013.