An Internet-Based Telemadicine System

Kyung-Tae Lee, Hyung-Joo Lim, Seong-Hwan Shin
Division of Computer, Information, and Communications Engineering
Kangwon National University, 192-1 Hyoja-dong, Chuncon, Kangwon, Korea

Sang-Wook Kim, Min-Hee Jang

College of Information & Communications
Hanyang University, 17 Haengdang-dong, Seongdong-gu, Seoul 133-791, Korea

Yun-Ho Ahn, Yeo-Dong Yoon
XRay21 Inc. Howon Building 502, 1550-8 Seocho-Dong, Seocho-gu, Seoul 137-873, Korea

Abstract — *This paper addresses development of an Internet-based telemedicine system that permits remote diagnosis of radiologic images. This system has been commercially served by XRay21 inc. for small and/or rural hospitals. The paper discusses technical experiences obtained in developing the system. We performed extensive experiments on our system through many radiologists. The results show that there are nearly no differences in terms of diagnosis accuracy between hard and soft copy images of computed tomograms and X-ray images.*

1. INTRODUCTION

THE Radiologic images are widely used in hospitals for diagnosing and treating patients' diseases. The radiologist accurately reads radiologic images, and thus helps diagnose patients correctly. The number of radiologists is much smaller than is necessary, however.

In some small or rural hospitals, there are no radiologists residing there. In these cases, correct diagnosis and treatment are infeasible. Also, radiologists are not resident at hospitals all day long[1][2][3]. Thus, emergency patients would suffer from the absence of radiologists at night.

XRay21 Inc. and Data & Knowledge Engineering Lab. at Kangwon National University* have been working together since 2000 in order to alleviate the problems above. Currently, the advances of the computer and network technology make it possible to connect a lot of radiologists and hospitals together through the Internet. By fully utilizing the Internet environment, we have developed a telemedicine system, thus permitting remote diagnosis of radiologic images. This paper presents the technical experiences obtained in developing the system.

This paper is organized as follows. Section 2 briefly overviews the overall system architecture of our telemedicine system. Section 3 presents our design decisions and implementation issues obtained in the system development in detail. Finally, Section 4 summarizes and concludes the paper.

2. SYSTEM ARCHITECTURE

Figure 2.1 shows the system architecture of our telemedicine system. Our system consists of sender clients, receiver clients, and a database server, which are all connected with one another through the Internet. A sender corresponds to a hospital, and a receiver corresponds to a

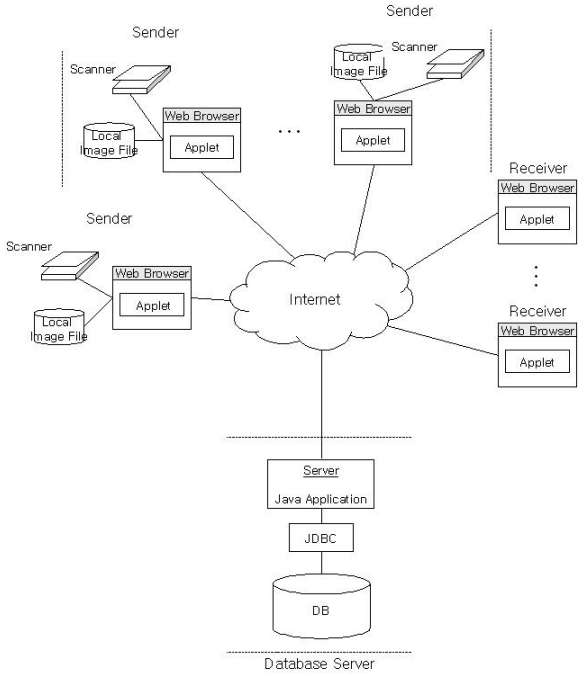

Figure 2.1. System Architecture.

radiologist. A database server plays two roles here: One is to handle requests issued by the sender and receiver clients, and the other is to store/retrieve patients' information into/from a database.

A hospital transmits the patients' information to the database server. The database server receives that information and stores it in the database. A radiologist retrieves that information by requesting it from the database server, reads it, and transmits its diagnosis results back to the database server. Then, the hospital refers to the results diagnosed by the radiologist.

3. DESIGN AND IMPLEMENTATION

Process Architecture

In The client-server model[4] is employed as a process architecture in our system. Clients are implemented via the Java applets[5][6] and classified into the sender applet and the receiver applet.

The sender applet transmits two kinds of information for a patient to the database server: ordinary attributes and radiologic images. The sender applet provides two different interfaces to transmit a radiologic image. The first one is to send an existing file in disk, which was generated by some medical devices. The second one is to send the image file scanned by a scanner on the fly.

In order to absorb all kinds of scanners, we use the TWAIN[7]. The TWAIN is a standard interface(protocol + APIs) for reading radiologic images via imaging devices such as digital cameras and scanners.

The receiver applet allows a radiologist to read and to diagnose the patients' information transmitted by the sender applet. In addition, the receiver applet provides a variety of functions for image processing. The receiver applet manipulates the patient's image appropriately and then transmits it together with its diagnosis results to the database server.

The database server is implemented via the Java application. It waits for the clients' requests, and performs the three actions requested by clients as follows: (1) to store patients' information transmitted by a sender into the database. (2) to retrieve and transmit patients' information requested by a sender or a receiver, (3) to store diagnosis results created by a receiver into the database.

Database Design

The database for our telemedicine system consists of the two basic relations: OrdinaryAttrInfo and ImageInfo. OrdinaryAttr Info contains ordinary attributes such as literal, numerical, and text, which describe a patient. 'name', 'age', 'social security number', 'doctor's opinion' are typical examples. ImageInfo contains the files having real images and additional attributes characterizing each file such as the file size, file format, and so on. Figure 3.1 shows the corresponding relational schema.

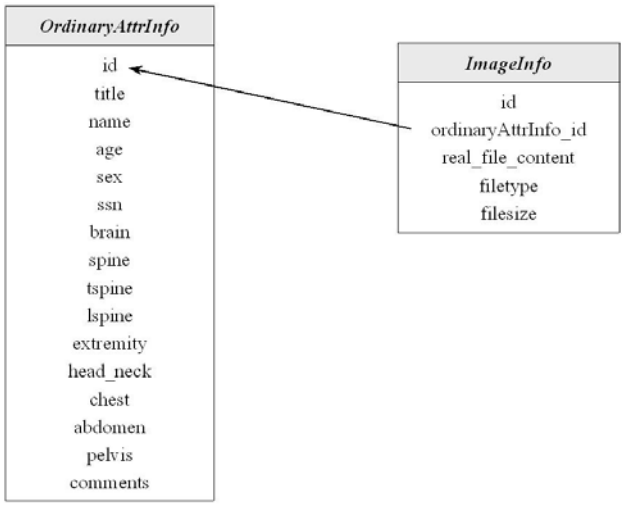

Figure 3.1. Relational Schema.

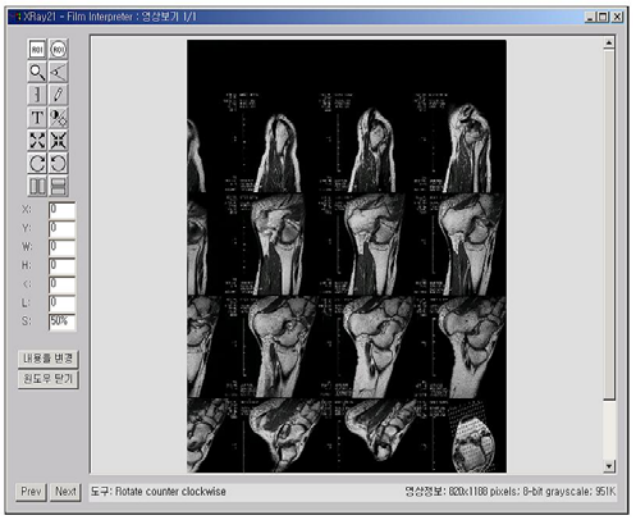

Figure 3.2. Example of Image Processing in the Receiver.

Image Processing

For processing of digital images, ImageJ 1.20s is employed in our implementation. ImageJ developed by NIH(National Institute of Health) is a Java application that targets image processing. Its source code is open to the public[8]. Some ImageJ classes are extended for enriching the functionality of our system.

Our telemedicine system supports various types of image formats such as TIFF[9], GIF[10], JPEG[11], DICOM[12], and BMP[13]. Also, it provides a variety of functions as follows:

(1) Drawing functions: a rectangular ROI(region of interest), a circular ROI, and a text ROI.

(2) Image manipulating functions: rotating, zoom-in, zoom-out, brightness adjusting, and contrast adjusting.

(3) Miscellaneous functions: ruler, protractor, and so on.

Figure 3.2 depicts an example of image processing in the receiver. We see that the large box in the middle shows the radiologic image of a patient, and the small boxes in the left shows buttons for the functions of image processing.

4. CONCLUSIONS

In this paper, we have presented our telemedicine system developed on top of the Internet environment. Our system has been commercially served by XRay21 inc. for small and/or rural hospitals. We have performed extensive experiments on our system through many radiologists. The results show that there are nearly no differences in terms of diagnosis accuracy between hard and soft copy images of computed tomograms and X-ray images. Therefore, our system makes it possible for patients to get correct as well as rapid diagnosis at any hospitals.

ACKNOWLEDGMENT

This research was partially supported by the MIC(Ministry of Information and Communication) of Korea under the ITRC(Information Technology Research Center) support program supervised by the IITA (IITA-2005-C1090-0502-0009).

REFERENCES

[1] P. Slomka, E. Elliott and A. Dredger, "Java-Based Remote Viewing and Processing of Nuclear Medicine Images: Toward the Imaging Department Without Walls," The Journal of Nuclear Medicine, Vol. 41, No. 1, pp. 111-118, Jan. 2000.

[2] S. Lee et al., "Consulting with Radiologists Outside the Hospital by Using Java," infoRAD, Vol. 19, No. 4, pp. 1069-1075, 1999.
[3] J. Wallis, "Java and Teleradiology," The Journal of Nuclear Medicine, Vol. 41, No. 1, pp. 119-122, Aug. 1999.
[4] B. Forouzan, TCP/IP Protocol Suite, McGraw- Hill, 2000.
[5] Sun Microsystems, What Applets Can and Can't Do, http://java.sun.com/docs/books/tutorial/applet/overview/security.html.
[6] Sun Microsystems, How to Sign Applets Using RSA-Signed Certificates, http://java.sun.com/j2se/1.4/docs/guide/plugin/developer_guide/rsa_signing.html.
[7] TWAIN Working Group, About Twain, http://www.twain.org/about.htm.
[8] NIH, Introduction, http://rsb.info.nih.gov/ij/docs/intro.html.
[9] TIFF, Tag(ged) Image File Format, http://home.earthlink.net/~ritter/tiff/.
[10] GIF, Graphics Interchange Format, http://astronomy.swin.edu.au/pbourke/dataformats/gif.
[11] JPEG, Joint Photographic Experts Group, http://www.jpeg.org.
[12] DICOM, Digital Imaging and Communica- tions in Medicine, http://medical.nema.org.
[13] BMP, BMP image format, http://astronomy.swin.edu.au/pbourke/dataformats/bmp.