

## Using Applet–Servlet Communication for Optimizing Window, Level and Crop for DICOM to JPEG Conversion

Aaron W. C. Kamauu, M.D., M.P.H.,<sup>1,2</sup> Scott L. DuVall,<sup>2</sup> Richard H. Wiggins III, M.D.<sup>3</sup>  
and David E. Avrin, M.D., Ph.D.,<sup>3</sup>

In the creation of interesting radiological cases in a digital teaching file, it is necessary to adjust the window and level settings of an image to effectively display the educational focus. The web-based applet described in this paper presents an effective solution for real-time window and level adjustments without leaving the picture archiving and communications system workstation. Optimized images are created, as user-defined parameters are passed between the applet and a servlet on the Health Insurance Portability and Accountability Act-compliant teaching file server.

**KEY WORDS:** Electronic teaching file, image manipulation, web technology

### BACKGROUND

The Radiology Interesting Case Server (RadICS) offers radiology institutions an efficient tool for harvesting interesting cases from picture archiving and communications system (PACS) without requiring any modifications of PACS configurations or workstation software.<sup>1</sup> Although RadICS handles the storage and presentation of study images from PACS, in the vendor-neutral environment, window and level settings continue to present a major obstacle for the conversion of images from the digital imaging and communications in medicine (DICOM) standard file format to the Joint Photographic Experts Group (JPEG) image format, a desktop publishing format.<sup>2</sup>

In RadICS, two DICOM-to-JPEG conversion processes are necessary: first, to generate the original JPEG images with appropriate window and level settings for correct and efficient key image selection from the interesting case; second, to allow real-time modification of the selected key

images with user-defined window and level settings for optimal display of the educational focus (i.e., an abnormality).

The first conversion process is necessary because RadICS does not query or access PACS directly or indirectly, but rather images are pushed from PACS to RadICS.<sup>1</sup> Therefore, RadICS only receives the original, unmodified DICOM image file and does not receive the presentation state of the image. RadICS applies an algorithm to automatically calculate appropriate window and level settings for this first pass image conversion.<sup>3</sup>

This manuscript describes the innovative integration of open source technology and information management in solving the optimum window and level problem encountered in the development of RadICS, a vendor-neutral digital teaching file (DTF), based on ideal characteristics.<sup>4</sup> This paper describes the Java applet and servlet communica-

<sup>1</sup>From the RemedyMD Inc., 9350 South 150 East, Sandy, UT, 84070, USA.

<sup>2</sup>From the Department of Biomedical Informatics, University of Utah, 26 South 2000 East HSEB 5700, Salt Lake City, UT, 84112-5750, USA.

<sup>3</sup>From the Department of Radiology, University of Utah Health Science Center, 30 North 1900 East #1A071, Salt Lake City, UT, 84132-2140, USA.

Correspondence to: Aaron W. C. Kamauu, M.D., M.P.H., Department of Radiology, University of Utah Health Science Center, 30 North 1900 East #1A071, Salt Lake City, UT, 84132-2140, USA, tel: +1-801-9496748; e-mail: siim@akamaau.com

Copyright © 2007 by Society for Imaging Informatics in Medicine

Online publication 30 May 2007

doi: 10.1007/s10278-007-9038-3

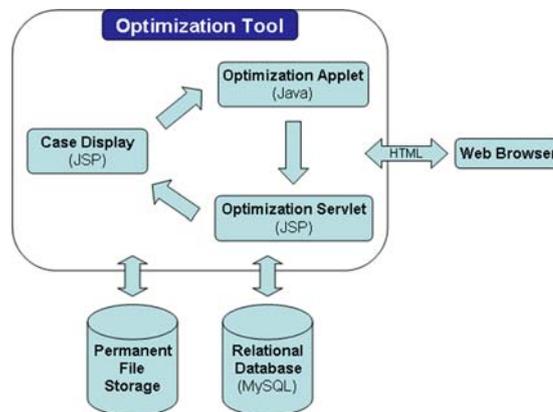


Fig. 1. The case information and images are displayed in the Case Display JSP-driven webpage. When the case author selects to edit an image, the original or resized DICOM image in permanent file storage is opened in the Java-based Optimization Applet. After window and level adjustments are made, the user-defined optimum window and level settings are passed as parameters to the JSP-driven Optimization Servlet and stored in the MySQL relational database. The servlet passes the optimum window and level settings to the OFFIS DCMTK to generate new full-size and thumbnail JPEG images that are then displayed in the Case Display webpage.

tion method developed to provide for user-defined, real-time optimization of selected key images in an interesting case. This optimization tool addresses window and level issues across all medical image modalities.

#### A Word About Applets and Servlets

Java applets were included in the initial release of the Java Software Development Kit (JDK) in 1996. An applet is a Java program that is run in the context of another application, typically a web browser with the Java plug-in. The Java plug-in is part of the Java runtime environment (JRE) and allows Java code to be interpreted inside a web browser. The JRE can be freely downloaded (<http://www.java.com>) and installed on computers running any operating system. Applets are downloaded to the client machine and run using the JRE. Although applets have access to server resources and can communicate with servlets running on the host server, security restrictions prevent applets from reading or writing files, running programs, or accessing information on the local client system.<sup>5</sup>

Servlets are Java programs that run on a web server to translate dynamic content into hypertext markup language (HTML) pages. They act as a middle layer between client requests and server resources such as data or other applications. Servlet functions are processed on the server;

however, they can be invoked through a network (like the Internet) by a web browser. Only servlet output, formatted as HTML, is returned to the web browser. Javaser pages (JSP) are an extension of servlets that allow dynamic servlet code to be interlaced in static HTML. JSP documents are translated into regular servlets by the web server the first time the page is requested. A single instance of the Java interpreter, called the Java virtual machine, remains active in memory on the server and creates a light-weight Java thread to handle new client requests.<sup>6</sup>

#### MATERIALS AND METHODS

When key images are selected for an interesting case in RadICS, the original DICOM file is archived with the associated, newly generated JPEG files in permanent file storage.<sup>7</sup> After the case creation is complete, the case information and images are displayed in the Case Display, JSP-driven webpage (Fig. 1). By default, the large version of the first image in the case is shown; clicking a thumbnail image will show its larger version (Fig. 2). For case authors, an “Edit Image” link is present below the large image on the webpage. When the link is clicked, the case author is presented with a modified, open source Java DICOM viewer applet (Fig. 3).

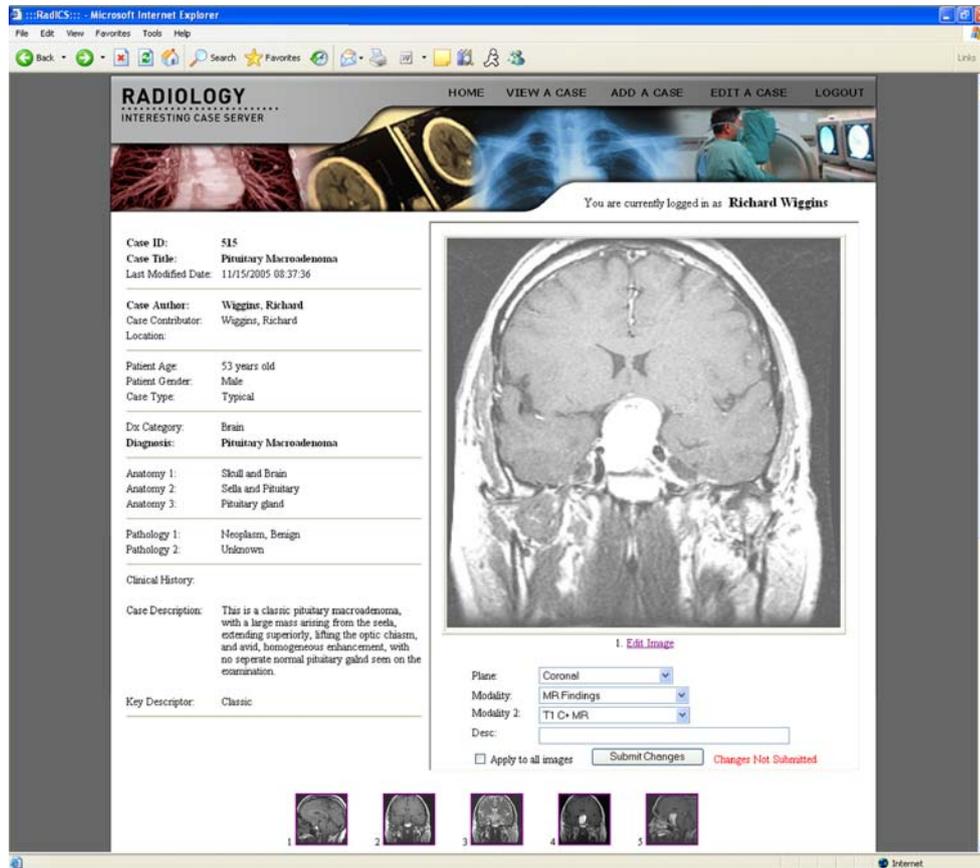


Fig. 2. A completed interesting case in the RadICS, prior to optimization of the images.

The open source Java DICOM viewer was obtained from the Nagoya Institute of Technology and modified to display only the selected image.<sup>8</sup> The Java DICOM viewer is embedded as an applet in a JSP webpage, allowing the optimization tool to be presented to the author via any commonly used web browser. The only requirement is installation of the JRE version 5.0 or more recent. It was also modified to allow exchange of user-defined crop and window and level settings with servlets. Window and level values and cropping boundaries for the selected image are retrieved from the database and dynamically added as parameters nested under the HTML APPLET tag. If window and level values are not set in the database, the WindowLevel and WindowWidth DICOM header elements are used. Parameters are also set to communicate the unique case and image identifiers and modality of the selected image to the applet. The values of these

HTML parameters are accessed within the applet using the getParameter method. As the DICOM viewer begins to run, it uses the window and level parameter values as the image defaults. After displaying the full image with these settings, a rectangle representing the cropping boundary is overlaid. In the absence of cropping boundary values, the image is displayed without an overlay.

There is a wide range of DICOM image sizes (i.e., typical magnetic resonance images are  $256 \times 256$  pixels, whereas typical chest plain films are  $1,760 \times 2,140$  pixels). Displaying large images in the applet requires a significant amount of memory resources on the client workstation because all image data are represented as Java objects, not downloaded as separate temporary image files. In addition, the spatial resolution limitations of displaying the entire web page on the screen without needing to scroll would require larger images to be scaled down. To address this, the resolution of the

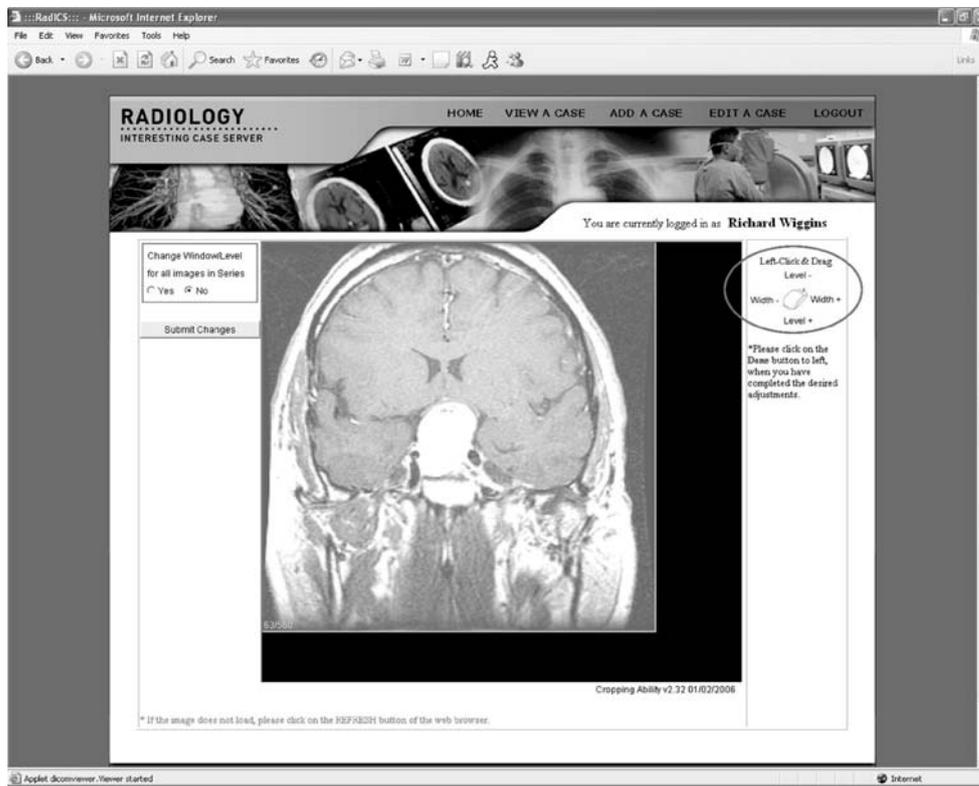


Fig. 3. When the radiologist clicks on the “Edit Image” link, the selected image is displayed in a modified, open-source Java DICOM viewer applet. The radiologist is able to adjust the window and level of the image according to the diagram provided (circle).

selected image is determined by row and column elements in the DICOM header. These values are stored in the MySQL database at the time of key image selection. If the image is  $512 \times 512$  or smaller, the DICOM image data is displayed in the applet at actual size. If the image is larger than  $512 \times 512$ , the image is scaled down and stored as a new DICOM file. For rectangular images, the longest dimension is resized to 512 pixels and the other dimension is scaled proportionally. This smaller DICOM image is displayed in the applet.

The applet provides the user with “windowing” and “leveling” functionality by clicking and holding the left mouse button while dragging across the image. Users are provided this instruction as an adjacent diagram on the same web page. (Fig. 3-circle). The applet shows the window and level changes immediately in a real-time, PACS-like fashion familiar to radiologists. In addition, the window and level values are displayed as text

in the bottom left corner of the applet for reference. The user can define cropping boundaries by clicking and holding the right mouse button while dragging across the image to create a square or rectangle around the desired focus of the image. An overlay is used to show cropping boundaries instead of displaying only image data within the boundaries. This allows the user to redefine more appropriate cropping boundaries in context of the full image. The relative  $x$  and  $y$  position along with the width and height of the cropping boundary are stored within the applet. The  $x$  and  $y$  position identify the pixel that will be the top, left pixel of the cropped image. The width and height of the boundary designate the number of pixels that enclose the desired focus of the image.

The case author is given the option to apply the window and level settings to all images in the same series or just to the selected image. The author is instructed to click on the “Submit Changes”

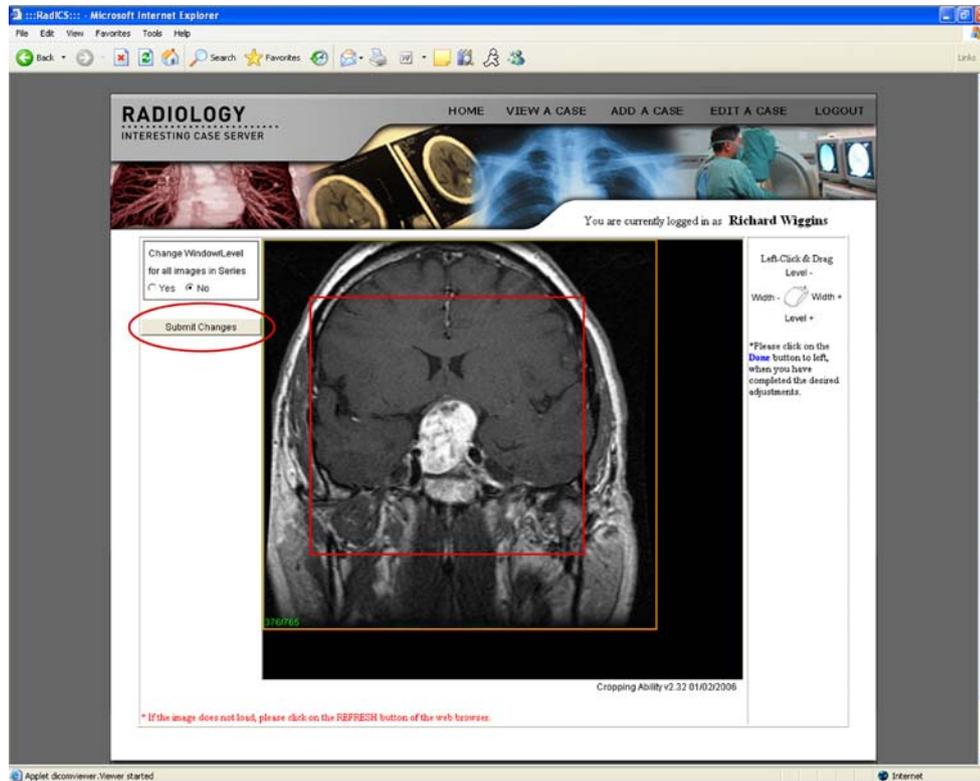


Fig. 4. After making the desired changes, the radiologist clicks on the “Submit Changes” button in the applet (*circle*). New thumbnail and full-size JPEG images are generated using the user-defined, optimum window and level settings.

button in the applet after adjustments are made (Fig. 4-circle). Clicking this button triggers the applet to create a uniform resource locator (URL) with parameters for window and level and crop settings, and the unique case identifier, image identifier, and image modality and whether to apply the window and level settings to all images in the same series. The applet passes this URL to the browser using the showDocument method. A servlet, invoked by the browser request, uses the URL parameters to generate new thumbnail and full-size JPEG images that reflect the desired window and level settings and crop adjustments. The DICOM-to-JPEG conversion is done using the OFFIS DICOM Toolkit (DCMTK).<sup>9</sup> The new JPEG images are then cropped using ImageMagick, an open-source tool commonly installed on Linux servers.<sup>10</sup> Although the resized DICOM image is displayed in the applet, all optimization adjustments are performed using the original full-size DICOM file, which always remains unchanged. The thumbnail JPEG is proportionally constrained to 128 pixels on its longest dimension.

The larger JPEG file maintains the same spatial resolution as the full-size DICOM file (Fig. 5).

## RESULTS

Use of the Java DICOM viewer applet and associated servlet was effective in providing a window and level optimization tool to case authors at the point of study interpretation, primarily at the PACS workstation, while only requiring installation of the standard JRE. This tool was successfully implemented in RadICS, which is currently in use in a large academic radiology department.

## DISCUSSION

By using Java applet technology, radiologists can view DICOM images on any workstation with a common web browser and the JRE. No other applications are required to be installed on the

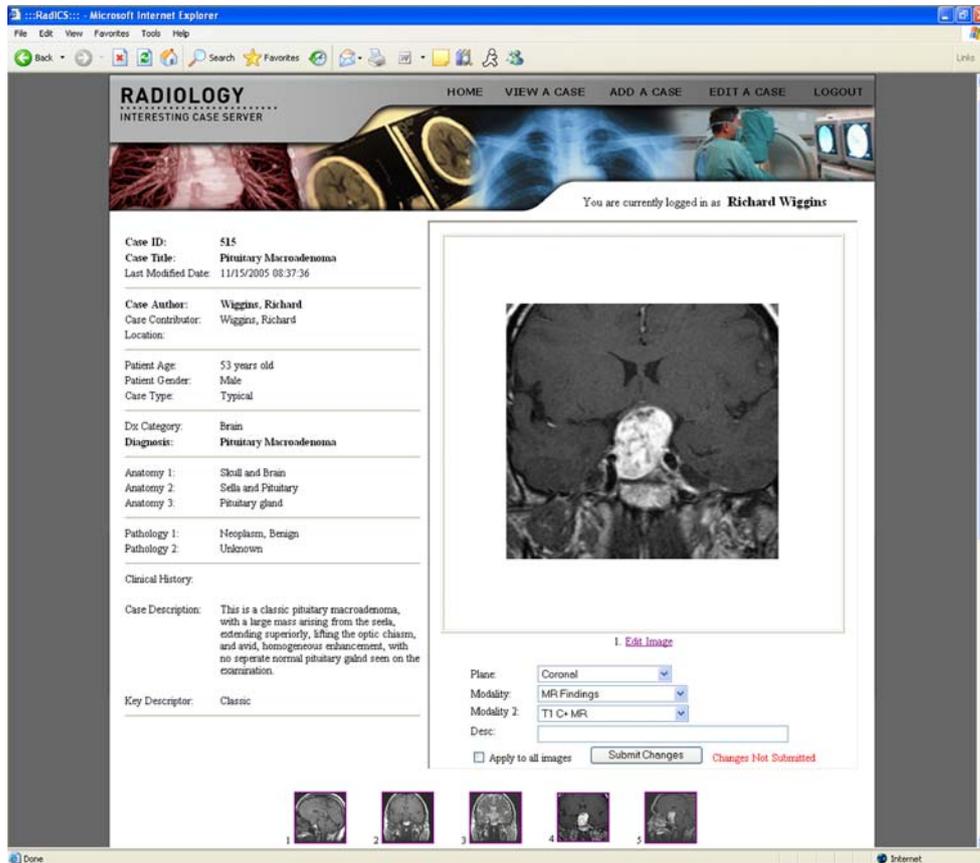


Fig. 5. The radiologist is instantaneously directed back to the interesting case in the RadICS displaying the newly generated JPEG images.

client workstation to use the optimization tool. Displaying DICOM images in the applet keeps all protected health information (PHI) on the server behind a Health Insurance Portability and Accountability Act-compliant firewall. This is because only the pixel data is loaded into a Java class on the server and sent to the applet. The actual DICOM file, with its associated metadata, is never sent from the server to the browser, and no PHI ever leaves the server. Therefore, the applet displays the DICOM image without the server really sending the DICOM file.

These benefits are particularly significant for use of RadICS on the control monitor of the PACS workstation. Many PACS vendors do not allow the installation of a DICOM viewer application or image manipulation software on their workstation; however, in any DTF, it is essential to provide this functionality at the time of interpretation.<sup>7,11</sup> The moment an interesting case

is identified on PACS, the user can send it to RadICS, select the key images, complete the coded and free text case information, crop the image, and optimize the window and level settings without leaving the workstation.

The use of a resized DICOM image file aids in maintaining efficient performance speeds by limiting the size of the applet. In addition, it allows for a standard applet and webpage size, thus displaying the entire image at once without needing to scroll horizontally or vertically. Although, the resized DICOM image is displayed in the applet, generation of new JPEG image files is always performed using the original DICOM image file.

### CONCLUSION

This window and level optimization tool is seamlessly integrated into the case authoring

process of a radiology digital teaching file.<sup>1</sup> In RadICS, the automatic window and level calculation algorithm and preset values provide appropriate window and level settings for the initial JPEG generation, whereas the applet provides the ability to optimize case images for effective display of the educational focus (i.e., abnormality).<sup>3</sup> The applet–servlet communication method provides an effective tool for optimization of image window and level settings in converting DICOM images into a desktop publishing format.

This optimization tool functions on all grayscale DICOM images across all modalities and anatomic locations. Window and level adjustment functionality is easy to use, as it closely resembles the process used on a typical PACS workstation.

#### ACKNOWLEDGMENTS

This project was partially supported by the NIH NLM 2T15 LM07124 training grant and Amirsys<sup>®</sup>. Authors DEA and RHW are consultants to and stockholders in Amirsys<sup>®</sup>.

#### REFERENCES

1. Kamauu AWC, DuVall SL, Robison RJ, Liimatta AP, Wiggins RH, 3rd, Avrin DE: Informatics in radiology (infoRAD): Vendor-neutral case input into a server-based digital teaching file system. *Radiographics* 26(6):1877–1885, 2006 Nov–Dec
2. Wiggins RH, 3rd, Davidson HC, Harnsberger HR, Lauman JR, Goede PA: Image file formats: past, present, and future. *Radiographics* 21(3):789–798, 2001 May–Jun
3. Kamauu AWC, DuVall SL, Liimatta A, Avrin DE, Wiggins RH, 3rd: Automatic window-level calculation for DICOM to JPEG conversion for vendor-neutral teaching file web application. Paper presented at: Radiological Society of North America (RSNA), McCormick Place, Chicago, IL, 2005
4. Scarsbrook AF, Foley PT, Perriss RW, Graham RN: Radiological digital teaching file development: an overview. *Clin Radiol* 60(8):831–837, 2005 Aug
5. Lowe D, Murach J, Steelman A: Murach's beginning Java 2 JDK 5. Mike Murach and Associates, 2005
6. Hall M: Core Servlets and JavaServer Pages, 1st edition: Sun Microsystems Press/Prentice Hall PTR, 2000
7. Kamauu AWC, DuVall SL, Robison RJ, Liimatta AP, Wiggins RH, Avrin DE: Vendor-Neutral Case Input to a Server-based Digital Teaching File System. *Radiographics*. Nov–Dec 2006
8. Muto K, Emoto Y, Katohji T, Nagashima H, Iwata A, Koga S: PC-based Web-oriented DICOM Server: The "DIY" DICOM Server—Cost-effective, High Performance and Easy to Customize. Paper presented at: Radiological Society of North America (RSNA), Chicago, IL, 2000
9. OFFIS: OFFIS DICOM Software, DCMTK—DICOM Toolkit. <http://dicom.offis.de/dcmtdk.php.en>. Accessed March 6, 2007
10. LLC IS: ImageMagick. <http://www.imagemagick.org>. Accessed March 6, 2007
11. Avrin DE, Andriole KP, Arenson RL: Integration of a digital teaching file at the diagnostic workstation. Paper presented at: Society of Computer Applications in Radiology (SCAR), June 6–9, 1996, Denver, CO, 1996