

Java Internet Viewer: A WWW Tool for Remote 3D Medical Image Data Visualization and Comparison

Chris A. Cocosco and Alan C. Evans

McConnell Brain Imaging Centre, Montreal Neurological Institute,
McGill University, Montreal, Canada
c.cocosco@ieee.org
<http://www.bic.mni.mcgill.ca/users/crisco/jiv/>

Introduction: There is a growing need in the research and clinical medical imaging community for Internet-capable tools that facilitate remote data dissemination and interaction. 3-dimensional (3D) medical imaging datasets typically require special-purpose, non-portable, software to be installed and maintained on each workstation. Internet technologies have potential for improving this.

We developed JIV: a powerful, robust, portable, extensible, and open-source Java application (“applet”) for visualization and side-by-side comparison of multiple 3D image datasets. It is designed to work through the WWW; it only requires a common Web browser, and can cope with slow networks and less capable workstations. Moreover, JIV provides features and a level of performance usually only found in traditional stand-alone workstation applications.

Previously reported projects [1,2,3] only have limited user interface functionality. Furthermore, they download slice images only when and if required; thus, their interactive performance is unsatisfactory when used over common long-distance Internet connections (which cannot guarantee a high transfer rate).

Design & Implementation: A convenient way to visualize 3D medical imaging datasets is by three orthogonal 2D slices through the same location in the volume. When several image volumes are to be compared, it is desirable to visualize their slices side-by-side, all at the same position in the volume.

With respect to how and when to download the 3D image data, the following three operation modes are supported by JIV:

1. *All up-front*: all of the data is downloaded and stored in client’s memory before the user can view and interact with any of it.
2. *On demand*: download slice image data only when and if the user wants to view that particular slice.
3. *Hybrid (background download)*: first download only the slices required by the initial cursor positions; then continue downloading all of the data in a background thread (as in (1)); if the user requests slice images which are not already downloaded, they will be downloaded with priority (as in (2)).



Mode (1) guarantees the best interactive response of the viewer; however, the user has to wait for all the data to download before the JIV interface becomes available – this can take a long time on a slow network link. Mode (2) minimizes the data downloads and the amount of memory required by the applet, but (like in [1,2,3]) its interactive response time is highly dependent on the server and on the network. Mode (3) is the best compromise for most situations. Our implementation does not temporarily freeze while waiting for data to arrive: instead, it displays a discernible pattern for the image areas it does not yet have, and replaces it with the real image as soon as it becomes available. To our knowledge, no other published work provides this background download feature.

The gray-level image data is displayed using a user-controlled color-mapping; an efficient mouse-based user input scheme provides fast roaming through the volume, and continuously adjustable zoom and pan.

Results: JIV was tested and was proved to be robust on a variety of computer platforms: Linux-i386, SGI IRIX, MacOS, and various MS Windows (Win32) versions. The interactive performance, defined as screen update time following user input, is good when JIV is running on recent PC hardware with enough memory (RAM). In conclusion, JIV is a convenient and platform-independent software for the remote visualization of 3D medical image data; for example, it can be used in remote data processing – when data goes to a central, well-equipped, site for image processing and storage. This software proved useful (both remotely and locally, in our lab) for efficient simultaneous visualization and comparison of many 3D image datasets, such as evaluating the performance of registration or segmentation methods.

Acknowledgements: Peter Neelin; Dr. Alex Zijdenbos; Dr. Louis Collins.

References

1. S. Vetsch, et al. A parallel pc-based visible human slice web server. *The Second Visible Human Project Conference*, 1998.
2. P. Golland, et al. *Computer Aided Surgery*, 4(3):129–43, 1999.
3. J. T. Lee, et al. *NeuroImage*, 11(5):S918, 2000.