

FAST PLATFORM INDEPENDENT WEB-BASED VISUALIZATION OF MRI DATA FOR COMPUTERS, TABLETS AND SMARTPHONES

Daniel Hähn^{1,2}, Nicolas Rannou^{1,2}, Thomas J. Re^{1,3}, P. Ellen Grant^{1,2}, and Rudolph Pienaar^{1,2}

¹Radiology, Boston Children's Hospital, Boston, MA, United States, ²Radiology, Harvard Medical School, Boston, MA, United States, ³Radiology, University of Milan, Milan, Italy

Target Audience: Researchers and physicians who would like to view and study MR volumes anywhere and on any device (phone, tablet, computer), as well as computer scientists who develop post-processing methods for DICOM data of any modality.

Purpose: Highly specialized software is typically required to display MR images or other image data for diagnostic, clinical therapeutic and research purposes. Traditionally, such software has been deployed on specific research and clinical workstations which limit the ability to share data or view data on portable devices. However, recent advances in computer technology have enabled interactive rendering of imaging data directly in web browsers without any required software installation or plugins, facilitating data access across all devices. We present a web based viewer for DICOM data with workstation-like performance.

Methods: Previous web-based radiology applications require either a server-sided backend or locally installed plugins like Java or Flash [1, 2]. In contrast, modern web browser technologies like HTML5 [3] and WebGL [4] as well as faster JavaScript engines allow the development of plugin-free radiology applications running fully on the client. Especially the introduction of typed array structures enable fast parsing of binary data inside the web browser. We developed an open source toolkit to accelerate the use of these technologies for scientific visualization and file parsing [5]. Among others, we studied the DICOM standard and the associated file format [6] to create a parser written in pure JavaScript. When invoked, a minimal subset of DICOM attributes are read to increase the parsing speed, assuming that all files belong to the same series. The header of the first file is parsed for the following data elements as defined by the DICOM standard: PixelData (7fe0,0010), Rows (0028,0010), Columns (0028,0011), BitsAllocated (0028,0100), BitsStored (0028,0101), SamplesPerPixel (0028,0002), PixelSpacing (0028,0030) and InstanceNumber (0020,0013). For subsequent files, only the PixelData, InstanceNumber and the SliceLocation (0020,1041) are scanned to properly order and scale the images along the z-axis. After parsing, a reslicing operation is applied to reshape the volume according to axial, sagittal and coronal views. In order to judge the parsing performance, we performed tests using 10 structural T1 MPRAGE MR head scans acquired on a SIEMENS Tim Trio scanner (3 Tesla) with an average spacing of 1x1x1mm and 256x256x160 voxels dimension. Each scan consists of ~40 MB DICOM data. We visualized this data with our web-based application Slice:Drop (slicedrop.com) [7] and timed the loading operation. For comparison, we loaded the same data into a local installation of the well-known imaging platform 3D Slicer (v.4.2) [8]. Each dataset was loaded 3 times and the fastest result was selected. All tests were performed on a Mac Pro workstation (2x2.26 GHz Quad-Core Intel Xeon CPUs, 16 GB 1066 MHz DDR3 RAM, ATI Radeon HD 5000 GPU 1GB, Mac OS X 10.8.2) using the Google Chrome browser (v.23) to access Slice:Drop.

Results: Using the traditional desktop application 3D Slicer, the average loading time of a whole head MR DICOM series was 5.82 seconds. The average loading time of the same data using the web-based application Slice:Drop was 2.57 seconds, resulting in reduced parsing and loading by over 55.84%.

Discussion: Our experiments show that modern browser technologies enable fast processing of MR data. The developed DICOM parser written in pure JavaScript decreases the loading time significantly when compared to traditional standalone desktop applications. This speed enhancement is not surprising since several restrictions apply to the input: only one series can be read at a time and the full DICOM header is only parsed once and not for every slice. Because a modern web browser is the only requirement, Slice:Drop also runs on mobile devices (smartphones, tablets). The rendering performance of this application was evaluated in previous experiments and was classified as workstation-like [5]. Our developed source code is MIT licensed and publicly available on GitHub [9, 10].

Conclusion: We present an open source web-based radiology application which enables interactive visualization of DICOM data without any required conversion, preprocessing or software installation. Our results show that the loading performance of this application is significantly better than using a traditional desktop application while also facilitating data access across devices.

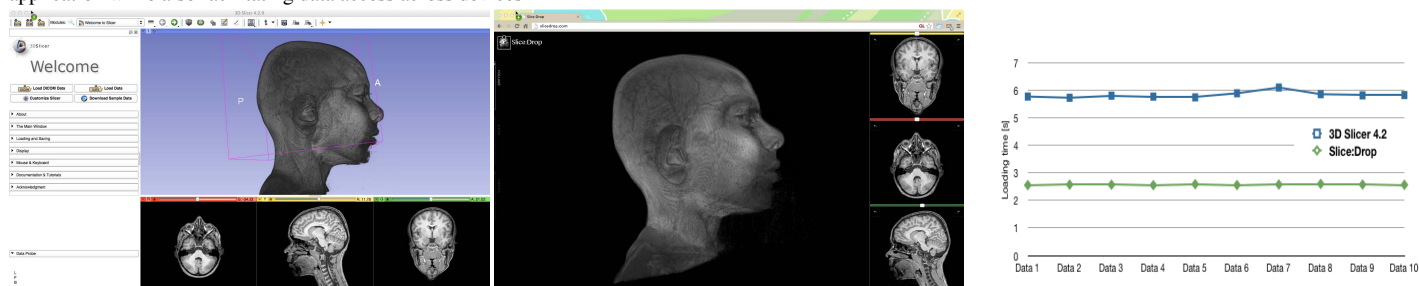


Figure 1. From left to right: 1) A sample MR head scan visualized in 3D Slicer 4.2. 2) the same data visualized in the web-based radiology application Slice:Drop. 3) a comparison of the loading time of 10 different structural T1 scans in both programs.

References:

- [1] Feingold, E.R. et al.: Web-based radiology applications for clinicians and radiologists. Proc. SPIE 3035, Medical Imaging 1997: PACS Design and Evaluation: Engineering and Clinical Issues, 60 (May 22, 1997); doi:10.1117/12.274630.
- [2] Le, T.H. et al.: Thin client (web browser)-based collaboration for medical imaging and web-enabled data. J Digit Imaging, Vol. 15 Suppl 1 (2002), pp. 261-263, doi:10.1007/s10278-002-5049-2.
- [3] Khronos Group, HTML Canvas 2D Context. <http://www.w3.org/TR/2dcontext/> (November 2012)
- [4] Khronos Group, WebGL Specification 1.0. <https://www.khronos.org/registry/webgl/specs/1.0/> (November 2012)
- [5] Hähn D. et al.: Neuroimaging in the Browser using the X Toolkit. Neuroinformatics, September 2012
- [6] The DICOM Standard. <http://medical.nema.org/standard.html> (November 2012)
- [7] Hähn D. et al.: Slice:Drop. IEEE VisWeek, SciVis Poster Session, September 2012.
- [8] Pieper S. et al.: 3D SLICER. Proceedings of the 1st IEEE International Symposium on Biomedical Imaging: From Nano to Macro 2004; 632-635.
- [9] The X Toolkit Source Code Repository. <https://github.com/xtk/X> (November 2012)
- [10] Slice:Drop Source Code Repository. <https://github.com/slicedrop/slicedrop.github.com> (November 2012)