

OPENWALNUT: AN OPEN-SOURCE TOOL FOR VISUALIZATION OF MEDICAL AND BIO-SIGNAL DATA.

Sebastian Eichelbaum¹, Mario Hlawitschka², Gerek Scheuermann¹,

¹Abteilung für Bild- und Signalverarbeitung, Universität Leipzig, Germany

²Lehrstuhl für Wissenschaftliche Visualisierung, Universität Leipzig, Germany

eichelbaum@informatik.uni-leipzig.de

Abstract: *The number of medical imaging modalities and bio-signal acquisition methods has increased dramatically in the last years. Each is designed to answer a certain set of questions or to explore certain features of living tissue. With visualization, it is possible to combine these different types of images and data to grasp their meaning in the context of each other. Unfortunately, many existing visualization tools are focused on certain modalities and signals. With OpenWalnut, we provide a tool which is designed to be used with different signals and modalities in combination with each other. It aims at providing interactive rendering and explorability with a clean data-flow-based interface. The project is open-source and well documented and has yet been extended and used by many groups and researchers. In this short-paper, we provide a coarse overview of the principles and focus-points of OpenWalnut.*

Keywords: *visualization, medical data, bio-signals, open-source, software*

Introduction

In the course of the ongoing research into neurological diseases and the function and anatomy of the brain, a large variety of examination techniques has evolved. The different techniques aim at findings for different research questions or different viewpoints of a single task.

Considering the different applications, it is evident that for many research areas only a combination of these techniques can help answering the posed questions. To name only one example, the combination of diffusion-weighted MRI and functional MRI with an anatomical context provided by T1 MRI images is very common.

To be able to analyze the data measured by the different techniques, a tool that can efficiently visualize the different modalities simultaneously is needed. The software (called *OpenWalnut*) we will present in this work aims at exactly this task. It does not only allow to display the different modalities together, but also provides tools to analyze their interdependence and relations.

In the following, we will point out the focus-points of OpenWalnut and its basic working principles.

Focus and Reasoning

As mentioned above, it is very crucial to handle a multitude of different kinds of images and signals. Besides this, there are several other criteria for visualization software,

especially in a scientific environment, which is our target group of users. There are many visualization tools available which have their specific strengths and weaknesses ([1, 2, 3, 4, 5, 6]), but none of them were able to completely fulfill our criteria. In this section, we point out these criteria, building the fundamentals of OpenWalnut's software design and implementation.

Open Source: In a scientific environment, it is very important to be able to re-produce results of other researchers and to comprehend algorithms of others. With the help of open-source software, this can be achieved. It provides a possibility to share algorithms and calculation pipelines with others in a common framework. Besides this, an open-source framework allows for easy extension and adaption of existing methods to new problems. Thus, OpenWalnut is developed as free and open software to allow us and others to work on a common framework.

General Purpose and Extensible: Very focused software is not able to handle the above mentioned multitude of signal and image modalities. Additionally, in a research environment, it is very often required to find new ways of solving a certain medical or neurological problem. To achieve this, the used software must not limit the researcher in terms of applicability of algorithms and in terms of easy programmatic extensibility. OpenWalnut is two things in this context: (1) a powerful visualization tool and (2) a powerful programming framework which provides maximum flexibility to the researcher. We even allow interfacing OpenWalnut with the famous script language Python.

Re-using existing Tools: There are a lot of very powerful tools and frameworks available for very different tasks. To utilize certain strengths of these toolkits, a software needs to interact with them instead of re-inventing them. OpenWalnut integrates several well-know toolkits like Eigen [7] and OpenSceneGraph [8] and interfaces other tools more indirectly, like FSL [9], teem [10], BioSig [11], NIfTI [12] or, due to our Python bindings, numPy [13], and NiBabel [14]. This way, we make OpenWalnut even more extensible and flexible.

Graphical User Interface: Most software aims either at the visualization researcher or the neuro-scientist, with accordingly designed GUIs. OpenWalnut aims at both groups. The neuro-scientist, who needs a usable tool to handle and visualize its data and the visualization researcher, who needs a powerful, programmable tool and flexible user interface. To achieve this ambitious goal, OpenWalnut focuses on a clean and straight interface, which is centered

around a data-flow network as shown in Figure 1. The principle of using a data-flow networks is very common to many image/signal processing tools, visualization tools and even programming frameworks [15]. We avoid complex and complicated GUI dialogs for algorithms, provide useful default values for all parameters, direct and visual feedback for parameter changes and allow to combine complex data pipelines into containers, to hide their complexity. **Availability:** The hurdle of using software which is not available on your machine or operating system or which needs to be compiled tediously for your system is very high. Thus, OpenWalnut focuses on portability and availability. We focus the three major operating systems in use today (Linux, Windows and Mac OS) and provide pre-compiled and packaged versions of OpenWalnut for them. This makes OpenWalnut available for most of the researchers directly.

Results and Conclusion

In the last section we gave a short insight into the criteria that have driven and still drive the development of OpenWalnut. Although there are many tools available for handling and visualizing medical images and bio-signal data, each has its advantages and disadvantages. We do not try to re-invent the wheel, nor do we claim our project is superior to the others. We created a tool which complies to a certain set of criteria and tries to re-use as many as possible of existing tools and frameworks. We have targeted OpenWalnut at researchers in the medical and neuroscientific field and provide them with an open and powerful tool to explore their data or to create new ways of visualizing them. The software is available at <http://www.openwalnut.org>.

Bibliography

- [1] “Amira - visualize analyze present.” <http://www.amira.com/>.
- [2] “MeVisLab: development environment for medical image processing and visualization.” <http://www.mevislab.de/>.
- [3] “Medinria: Medical image navigation and research tool by inria.” <http://www-sop.inria.fr/asclepios/software/MedINRIA/>.
- [4] “SCIRun: A scientific computing problem solving environment, scientific computing and imaging institute (SCI).” <http://www.scirun.org>.
- [5] J. Ahrens, B. Geveci, and C. Law, “Paraview: An end-user tool for large data visualization,” in *Visualization Handbook* (C. Hansen and C. Johnson, eds.), Elsevier, 2005.
- [6] W. Benger, G. Ritter, and R. Heinzl, “The concepts of vish,” in *4th High-End Visualization Workshop, Obergurgl, Tyrol, Austria, June 18-21, 2007*, pp. 26–39, Berlin, Lehmanns Media-LOB.de, 2007.

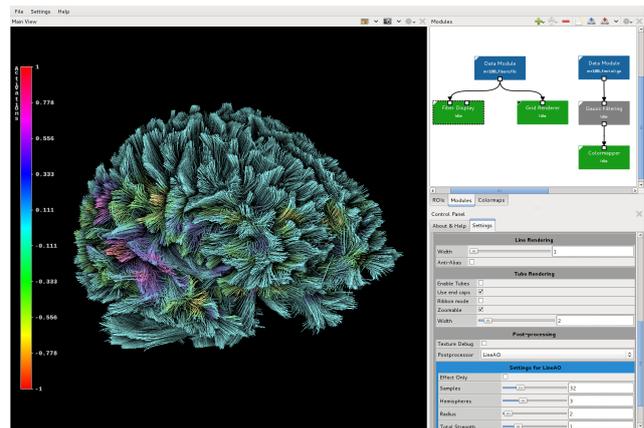


Figure 1: The GUI of OpenWalnut. On The right, you see the data-flow network and the properties (parameters) of the module “Fiber Display”. You can also see the tool-buttons in the title-bar of each dock. We strictly provide options and operations only where they can be applied, making the GUI more structured. The left part of the GUI contains the final rendering of fiber tractography data in combination with fMRI activations, improved with the LineAO algorithm [16]

- [7] “Eigen: a c++ template library for linear algebra: matrices, vectors, numerical solvers, and related algorithms.” <http://eigen.tuxfamily.org/>.
- [8] “OpenSceneGraph: an open source high performance 3d graphics toolkit.” <http://www.openscenegraph.org/>.
- [9] “FMRIB software library.” <http://www.fmrib.ox.ac.uk/fsl/>.
- [10] “Teem: Tools to process and visualize scientific data and images.” <http://teem.sourceforge.net/>.
- [11] “BioSig: an open source software library for biomedical signal processing.” <http://biosig.sourceforge.net/>.
- [12] “NifTI: Neuroimaging informatics technology initiative.” <http://nifti.nih.gov/>.
- [13] “numPy: Scientific computing tools for python.” <http://www.numpy.org/>.
- [14] “NiBabel: Access a cacophony of neuro-imaging file formats.” <http://nipy.sourceforge.net/nibabel/>.
- [15] “VTK visualization toolkit.” <http://www.vtk.org/>.
- [16] S. Eichelbaum, M. Hlawitschka, and G. Scheuermann, “LineAO — improved three-dimensional line rendering,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, pp. 433–445, March 2013.