

BRIAN (Brain Image Analysis) - A Toolkit for the Analysis of Multimodal Brain Datasets

Frithjof Kruggel and Gabriele Lohmann

Max-Planck-Institute of Cognitive Science
Inselstraße 22-26, 04103 Leipzig, Germany

The analysis of cognitive processes in man usually involves multiple examination modalities which map different aspects of the brain. Among these procedures, at least one modality yielding anatomical information (i.e. MRI^{*}) besides one or more functional modalities (fMRI, PET, SPECT, EEG, MEG) are involved.

Because these different examination methods yield complimentary information about the anatomical, metabolic and neurophysiological state of the brain, a combined data evaluation is highly desirable and will lead to results not achievable within one examination domain.

Such studies are of importance in research (cognitive neuroscience) and - with an emphasis on pathological processes - in clinical disciplines like neurology, neurosurgery and psychiatry.

We have developed a program package for the handling of image datasets (MRI, PET, SPECT, CCT) and signal datasets (EEG, MEG) which allows a combined analysis of these data sources in a four-dimensional coordinate space (x, y, z, and time).

1. DESIGN OUTLINE

We want to address the needs of two different groups of people working with our package. The first group consists of the medical and technical personnel involved with the generation and routine analysis of the datasets. The second group consists of the scientists who perform data exploration as well as researches who develop and integrate new image processing algorithms. While the first group expects a highly interactive and easy-to-use environment with well laid-out work paths, the second group requires an open environment which allows the rapid prototyping of new functions or image processing chains.

Thus, the design outline of our package consists of two layers. A highly interactive first layer features easy-to-use viewers for visualization, editors for inter-

* Abbreviations translate as follows: MRI = magnetic resonance imaging, fMRI = functional MRI, PET = positron emission tomography, SPECT = single photon emission computed tomography, CCT = cranial computed tomography, EEG = electro-encephalography, MEG = magneto-encephalography, ERP = event-related potential, CSG = constructive solid geometry

active measurements or marking and dialog boxes for the execution of pre-defined high-level processing chains. Physicians and trained personnel use this layer for day-to-day tasks like aligning a PET scan to a MRI dataset, performing a volumetric analysis based on a segmentation or evaluating an ERP experiment by combining MRI and EEG data. The second layer allows an experienced user to extend the functionality of this package by integrating new modules, defining image processing chains or by modifying parameter sets.

This two-layer structure is reflected in two components of the package (i) an interactive kernel for viewing and editing datasets and (ii) a visual editor for the design and execution of image processing chains. Both parts are described below in more detail.

2. VIEWERS AND EDITORS

The first layer of functionality is provided by a kernel containing the interactive display and editing capabilities of the package. Datasets in our problem domain fall into three categories:

- *Volumes*: 3D volume datasets from MRI and PET examinations to be displayed as images (slices or projections),
- *Signals*: 2D (position vs. time) datasets from EEG and MEG measurements usually displayed as curves, and
- *Geometrical Objects*: Surfaces defined in 3D which stem either from measurements with a tracking device or are computed from volumetric datasets. Also to this category belong space-filling geometric meshes for finite element analysis and CSG models of anatomical objects.

For these three data categories we provide separate viewers (for the display) and editors (for the modification of a dataset). Wherever appropriate, these viewers allow a *combined display* of datasets from two categories: a geometrical object (i.e. the surface of a tumor) is rendered inside a volumetric dataset of the brain; the electric potential recorded on the scalp is mapped as a color-coded texture onto the geometrically defined surface of the head. Furthermore, viewers may be *linked*: cursor movements or contrast adjustments in an orthogonal viewer are propagated to the other projections (or the 3D viewer); selection of a time point in a signal viewer maps the activity onto the head's surface display; selecting a position on this surface highlights the nearest electrode trace in the signal viewer.

Interactive editors provide functions which are (currently) no automatic procedures exist, for example: the definition of the stereotactical coordinate system in a volume dataset by marking reference structures; the collecting of segments which form an anatomical object („intelligent paintbrush“); the hand-segmentation of anatomical objects in a volume dataset.

To avoid a cluttered screen, a single display pane provides 4x3 windows for viewers. Windows may be enlarged or linked to exchange information. Care has been taken to allow near-real-time interaction on standard workstation without

specialized hardware. Drag-and-drop functionality is implemented for frequent operations like displaying the contents of a file by dropping it on a window, moving a window on the display pane, or saving the contents of a viewer into a file. The state of the kernel including all viewers may be saved into a file as a „workspace“. Figure 1 presents an example.

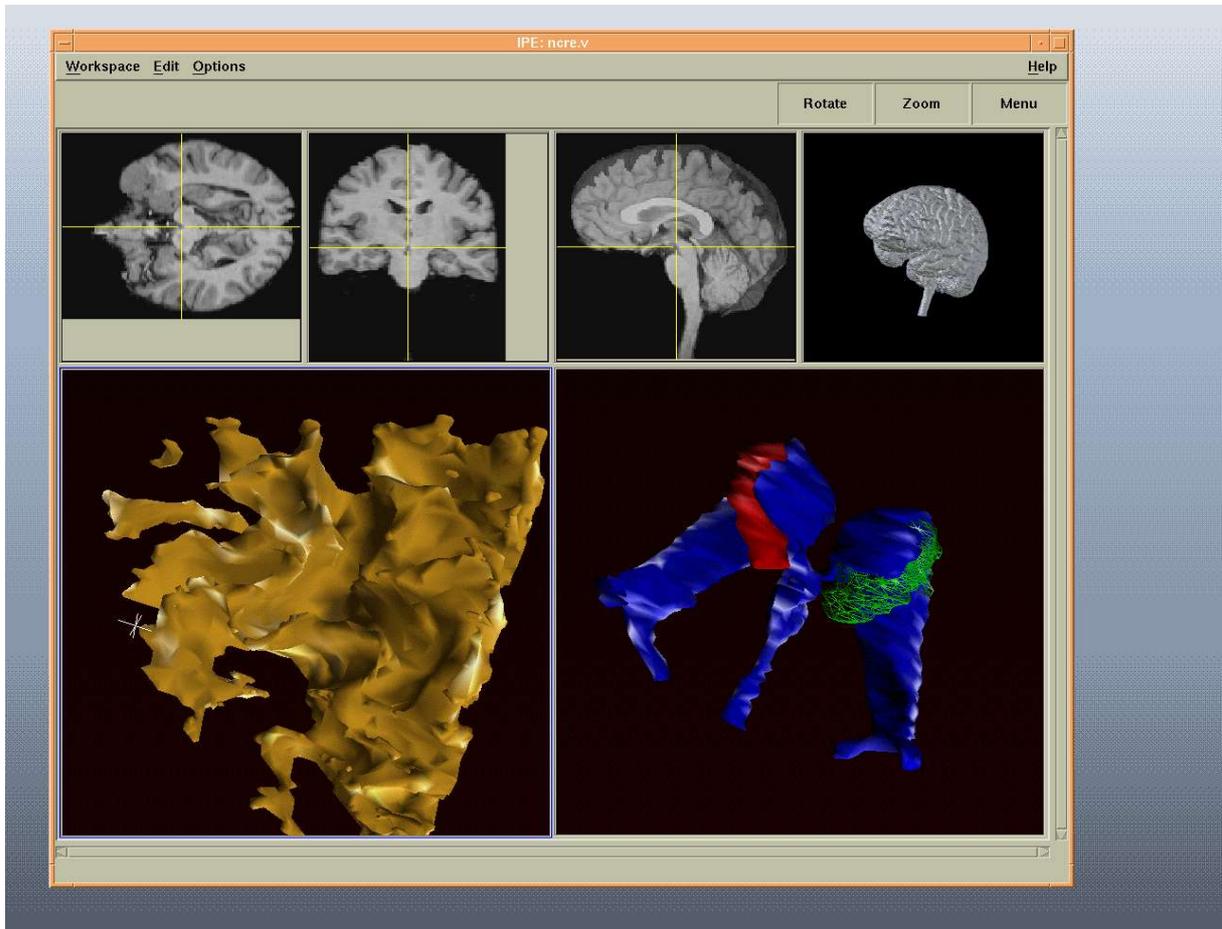


Figure 1. Example BRIAN workspace showing a geometrical viewer (top left), a volume renderer (top right) and three orthogonal viewers (bottom row). All viewers are connected via links, so that cursor movements are coordinated.

3. PROCESSING MODULES

The second layer of functionality is provided by external processing modules. Analysis algorithms under development, complex or time-intensive non-interactive applications (i.e. registration of datasets), or infrequent operations (i.e. data conversion) are placed into self-contained modules.

Currently, our package contains about 90 modules falling roughly into the following categories:

- access to a distributed network-wide database [1],
- import/export facilities to standard formats and devices,
- 3D operators for images (filters, segmentation procedures, morphology and arithmetic operators) [2],
- 2D operators for signals (filters, time-correlation, principal component analysis),
- operators for geometrical objects (extraction and concatenation of boundaries, surface fitting, fitting of deformable models) [3]
- correlation modules (intramodal (MRI-fMRI), crossmodal (MRI-PET, MRI-EEG, MRI-MEG)) [4],
- 3D and signal editors for the interactive manipulation of these datasets,
- a series of „standalone“ viewers (signal, 2D images, 3D surfaces, 3D volumes).

Much of our former work has gone into the development of these modules and has been cited here for reference. These modules extend the basic functionality of the interactive programs by adding the image and signal processing part.

A visual programming tool for the rapid development of processing chains is integrated within the viewing environment. Processing modules are represented visually as small buttons in a process editor. Applying the dataflow approach of well-known image processing packages like Khoros or AVS, the functional dependency between modules is represented by unidirectional links between these buttons. Clicking on a module button opens a dialog window in which parameters of a module may be supplied. During the execution of such a chain, the processing status of a module is indicated by changes of the button's color. Input to and output from a processing chain are linked either to external files or run directly into windows of the viewing environment.

The native modules allow pipe- and file-based connections. In the context of the visual editor we favor file-based data connections, because a processing chain is easier to restart and intermediate results are available. Given the processing time of single modules ranging from seconds to hours the additional time for file i/o is considered insignificant.

This environment was conceived as our design tool for new algorithms as well as a test bed for the development and stability of processing chains. Figure 2 shows such a processing chain involved in the removal of non-brain parts in a MR tomogram of the head. Once the functionality of a specific chain has settled it is repackaged into a single module.

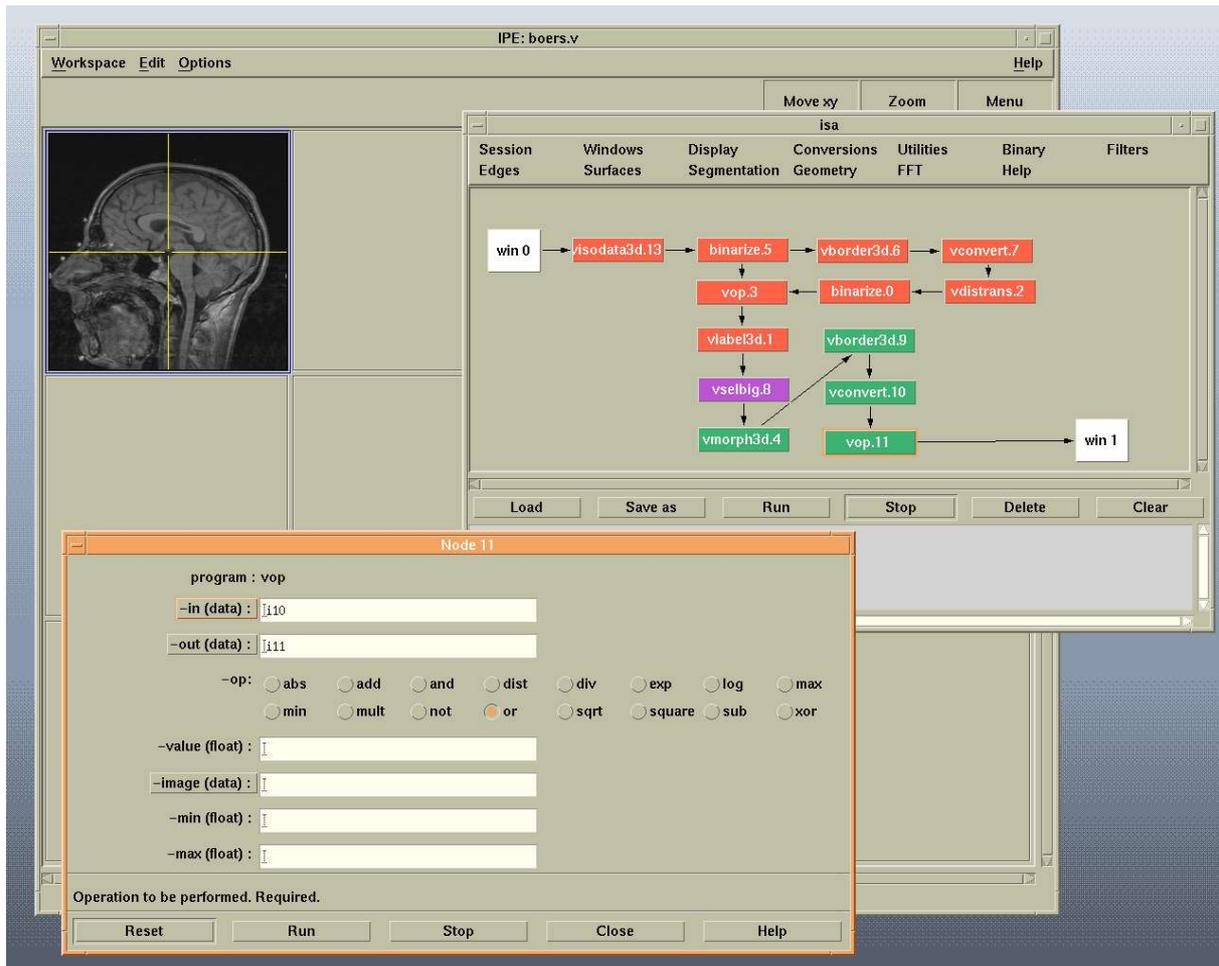


Figure 2. Example view of a processing chain in the visual editor. Individual modules are represented as small buttons whose status is color-coded. The result of a chain is either displayed in a stand-alone viewer or fed back into a workspace window.

4. IMPLEMENTATION FRAMEWORK

Our BRIAN package has been implemented in the C/C++ programming language using X-Windows 11R6 and Motif 1.2.4 as a graphical user interface. For the display of object surfaces, OpenGL (or its public domain clone Mesa) is employed. Some of the time-consuming modules have been parallelized using MPI [5] on a workstation net. File i/o is based on the Vista toolkit [6] which is machine-independent and allows a very flexible and extensible handling of various

data representations. The package has been ported to standard Unix workstations from different vendors and works in a heterogeneous environment.

5. FUTURE ENHANCEMENTS

The BRIAN project was conceived two years ago and is under constant development. While its basic structure has stabilized, we are now focusing on the development of modules to support recurrent procedures:

- An *anatomical database* including models of brain structures will be incorporated into BRIAN. It will support segmentation algorithms and aid in the (automatic) identification of brain structures as well as the location of brain activation centers from fMRI and PET measurements.
- Mechanical and electro-magnetical *finite element models* of the head will be developed to monitor changes in anatomical structures during disease processes and to identify sources of electro-magnetical brain activity.
- The *visualization components* of this package will undergo constant improvement to satisfy the growing needs of explorative data analysis in our institute.

Furthermore, a few of the computationally intensive applications will be ported to a dedicated parallel computer. The use of this machine is transparent within this environment.

By scientific visualization we want to promote a deeper level of understanding of the datasets under investigation, relying on the human's powerful ability to visualize. In our case, we want to gain new insight into the underlying cognitive processes reflected in the functional datasets. The BRIAN package tries to make a contribution to this field.

REFERENCES

1. F. Kruggel, A. Horsch, D.Y. von Cramon, A Collaborative Image Database out of the box, Proceedings IMAC 1995 (Hawaii), IEEE Press, Washington.
2. G. Mittelhäußer, F. Kruggel, Fast Segmentation of Brian magnetic resonance Tomograms, Proceedings CVRMed 1995 (Nice), N. Ayache (ed.), Springer, Heidelberg.
3. F. Kruggel, G. Lohmann, Oberflächen aus Konturen, 16. DAGM Symposium, Mustererkennung 1994 (Wien), W. Kropatsch (ed.), Springer, Heidelberg.
4. F. Kruggel, A. Bartenstein, Information Processing in Medical Imaging 1995 (Brest), Y. Bizais, C. Barillot, R. Di Paola (eds.), Kluwer, Dordrecht.
5. W. Gropp, E. Lusk, A. Skjellum, Using MPI, MIT Press, Cambridge, 1994.
6. Pope AR, Lowe DG (1994) Vista: A Software Environment for Computer Vision Research, Proceedings CVPR 94 (Seattle), IEEE Press, Washington.