VisiQuest : A Problem Solving Environment for Scientific Computing and Visualization

AccuSoft
www.AccuSoft.com

Introduction

VisiQuest is an integrated problem solving environment which allows researchers and scientists to perform a variety of tasks related to scientific computing, including medical image and signal processing, data manipulation, and scientific visualization. VisiQuest contains over 300 operators for information processing, data exploration, image and signal processing, and data visualization. Generalized to form a broad base-level technology, these operators facilitate problem solving in a wide variety of application domains used in research, science, government and industry. All VisiQuest operators may be run independently from the command line or executed from within the visual programming environment. With the visual programming environment, visual programmers can combine the scientific operators of VisiQuest in a graphically expressed, data flow visual language for experimentation, data exploration, and to rapidly prototype new solutions.

Designed to act as an integrated software development environment, VisiQuest allows you to write new data processing operators and create new visualization applications. The visual programming environment will automatically include any new user created operators, making it effective as a software integration environment. Complex algorithms can be prototyped in the visual programming environment and then deployed to a stand-alone application for use by an image analyst.

Scientific Problem-Solving Operators

For problem solving by researchers and scientists, VisiQuest provides a wide variety of programs, or operators, for information processing, data exploration, and data visualization. VisiQuest programs are typically generalized to form a broad base-level technology rather than any specific scientific area. In addition to medical imaging, some of the other application domains in which VisiQuest has been used include: high performance computing, modeling & simulation, applied research, image understanding & analysis, remote sensing, and automatic target recognition.

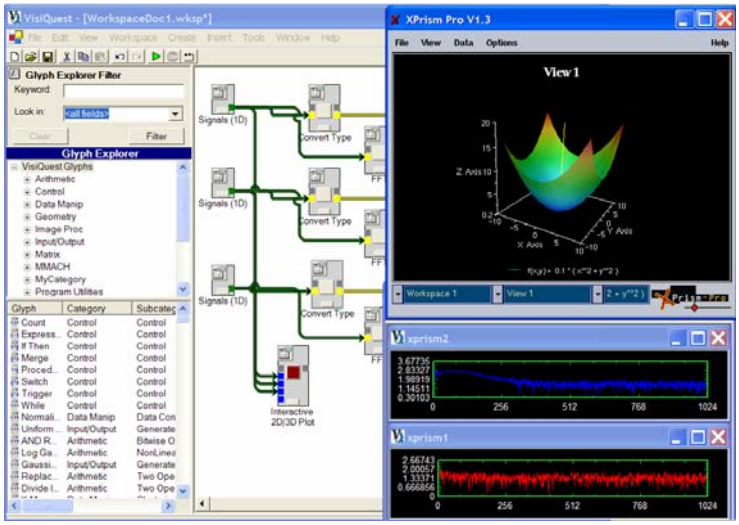

Figure 1
VisiQuest is readily used for the processing and exploration of multi-dimensional data. Here, several NMR brain scans are compared.

All VisiQuest programs provide automatic data format conversion, data type conversion, file format conversion, support of large data sets, and support for multidimensional data. VisiQuest programs offering end-user solutions to scientific problems fall into a number of categories:

- Multidimensional (1D to ND) data manipulation operators, including data import, data export, data generation, single & double operand arithmetic, trigonometric and nonlinear functions, bitwise and complex operators, comparison operators, linear transforms, histograms, size & region operators, threshold & clip operators, noise generation/introduction, clustering operators, FFTs and convolutions.

- Image processing of multiband images, including convolution, data compression, feature extraction, frequency filters, geometric operations, nonlinear filters, and spatial filters.

- Matrix operations, including extraction of rows, columns, and diagonals, computation of eigenvectors and eigenvalues, computation of the inverse matrix, computation of the least-squares solution to a set of linear equations, LU decomposition, computation of the matrix product, and singular value decomposition.

- Data visualization applications, including an image sequencing/animation program, an image display & manipulation package, an ROI extraction program, an image capture program, a 2D/3D plotting package, a non-interactive data display tool, a colormap alteration tool, and an interactive image/signal classification application.

- 3D visualization and software rendering, which consists of a 3D geometry library, a number of data processing routines for 3D visualization, and a software rendering application.

The primary objective in developing these operators was to facilitate domain-specific work while simultaneously enabling cross-domain collaboration. When designing the data processing and visualization operators, several issues critical to increasing the productivity of the scientist in solving data processing and analysis problems were addressed. These issues include providing domain interoperability in order to promote the reuse of software solutions over diverse domains; providing format and system independence, including the capability to process very large data sets; providing the ability not only to visualize data using traditional methods, but to allow for data exploration; and instilling confidence in these tools by ensuring reliability and stability. This approach provides flexibility to the end user and promotes collaboration within and across application domains.

Visual Programming Environment

Visual programming environments provide iconic elements that can be interactively manipulated according to some specific spatial grammar for program construction. Data flow is a naturally visible approach in which an algorithm is described as a directed graph where each node represents an operator or function and each directed arc represents a path over which data flows. By connecting the data paths between nodes, users can interactively draw out a solution in an intuitive manner that matches their mental representation of the problem. Thus, a visual environment for problem solving introduces a level of abstraction which provides accessibility to the functionality represented by the underlying operators, regardless of the developer's programming experience; moreover, it increases the productivity of both experienced and novice developers.

The extensive image processing, data manipulation, matrix processing, and scientific visualization operators included in the VisiQuest system can all be interconnected within the visual programming environment. Visual hierarchy, iteration, flow control, and expression-based parameters extend the data flow paradigm to provide a powerful prototyping, experimentation, simulation, and problem solving system.

Programs from the VisiQuest system are represented as visual objects called glyphs. The program which corresponds to the glyph is called an operator. To create a visual program, the user selects the desired operator, places the corresponding glyphs on the workspace, and connects the glyphs to indicate the flow of data from operator to operator, forming a network within a workspace.

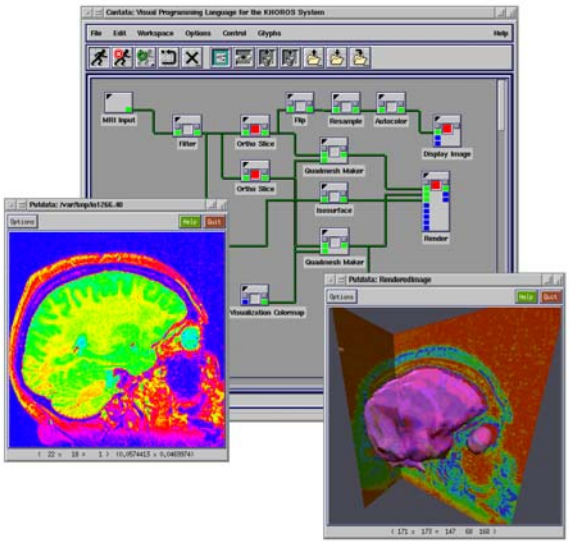

Figure 2
The visual programming environment can be used to prototype complex analysis and visualization algorithms. Here, a workspace that identifies features of interest in a cranial NMR scan and constructs a 3D rendering of the brain surface.

Control structures can be used to branch and merge data flow, or to implement loops. Workspaces can be executed, saved and restored to be used again or modified later. Workspaces may also be encapsulated into independent applications with a very simplified graphical user interface so that they may be treated as stand-alone VisiQuest applications.

Visual hierarchy, iteration, flow control, and expression-based parameters extend the data flow paradigm to allow the visual programming environment to be used as a simulation and prototyping system. Data and control-dependent program flow is provided by control structure glyphs such as if/else, while, count, and trigger. Visual subroutines, or procedures, are available to support the development of hierarchical data flow graphs. Variables may be set interactively by the user, or calculated at run time via mathematical expressions tied to data values or control variables within the visual network.

By combining a natural environment of visual constructs with the programming features typically found in textual languages, the visual programming environment provides a powerful problem solving and prototyping system. Visual hierarchy, iteration, flow control, and expression based parameters augment the traditional data flow paradigm so that the visual programming environment can be used effectively in a number of domains, including process control, simulation, and system integration. When combined with the data processing and data visualization programs available in the VisiQuest system, the visual programming environment is particularly well suited for scientific data processing and visualization.

Software Development Environment

The VisiQuest software development environment consists of are three interactive tools which operate on toolboxes and software objects: Craftsman, the toolbox creation and management tool, Composer, the software object editor, and GUISE, the direct manipulation GUI design tool.

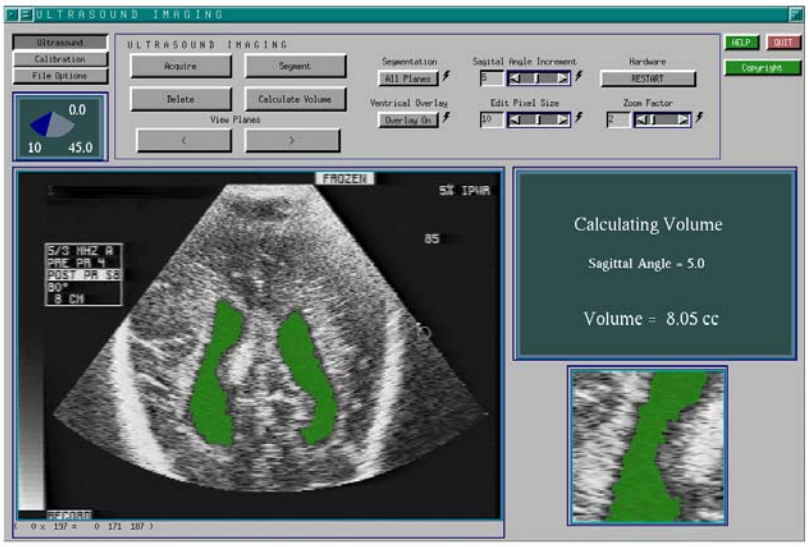

Figure 3
VisiQuest can be used to create deployable applications for image analysts. Here, a custom application for identifying ventricular areas within cranial ultrasounds was created for use by a medical technician.

To customize and extend the scientific capabilities to the specific needs of a particular application domain, these software development tools allow VisiQuest developers to build and integrate new operators that may then be accessed from the visual programming environment. Developers may also create interactive scientific visualization applications. The VisiQuest software development system is often used for software integration, where existing programs can be brought together into a consistent, standardized, and cohesive environment.