

INTERACTIVE DATA EXPLORATION AND TELECOLLABORATION IN BIOMEDICINE USING AVS

Gudrun J. Klinker
Cambridge Research Lab, Digital Equipment Corporation
One Kendall Square, Cambridge, MA 02139

April 7, 1993

Abstract

While AVS is a very successful visualization system, it is not currently suited for data exploration and telecollaboration applications because it has not provided the necessary interactive data probing, cursor linking and window migration tools. We have extended AVS in this direction. Interactive, telecollaborative data exploration is a valuable tool for many applications. In this paper, we show how it can be used for the analysis of multi-spectral biomedical data to quantify tissue vascularity in the thyroid of the rat.

INTRODUCTION

In the past, AVS has not been suited for data exploration and telecollaboration applications because it has not provided the necessary interactive data probing, cursor linking, and window migration tools. Its *display image* module encapsulates all handling of user interaction, and no feedback is provided to other modules. The module also does not provide means to overlay geometric data (such as a label or a mark) on images or to show the data anywhere else than on the screen where AVS has been started up. We have extended AVS with the following concepts to provide such capabilities [2]:

- **Feedback Channel:** Our system provides a new X-based *display data* module which makes user interaction available to other modules.
- **Log Record:** Modules which use the feedback channel need to invert all geometric transformations to relate the selected window position to the correct pixel position in the original data set. All modules in our system send a *log* record along with the array data in which they record all coordinate transformations, as well as other descriptive information about the data.
- **Window Migration:** Our *display data* module is able to move windows between arbitrary, user-specified displays. The module adapts its display mechanism to the capabilities of the selected frame buffer.
- **Mixed Displays:** The *display data* routine accepts both image data and geometric data, overlaying the geometric data on the image. Geometric coordinates are specified in floating point format so that geometric objects can be positioned to sub-pixel precision.

As a result, users can now probe the image data. They can also attach data manipulation routines or special data presentation routines to the *display data* module and apply the routines to an interactively

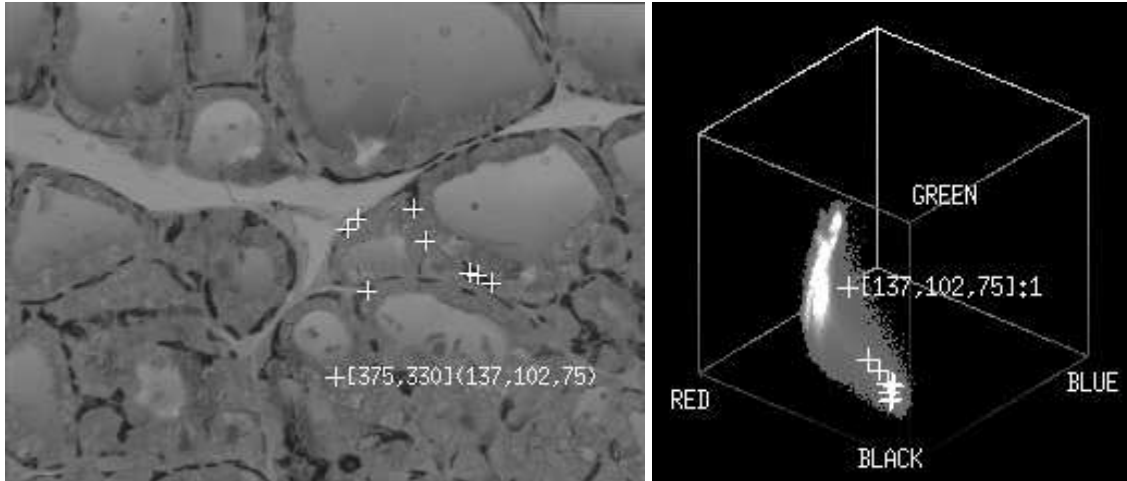


Figure 1: Thyroid image and color histogram with overlaid cursor positions.

specified portion of the image data. Because our module overlays geometric data on the window, cursors, graphs, icons etc. can be shown with the image data. Furthermore, our extensions provide the means to establish linked image cursors between several windows, providing joint exploration of data from several modalities. Our window migration and cursor linking capabilities together are the key to telemedicine applications. Because of this interactive flexibility, our system is amenable to many forms of data exploration in a telecollaborative environment.

In what follows, we focus on one application to demonstrate these features. Many other applications have also been implemented with similar success.

MULTI-SPECTRAL IMAGE DATA

Multi-spectral data is increasingly used in biomedical applications. The additional spectral information can help researchers distinguish between different types of tissue within microscopic images of sectioned and stained samples. For example, quantitative analysis of such images can provide valuable information to biomedical researchers interested in assessing structural changes within organs, either as a result of normal growth and development, or indicative of various pathologies.

We illustrate interactive tools for the analysis of spectral data at the example of a color image of a serial section from the thyroid of the rat (Figure 1). For the thyroid image, biomedical researchers are interested in developing methods to automatically quantify tissue vascularity. They have stained the tissue so that blood capillaries appear bluish. The task is to design an automatic color segmentation procedure which finds all blood capillaries so that the proportional occurrence of capillaries in the tissue can be determined. Such segmentation is not easy to achieve because the blood vessels do not form one contiguous area in the image, their boundaries with neighboring tissue are sometimes imprecise, and the stained capillaries do not exhibit a color shift large enough to isolate the colors into a separate cluster. The design of an automatic color segmentation procedure therefore is not straightforward. Our system provides the tools for researchers to design a suitable algorithm while familiarizing themselves with the specific color distribution properties of their images.

To study the color distribution in the image, we histogram all pixels in a three-dimensional color histogram. Every pixel (r, g, b) in the image constitutes an index into the histogram. The right half of Figure 1 shows the color histogram of the thyroid image. The brightness of each slot in the histogram indicates how often this particular color occurs in the image.

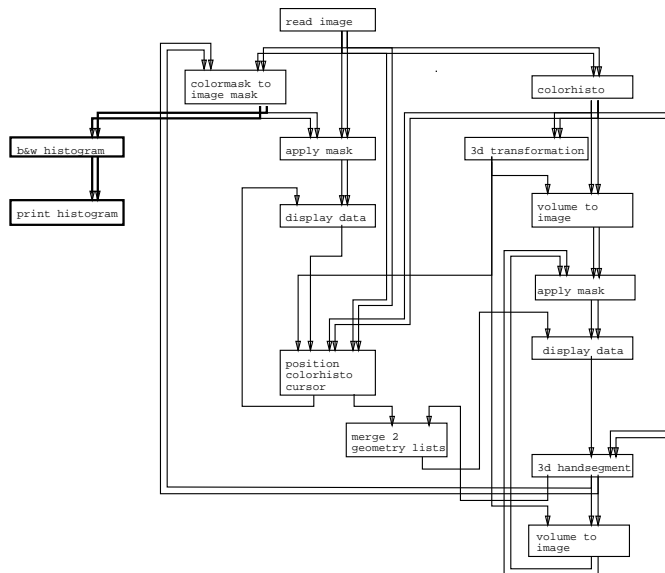


Figure 2: AVS network with cross-linked cursors between the image and the histogram.

Figure 2 shows the AVS network to create the histogram, display the data, cross-link the cursors and statistically analyze segmentation results. The components of the network will be described throughout this paper. Module *colorhisto* takes a color image as input and generates the color histogram. Module *volume to image* is a volume renderer. Using the viewing direction specified in the *3D transformation* module, it projects the color cube into an image plane by selecting either the closest pixel or the pixel with the maximal histogram count along each projection ray. The histogram in Figure 1 displays the pixels with maximal count values, thus exposing the internal structure of the histogram rather than its external shape. The thyroid image and the projected histogram data are sent to two *display data* modules which present the data in two windows on the screen.

CURSOR LINKING FROM THE IMAGE TO THE HISTOGRAM

For the analysis of color variation in color images, it is essential that researchers be able to visualize the relationship between pixels in the image and their position in the color histogram. With our extensions to AVS, users can establish a connection between the image data and the histogram data so that a selected cursor position in the image is highlighted in the histogram.

Overlaid on the thyroid image is a crosshair with text indicating the index and the red, green and blue color values of a selected pixel. The corresponding color slot in the color histogram is also marked with a crosshair and text describing the color index and the histogram count. We show several other crosshairs on the thyroid image and their corresponding marks on the histogram to demonstrate how researchers can use this technique to visualize color similarities between several selected pixels.

In the network in Figure 2, module *position colorhisto cursor* provides this cursor-linking capability. It accepts a selected window position as part of the *log* record from the *display data* module. The module transforms the window position into the correct pixel index and then retrieves the color pixel from the original image data. The color triple is then used as an index into the color histogram to retrieve the histogram count for this color. The information is packaged into two drawing commands which are sent back to the two *display data* modules and overlaid as crosshairs and descriptive text on the respective windows.

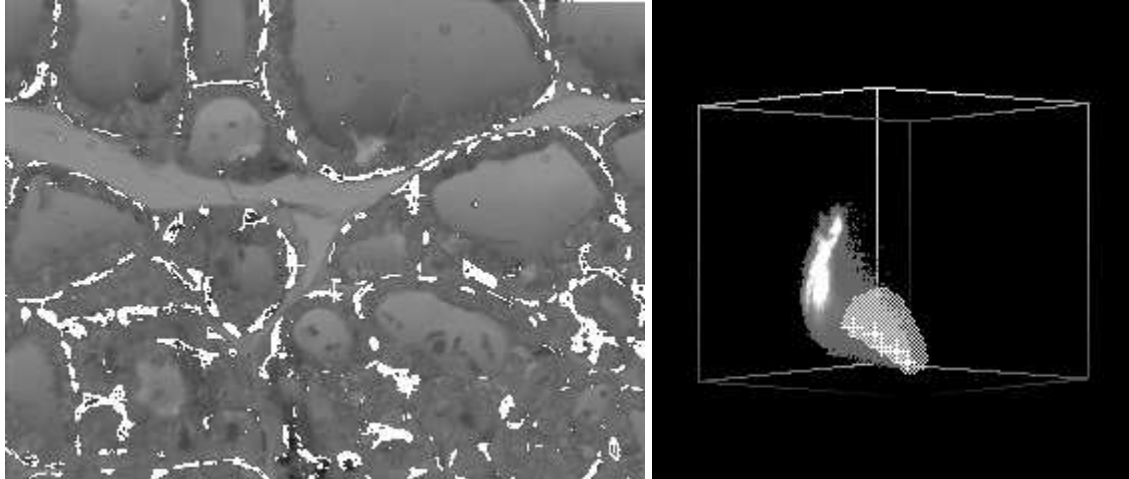


Figure 3: Thyroid image and color histogram with overlaid selected mask.

CURSOR LINKING FROM THE HISTOGRAM TO THE IMAGE

Users can also select interesting areas of the histogram and see the corresponding image pixels, as shown in Figure 3. Overlaid on the color histogram is a mask which was created interactively. The user has outlined an area of the histogram. Module *3D handsegment* generates a three-dimensional color mask which covers all color slots behind the outlined area. The three-dimensional color mask is projected and overlaid on the histogram window using modules *volume to image* and *apply mask*.

The color mask is also mapped into a two-dimensional image mask for the thyroid image: modules *color mask to image mask* and *apply mask* test all pixels of the thyroid image and mark them if they correspond to a marked slot of the color mask. The resulting image is shown in Figure 3. All selected bluish pixels are shown in white.

THREE-DIMENSIONAL SEGMENTATION IN THE COLOR HISTOGRAM

The *3D handsegment* module is a sophisticated tool to select voxels from a three-dimensional volume, such as the color histogram. Users can outline areas of the histogram to be included into the color mask. The left picture in Figure 4 shows a side view of the color mask in Figure 3. Users can incrementally segment many disjoint parts of the histogram and include them under changing viewing directions into the color mask. They can also remove parts of a selected mask by outlining an area, as shown in the right picture of Figure 3.

Several separate masks can be specified together by using different mask labels with different colors. When a mask with several labels is sent to the thyroid image, the corresponding pixels are highlighted in the respective colors. In Figure 5, we have selected a second segment at the tip of the histogram in black. The corresponding thyroid image shows pixels with colors under the black mask in black and colors under the previously generated white mask in white. Users can thus simultaneously visualize the occurrence of several interesting segments of the histogram.

QUANTITATIVE DATA ANALYSIS

The purpose for segmenting the bluish pixels in the thyroid image is to provide a mechanism to quantify tissue vascularity in the thyroid. After the bluish pixels have been determined, we can now count them.

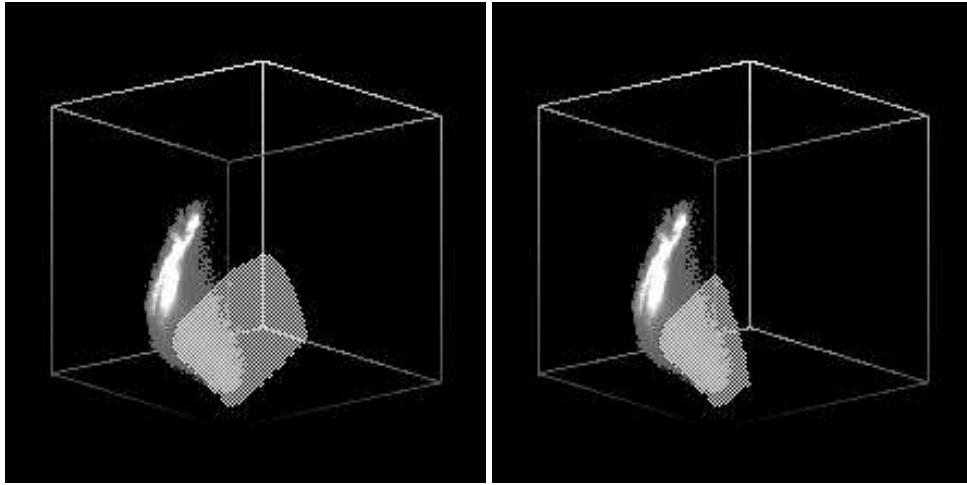


Figure 4: Three-dimensional histogram segmentation. The left picture shows a side view of a selected mask. In the right picture, a part of the mask has been removed.

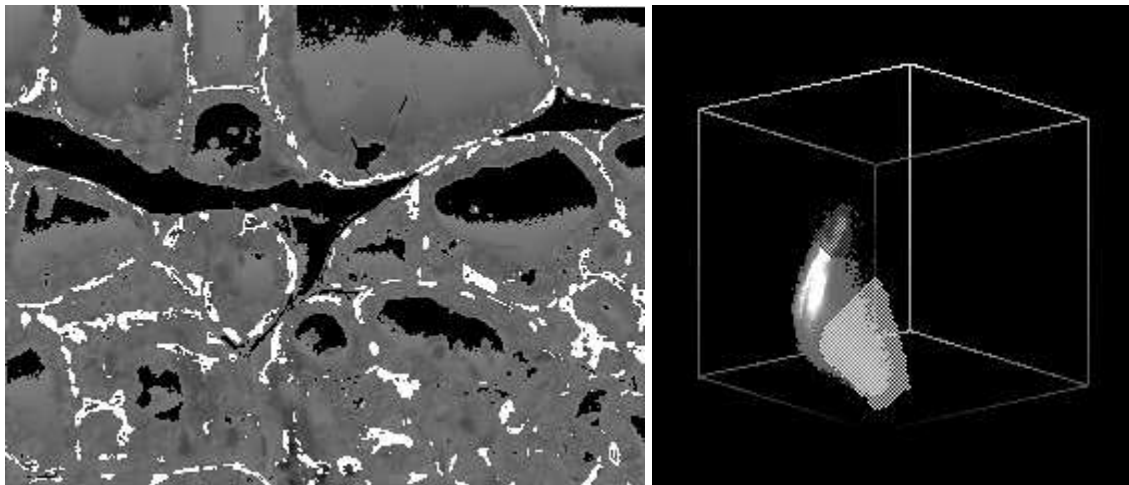


Figure 5: Color histogram with two separate masks.

```

vector bwhisto(thyro24.tga)
=====
256 values
0:      286914   93.4
1:      20286    6.6
total:   307200
=====

```

Figure 6: Histogram of the masked (1) and unmasked (0) pixels of Figure 3.

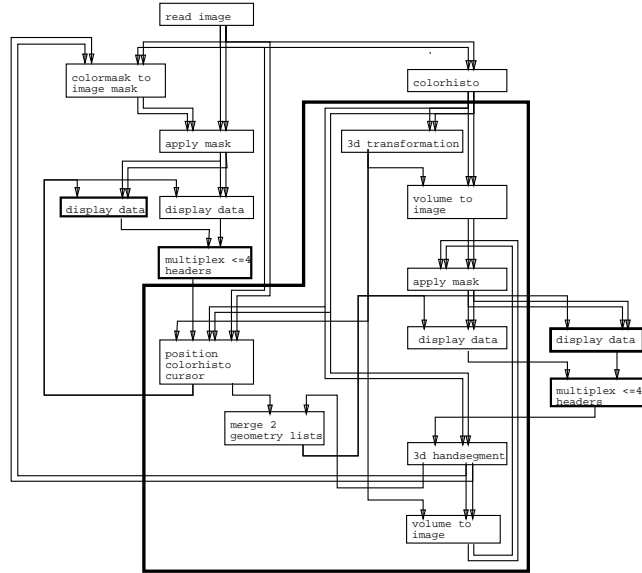


Figure 7: AVS network to explore the thyroid data.

Module *bw histogram* in the AVS network in Figure 2 generates a histogram, indicating how often each color label occurs in the image mask. Module *print histogram* prints out the results. Figure 6 shows the results for the white mask: 6.6% of the image are covered with bluish pixels (mask value 1). The remaining 93.4% of the pixels are unmasked. Other more sophisticated statistical data evaluation packages can replace the histogramming modules as necessary.

TELEMEDICINE

Telecollaboration is integrated in our data exploration environment. All capabilities available to one user are also available to two or more users working on the same problem. This approach stands in contrast to current practice where users have to switch back and forth between exploring data in one system and sharing the results in a different system.

We can provide integrated telecollaborative capabilities because we combine cursor linking with a window migration capability. Users can send any window to any screen simply by specifying a *display name* parameter. We have successfully demonstrated telecollaborative arrangements between Chicago and Boston as part of the Innovation Showcase at Siggraph 92, as well as between Boston and Sweden.

Figure 7 shows how the AVS network of Figure 2 can be extended to that end. Both *display data* modules have been duplicated so that the thyroid image and its color histogram are now each shown twice. Two

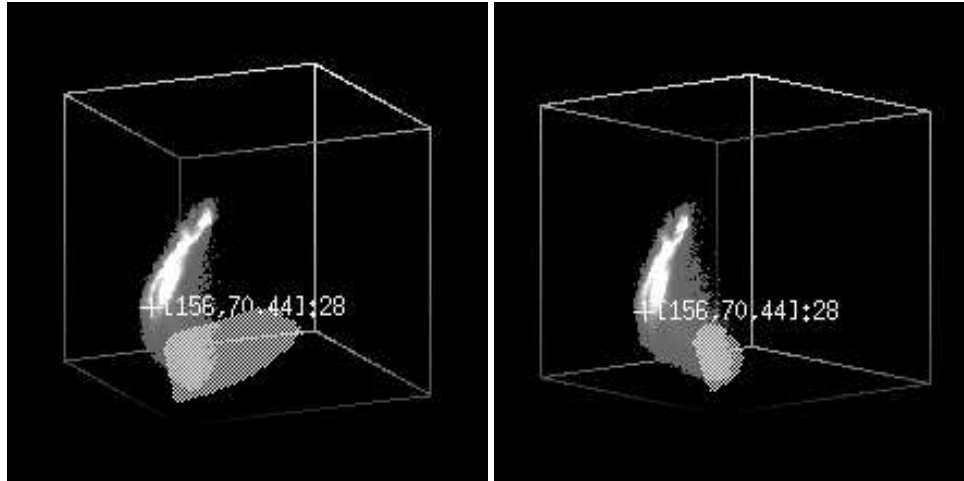


Figure 8: Different views of the color histogram during a telecollaborative data exploration session.

modules *multiplex up to 4 headers* receive the *log* output channels with the cursor positions from the respective module pairs. As a result, cursor interaction in either of the two thyroid windows repositions the crosshair in both windows, as well as in both histograms. Similarly, the histograms can be segmented in either of the two histogram windows.

Figure 8 demonstrates that telecollaboration partners can have different views of the data while they discuss it. The histograms are presented from different viewing directions. Yet, the selected color pixel and color mask are showing the same three-dimensional positions. Independent three-dimensional viewport capabilities can be achieved by duplicating the part of network in Figure 7 that is outlined by a bold polygon. AVS provides a mechanism to collapse this part of the network into a macro-module which can then be duplicated easily. Individual windows can also be enlarged or reduced in size. They can be rendered on framebuffers with different depths and shown in different colors. Figure 8 emphasizes that our system combines all of its data presentation and exploration capabilities with a very flexible data communication scheme. Whereas other telecommunication systems typically provide an identical arrangement of shared windows, users of our system are free to assemble any window arrangement they need and share all or parts of it with as many colleagues as necessary.

We expect these telecollaborative capabilities to be an important extension to data exploration. Many data exploration tasks in real applications are group efforts rather than problems solved by an individual. There will be an increasing demand for systems that allow researchers to share their views and truly explore them with their colleagues. The current situation where researchers explore data in one system and share it in another one is an obstacle to remote collaboration. Our system can enhance existing teleconferencing technology in this respect. Existing systems provide sophisticated techniques to send compressed image data and sound for real-time teleconferencing. Tools also exist to share and edit arbitrary X-based windows on several displays [1]. Such tools are well suited for sharing text windows but they lack the means to provide in-depth exploration capabilities for large data sets. We expect future telecollaboration systems to combine traditional window sharing and teleconferencing arrangements with data exploration systems to analyze large data sets.

SUMMARY

Telecollaborative data exploration imposes special requirements on visualization environments. It requires a high degree of interactivity which has not been available in AVS. Our extensions to AVS provide such interactivity.

In this paper, we have demonstrated how interactive data analysis is used for determining the tissue vascularity of the thyroid of the rat. In a reconfigurable AVS network, we have linked the thyroid image to its three-dimensional color histogram. Cursor interaction with the thyroid image causes a corresponding cursor in the histogram to mark the color of the selected pixel. Conversely, when researchers select a group of colors in the histogram the corresponding pixels in the original image are marked. We have shown how this technique can be used to interactively segment the capillaries in the image by selecting bluish colors from the histogram. Tissue vascularity can then be quantified by counting the segmented pixels. The setup can be extended for telemedicine by duplicating the modules which display the data and establishing the appropriate linked cursors between all windows.

Interactive data exploration and telecollaboration are essential requirements for many applications. Specialized systems performing a particular interactive task, such as scatter plot analysis, have been available for a while. Our extensions provide general-purpose tools duplicating such functionality while also being easily modifiable to serve other tasks. We therefore expect our extensions to AVS to be useful in many old and new approaches towards exploring large data sets.

ACKNOWLEDGMENTS

I would like to thank Michael Doyle for the thyroid data and Ingrid Carlbom and the Visualization Group at CRL for many discussions and suggestions.

References

- [1] Michael Altenhofen, Burkhard Neidecker-Lutz, and Paul Tallett. Upgrading a window system for tutoring functions. In *European X Window System Conference and Exhibition (EX'90)*, November 1990.
- [2] G.J. Klinker. An environment for telecollaborative data exploration. Submitted to *Visualization 93*, Oct 1993.