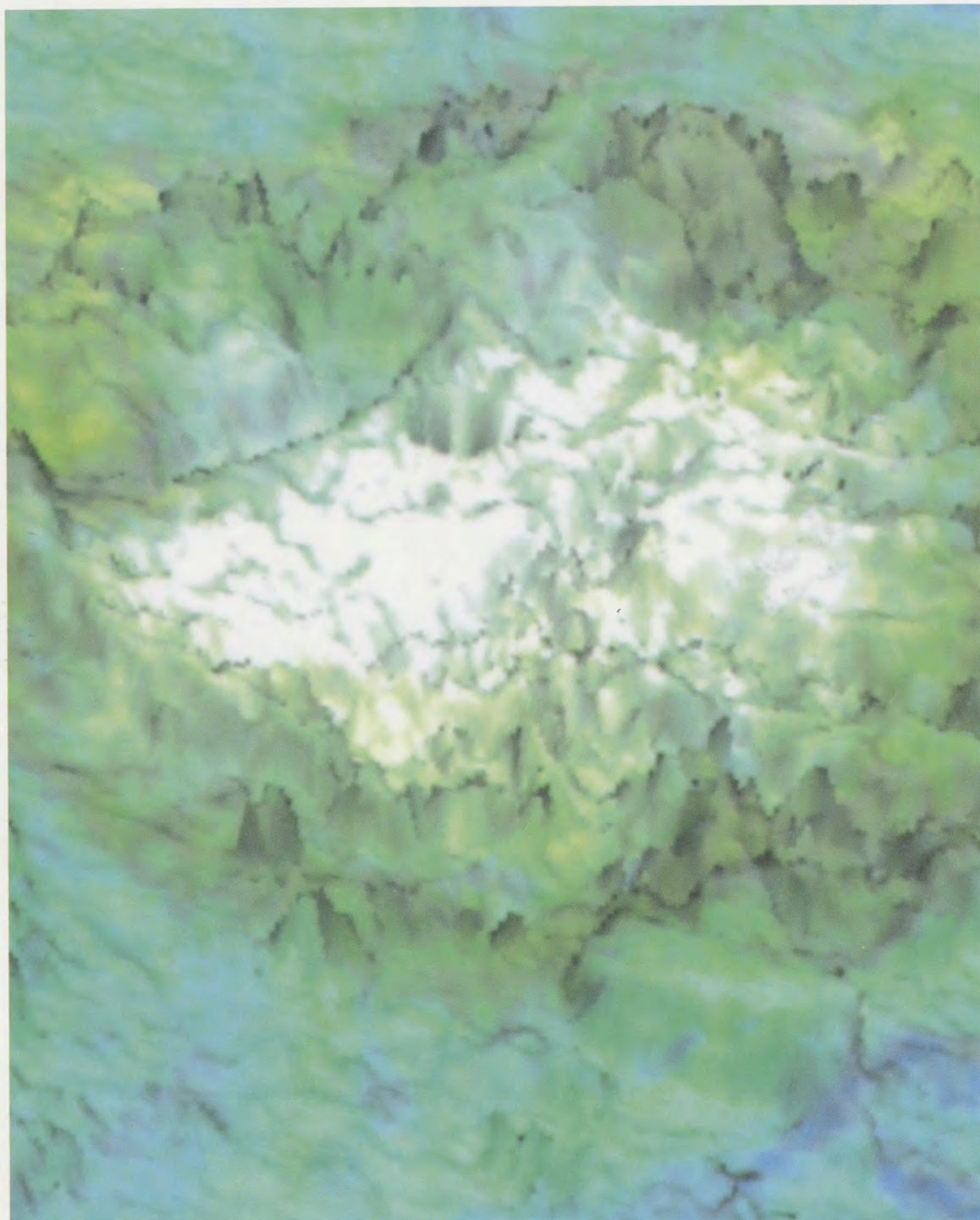


# IN PHYSICS

Reprinted from  
**MAY/JUN 1994**

# COMPUTERS



AMERICAN  
INSTITUTE  
OF PHYSICS

**MOSAIC**  
**Computer Algebra**

# A high performance Interactive Image SpreadSheet (IISS)

A. F. Hasler, K. Palaniappan, and M. Manyin

*Laboratory for Atmospheres, NASA/Goddard Space Flight Center, Greenbelt, Maryland 20771*

J. Dodge

*NASA Headquarters Washington, DC 20546*

*(Received 30 July 1992; accepted 12 April 1993)*

---

New methods are needed for visualizing, interpreting, comparing, organizing, and analyzing immense multispectral satellite datasets. The traditional numerical spreadsheet paradigm has been extended to develop a new scientific visualization approach for processing multisensor image datasets interactively. Exploring the advantages of extending the powerful spreadsheet style of computation to multiple sets of images and organizing image processing tasks is the objective of the Interactive Image SpreadSheet (IISS) project at Goddard Space Flight Center. In the IISS each cell can display any portion of an original or calculated image, a projection of a multidimensional dataset such as a 3-D surface, a glyph (graphic symbol) representing an image, digitized maps, digital terrain models, graphs, or vector drawings. The term *image* is used in a general sense to refer to any 2-D multisource dataset. The IISS typically contains an array of image cells of arbitrary size each of which can contain one or more frames (images). The user can scroll or page through this multidimensional cube of frames along any dimension. The IISS emphasizes an immediate visual approach to interacting with data. A unique capability that the IISS provides are the highly interactive browsing tools, accessible through a graphical user interface, for effectively inspecting large sets of image arrays using synchronized cell level operations such as zoom, roam, animation, and function execution. The IISS combines the quantitative aspects of a numerical spreadsheet with powerful visualization tools to enable an investigator to easily experiment with various combinations of multispectral image data using a library of standard algorithms and to interactively develop custom algorithms. Remotely sensed datasets from multispectral instruments on satellites such as GOES, NOAA, Nimbus, DMSP, and Landsat have been used to develop and evaluate the functionality of the IISS. Formula expressions for creating color composites, implementing image enhancements, calculating vegetation indices, viewing perspective and stereo imagery have been developed using multispectral data. The IISS can also be used in a variety of imaging disciplines that routinely need to organize and manipulate large volumes of visual data including numerical simulation data, observational field data, astronomical imaging, biomedical imaging, computer vision and manufacturing robotics, business document imaging, and multimedia. The practical realization of the computationally challenging IISS project relies on the fact that personal superworkstations have become inexpensive enough that one can extend the interactive scalar spreadsheet concept to the image processing field. The hardware features that make this possible include multiple processors, large amounts of general purpose memory, high-performance data buses, large mass storage, and pipelined or other advanced architectures for graphics and image operations. The need for increasingly more interactive imaging and visualization applications using high definition displays in collaborative environments will continue to drive the demand for more powerful hardware features and information network capabilities that are widely accessible.

---

## INTRODUCTION

The motivation for exploring new methodologies for rapidly browsing, verifying, organizing, analyzing, and visualizing large volumes of remote sensing datasets is the current focus on global change studies. The need to work with datasets from new instruments that are at least two orders of magnitude larger than those from current instruments demands the reevaluation of traditional approaches to image manipulation and analysis. Researchers are now using entire image archives of historic data, such as the 120 ter-

abyte Geostationary Operational Environmental Satellite (GOES) dataset,<sup>1</sup> to study long term environmental phenomena or human impact on the biosphere. The ability of researchers to browse such immense datasets from multiple databases in order to extract, compare, and correlate the observations will depend upon the development of innovative tools.

The next generation of remote sensing observing systems like those in the Earth Observing System (EOS) series will routinely provide multispectral data from a very large number of sensors. For example, the MODerate resolution



Imaging Spectroradiometer (MODIS) will be the keystone instrument on the first EOS platform for satellite-based global observations of atmosphere, land, and ocean processes. The design specifications for the MODIS instrument have been based on multidisciplinary experience gained from a number of instruments including the Advanced Very High Resolution Radiometer (AVHRR) and the High-Resolution Infrared Radiation Sounder (HIRS/2) instruments used for operational meteorological observations aboard the National Oceanic and Atmospheric Administration (NOAA) satellites, and the Coastal Zone Color Scanner (CZCS) instrument aboard the Nimbus-7 spacecraft.<sup>2</sup> The MODIS instruments will provide 12-bit imagery in 36 discrete bands between 0.4 and 15  $\mu\text{m}$ , with spatial resolutions ranging from 250 m to 1 km. In comparison, there are 5 NOAA/AVHRR channels at 1 km resolution and 20 NOAA/HIRS/2 spectral channels at a spatial resolution of 25 km on the current operational system. The Land Remote Sensing Satellite Thematic Mapper (Landsat TM) type instrument for a possible future EOS platform is the High Resolution Imaging Spectrometer (HIRIS). The Airborne Visible and Infrared Imaging Spectrometer (AVIRIS) aircraft instrument with 224 spectral channels is being used to simulate HIRIS measurements, in comparison to just seven Landsat TM channels. The first of the EOS observatory platforms is scheduled to have 5 instruments including MODIS. The prodigious output of the EOS instruments including the *hyperspectral* imagers will be approximately one terabyte of data every day.<sup>3</sup> New methods for visualizing multispectral time sequential satellite data need to be investigated to fully exploit these immense datasets efficiently.

Papathomas *et al.*<sup>4</sup> provide a historical perspective and summary of four current computer graphics systems for visualizing meteorological data that are primarily tuned to looking at one dataset or one data sequence at a time. An innovative approach to scientific visualization that has not yet been explored is the ability to combine visual data analysis with the spreadsheet style of presentation and computation. An effective combination of qualitative visualization techniques with formula-style calculation tools promises to radically alter by several orders of magnitude the volume of data that a single researcher can interactively explore and quantitatively analyze for patterns. The approach of the effort described in this paper is to extend the traditional numerical or scalar spreadsheet paradigm to a new domain of application for organizing, manipulating,<sup>5-7</sup> and processing multisensor image datasets interactively.

The exponential growth in the use of interactive spreadsheets on personal computers, since the commercial introduction of the VisiCalc spreadsheet in 1979 is put into historical perspective by Licklider.<sup>8</sup> The success of the first interactive financial spreadsheets was so great that they played an important role in the establishment of the multi-billion dollar personal computer industry and resulted in large productivity gains. The pervasiveness of large financial spreadsheet applications led to the trend of spreadsheet software becoming a driving force for the development and marketing of more powerful hardware.<sup>8</sup> The capability to enter data (once) in matrix format, perform mathematical operations on different cell combinations, and display the results nearly instantaneously has become an indispensable computing environment to a large number of users. The

ease of developing custom applications and performing multiple *what-if* exercises without having to reenter all the data has made the spreadsheet an essential tool for millions of users in the financial, engineering, and scientific communities.

Electronic spreadsheets have now become a pervasive easy-to-use problem solving tool in a variety of applications. The continued evolution of spreadsheets for financial tasks and their diversification to other fields attests to the success of the spreadsheet style of computation. Some of the novel areas to which the spreadsheet paradigm has been extended are discussed here in the context of guiding the design and development of the IISS. The effectiveness of using numerical spreadsheets for solving scientific problems, such as solutions to nonlinear equations, ordinary differential equations and even partial differential equations, has been discussed in detail by Orvis.<sup>9</sup> The ability to solve complex numerical problems within a spreadsheet supports the idea of extending the spreadsheet interface to the area of image analysis. The success of the spreadsheet model of computation has led to a number of extensions for supporting nonnumerical applications including database functions, Smalltalk data objects and bit images, logic programming, and the creation of interactive graphics applications that incorporate an animation capability.<sup>10</sup> The interactive graphics application developed by Lewis<sup>10</sup> is a unique extension of the spreadsheet model of computation. One of the guiding principles of their design was to maintain as far as possible the basic spreadsheet paradigm that many computer users are already familiar with and understand, in order to provide as consistent a user interface and style of computation as possible. This consistency principle has been useful in resolving some of the image spreadsheet design and implementation issues as discussed in the following section. Gutfreund<sup>11</sup> considers spreadsheets, along with matrices, Simplex tableaux, Pert charts, and Karnaugh maps to be examples of *reasoning tableaux* which provide a graphical representation of a system with a specified set of visual manipulation functions and operations. Such tableaux can be used not only for displaying data but also for exploratory data analysis. Reasoning tableaux can be generalized to handle a variety of complex datasets (such as 2-D charts and 3-D terrain maps), and the range of tools can be extended appropriately for each type of complex dataset within the same tableau. The IISS can also be considered to be in the category of an exploratory data analysis tool.

Spreadsheets along with their extensions and generalizations provide a general computational environment with capabilities for manipulating complex data types. Exploring the advantages of using such a powerful tool tailored to visualizing multiple sets of imagery, and organizing and executing image analysis tasks is the objective of the Interactive Image SpreadSheet (IISS) project at the NASA/Goddard Space Flight Center. In the IISS each cell can display an image which may be one of the following: (1) a complete original or derived image, (2) a partial original or derived subimage, (3) a projection of a multidimensional dataset such as a 3-D surface, or (4) a glyph (graphic symbol) representing an image. The term *image* is used in a general sense to include multisource datasets such as digitized geographical and geological maps, digital terrain models (DTM), contour plots, graphs, vector drawings, etc.

The term *derived* is used to indicate the result of applying formulas, operators, algorithms, or visualization techniques such as creating a surface from two images. The necessity to deal with a variety of data types is common in remote sensing applications.<sup>12</sup>

Processing, visualization, and analysis of multispectral satellite data using an interactive image spreadsheet has several primary objectives. The IISS facilitates the understanding of complex multispectral data time series via a selected set of interactive tools. In the multistep processing of satellite imagery, the spreadsheet improves the user's ability to trace the raw data through the complete correction, enhancement, and product derivation cycle. The IISS also enables the efficient comparison of different algorithms, and enhances the identification of deficiencies in the data processing and analysis steps. The spreadsheet thus provides feedback for improving the algorithms. The spreadsheet also allows an investigator to easily and interactively experiment with various combinations of channels, perform operations using a library of standard algorithms, and enter custom algorithms. Qualitative inspection and browsing of large sets of image arrays is conveniently provided in the IISS using an intuitive user interface for cell operations such as zoom, roam, animation, and function execution for a single cell or in synchrony for a group of cells. Animated time series and image alternation to interactively detect spatial, temporal and spectral differences between images has been found to be a very effective feature of the IISS.

The first commercial numerical spreadsheet, VisiCalc, was a 16kilobyte (KB) program written in 6502 assembly language to run on a 32KB Apple II with a bare minimum of features.<sup>8</sup> Modern spreadsheets on personal computers require 1megabyte (MB) or more of memory and can handle large amounts of data with complex calculations that only a few years ago would have taken prohibitive amounts of time. They also provide a variety of features including integrated graphics and database functions. The IISS approach for manipulating large complex image arrays takes advantage of the remarkable improvements in CPU performance, memory densities, and I/O channel capacity at continually decreasing costs. The current IISS prototyping environment consists of a multiprocessor Silicon Graphics Inc. 4D/340 VGX superworkstation configuration with 256MB of Random Access Memory (RAM), a 64MB/s data bus, and a graphics engine capable of rendering 1.1 million triangles per second as described in Sec. II.

Numerous remotely sensed sample datasets have been used to develop and evaluate the functionality of the IISS. Some of the test datasets were derived from the following instruments: (1) NOAA/AVHRR, (2) GOES/VISSR (Visible and Infrared Spin Scan Radiometer), (3) the first European Remote-Sensing Satellite/Along Track Scanning Radiometer (ERS-1/ATSR), (4) the Nimbus-7/Total Ozone Mapping Spectrometer (Nimbus-7/TOMS), (5) Landsat TM, (6) the Defense Meteorological Satellite Program/Special Sensor Microwave/Imager (DMSP/SSM/I), and (7) the Airborne Visible and Infrared Imaging Spectrometer (AVIRIS) with 224 registered spectral channels. Image analysis products that have been derived from multispectral data include color composites, contextual-based image enhancement, vegetation indices, thin cirrus classifications, perspective and stereo views.

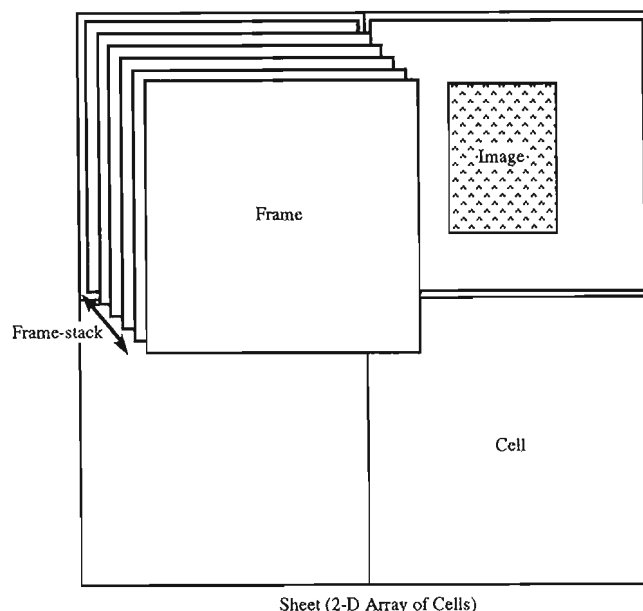


Figure 1. A diagrammatic representation of the organization of the IISS. The illustration shows the terminology for the composition of the sheet. The image data is organized in the sheet in terms of frames that are grouped together as frame-stacks within cells.

## I. IISS SOFTWARE SYSTEM DESIGN

### A. Spreadsheet organization

The IISS is a matrix of cells, where each cell contains an array of frames called a frame-stack, and each frame contains image data as illustrated in Fig. 1. The IISS, like numerical spreadsheets, is composed of an unlimited number of cells arranged in an arbitrary number of rows and columns.

All the cells in a given row have the same height (number of pixels in the y dimension) and all the cells in a given column have the same width (number of pixels in the x dimension). Resizing a cell affects all the other cells in the same row and column to maintain a gridlike interface typical of numerical spreadsheets. Image spreadsheets requiring more pixels than available on the display monitor may be scrolled through in an analogous manner to the traditional numerical spreadsheets. For example, Fig. 2 represents an 8×8 matrix of image cells. When the full 8×8 matrix of cells is visible on the screen then each cell occupies 160×128 pixels on a 1280×1024 pixel monitor. For larger cell sizes, only a portion of the IISS is visible on the screen and the remainder may be brought into view by scrolling along the columns or rows. For a 4×4 subset of neighboring cells, as shown by the grayed region in Fig. 2, each cell occupies 320×256 pixels using the full screen.

The matrix configuration of the IISS is adaptive and application dependent. In a typical configuration, each row of the spreadsheet is dedicated to an instrument channel or combination thereof while each column is reserved for a particular enhancement of the raw data as shown in **Configuration 1** of Fig. 2. A multispectral instrument which makes observations as a function of time can also be

easily accommodated in the IISS as shown in **Configuration 2** of Fig. 2. The spreadsheet rows represent different spectral bands and the spreadsheet columns represent the different temporal epochs. If time averages or composites are useful, they can be computed using formulas and included as an additional column of cells.

In addition to controlling the spatial layout of the image data the user also has control over the configuration of the frame-stack for each cell. The frame-stack can be used to load all of the channels of a multispectral image such as a 224 channel AVIRIS dataset into a single cell. The frame-stack can also be used to compactly hold a temporally contiguous image sequence. An effective visualization technique has been to view all or part of the matrix of cells as a synchronized animation. The frame-stack can be advanced (forward or backward) one frame at a time or animated as a video-loop. The time sequence in a single cell may also be viewed as a fullscreen animation. The Graphical User Interface (GUI) which controls video animation of cells, zoom, and roam, and the overlay of annotation, legends and geopolitical boundaries is discussed in later sections.

Two additional example IISS configurations are given in Fig. 3. Figure 3(a) schematically represents a multichannel dataset with the original data as well as the results of various processing steps. Since raw and processed datasets are available side-by-side, this enables artifacts in the final product to be attributed either to the original data or any of the processing steps. Many climatological datasets can be effectively viewed by using a 3-D spreadsheet where the third dimension contains images in a time sequence (by day, month, or year). An example of such a spreadsheet organization is schematically represented in Fig. 3(b) where the third dimension represents data from different years.

## B. Data structure description

The IISS data structure contains all of the data required to define, display, and manipulate a set of image data. A hierarchical tree-based data structure was designed to internally represent the IISS and facilitate the interactive manipulation of the image data residing in cells of the spreadsheet. A hierarchical scheme was chosen to define objects that closely corresponded to items being manipulated by the user in terms of Sheet, Cell, Frame-stack, and Frame as shown in Fig. 1, and to conveniently hide information from functions that use different parts of the data structure. Information hiding is essential to simplify software development and to prevent functions that manipulate small parts of the data structure from corrupting larger portions of the spreadsheet data structure.

The chart in Fig. 4 shows the overall organization of the IISS data structure. Note that some leaves in the tree represent individual members of the data structure while others refer to categories of items coalesced to simplify the chart. The Sheet is a two-dimensional array of Cells with each cell having its own screen dimensions subject to the constraint that all cells in a row share the same height and all cells in a column share the same width. Information at the Sheet level includes the screen size of the sheet, size of the cell matrix, screen size of the cells, and hardware features and limitations like double buffering the screen display, the size of the colormap (8 bits versus 12 bits for

example), and the capability to zoom an image by fractional amounts.

The row-column constraint on the dimensions of a cell enforces a strict matrix or grid organization. While it might be more convenient to have cells of unconstrained sizes to accommodate viewing images of different sizes, this would complicate the regular organization of a spreadsheet that affords its ease of interpretability. In fact the row-column spreadsheet interface affords a convenient mechanism for organizing large amounts of image data in a convenient fashion that maximizes the use of display screen space and reduces the clutter generated in other image processing environments. The lack of flexibility in changing both dimensions of a single cell is compensated by the ability to roam and zoom within any cell or to view a single cell at full screen resolution. The constrained three-dimensional organization of data in the IISS framework has been found to provide a practical and convenient interface for managing large volumes of data that can be hundreds of megabytes or more in size.

Cells are arranged in a matrix format and are accessed via matrix addressing since the two-dimensional arrangement of cells will typically remain stable during a user's session or will change infrequently. There are no *a priori* limitations (other than memory and window manager constraints) on the size of the matrix of cells that can be created and manipulated; the spreadsheet's size (the number of rows and columns of cells) can also be dynamically changed by the user.

Each cell contains one or more frames arranged as a doubly linked list and denoted by the term Frame-stack. Cells that are grouped together can be accessed by a linked list of pointers to cells that are part of the group. Two groups of pointers are maintained, one to keep track of all the frames in the Cell and another to keep track of the animation sequence when a subset of frames are looped together.

The frame data structure is the most complex since it contains all of the information pertaining to the image that is visible on the display as well as the original source of the image data.

## C. Graphical user interface (GUI)

The appropriate human-computer interface for controlling the IISS interactive functions is being designed and implemented to facilitate ease of use, consistency, portability, extensibility, and maintainability. In order to provide a consistent look and feel, and reduce development and maintenance time, the use of a preexisting library of user interface "building blocks" is considered essential. A number of such libraries exist, and all have associated graphical tools ("builders") to aid in program development. Libraries which were considered include (1) X-Windows based Motif for which there are commercial builders such as X-Designer™ (Imperial Software Technology, Ltd.), Builder Xcessory™ (Integrated Computer Solutions, Inc.), and UIM/X™ (Visual Edge Software, Ltd.) (The X-Window System is a trademark of the Massachusetts Institute of Technology. Motif is a trademark of the Open Software Foundation.); (2) InterViews from Stanford Univ. based on X-Windows and C++, (3) Transportable Applications Environment (TAE Plus) based on Motif and X developed at NASA/Goddard Space Flight Center, and (4)

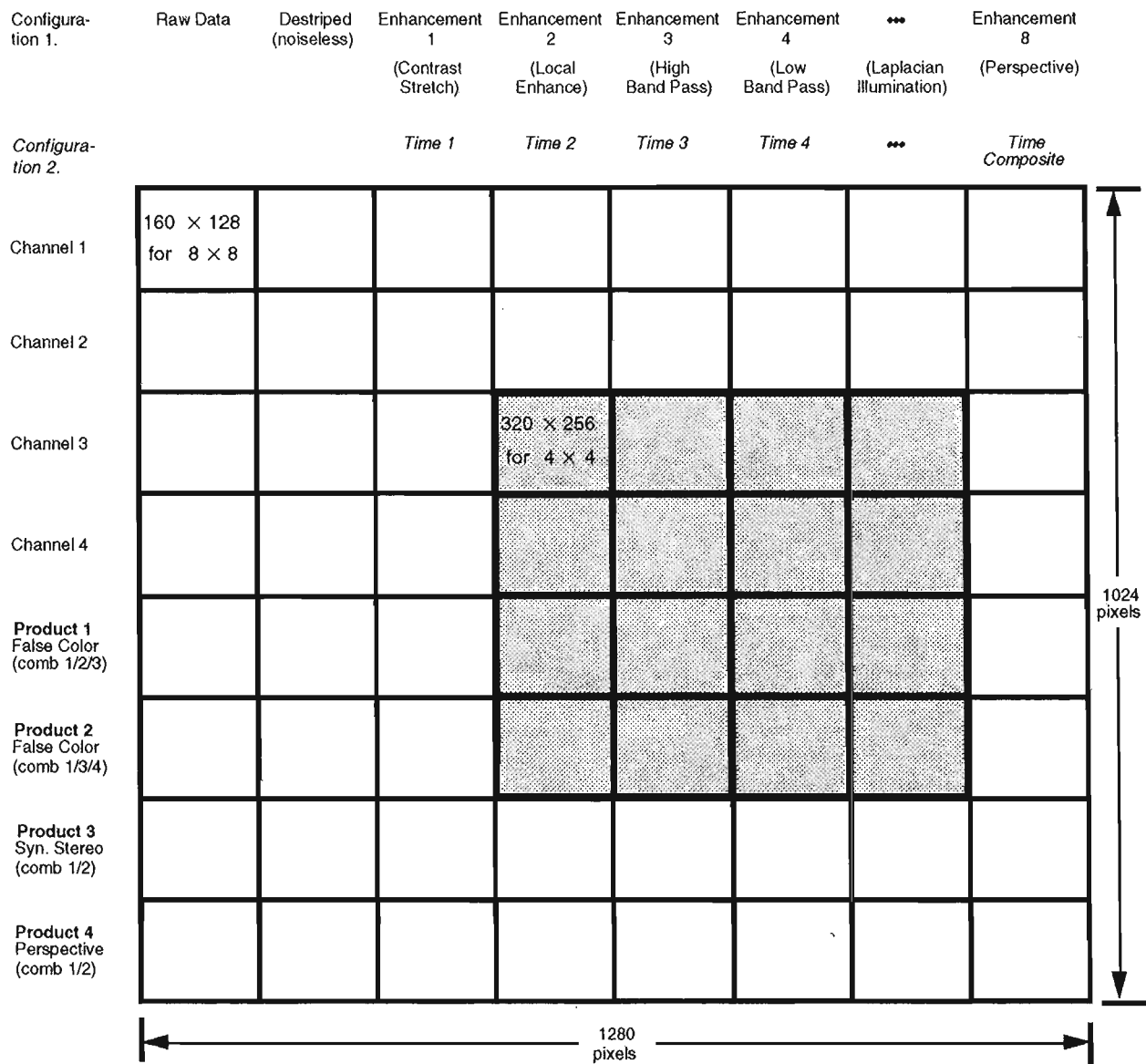


Figure 2. A multidimensional IISS consisting of an  $8 \times 8$  matrix of cells each containing a stack of images called frames. When just the  $4 \times 4$  array of cells (gray area) is displayed on the monitor each cell can occupy more screen space. **Configuration 1** shows multiple channels and products at various processing stages. **Configuration 2** shows channels over time and a time composite.

the `forms` user interface toolkit<sup>13</sup> based on **GL**, the Silicon Graphics Computer Systems **Graphics Library**. The `forms` library was chosen for several reasons: it provides high-level functionality, allowing the programmer to accomplish with one function call the equivalent of many Motif function calls. It is relatively easy to learn, produces compact executable code as compared with the other libraries, and its event handling is versatile, allowing for both callbacks and event queue processing. The source code for `forms` is available (unlike that for most implementations of Motif) thus allowing custom modification. Finally, the default appearance of various widgets such as buttons, selection items, text input fields, etc. is aesthetically pleasing.

The primary drawback of `forms` is its dependence on **GL**, which requires dedicated hardware, thus reducing portability. Libraries based on X-Windows graphics have

no such dependency, hence greater portability. Fortunately, the family of machines which support **GL** and in the future OpenGL is steadily growing, now including all Silicon Graphics machines, the IBM RS/6000, DEC workstations using Kubota Pacific Computer Inc. hardware, Sun Microsystems workstations, and personal computers using graphics adapters from Pellucid Inc. Also, the IISS code supporting the user interface is organized so as to facilitate conversion to Motif, if portability becomes an overriding concern. Currently the interactive image display functionality of the IISS relies on **GL**.

User interaction is handled via the `forms` graphical interface library, as well as direct cell interaction via mouse, cursor, and icon point-and-click inputs. Some example panels used in the IISS and designed using `forms` are shown in Fig. 5. The Options form at the left serves as

	Raw Data	Destriped (noiseless)	Radiometric Correction	Geometric Correction	Map Projection
Water Vapor					
Outgoing Longwave					
Vertical Wind Speed					
False Color					
Perspective					
Detect Clouds					

3(a)

	Water Vapor	Outgoing Longwave	Vertical Wind Speed
1993			
1992			
1991			
1990			
Apr			
May			
Jun			
Jul			
Aug			
Sep			
Oct			
Nov			
Dec			
Jan			
Feb			
Mar			

3(b)

Figure 3. Examples of organizing satellite data using the IISS. (a) Shows a specific IISS illustration of how multichannel data may be displayed and analyzed; the /// indicates empty cells. Note that the original data as well as the results of various processing steps are readily available. This enables the detection of artifacts in the final product which may be attributed either to the original data or processing steps. (b) Shows a 3-D spreadsheet where both the second dimension (rows) and third dimension (shown here as pages) represent temporal observations on a monthly and annual scale, respectively.

a control panel that can be accessed by the user at any time. The Options form provides scrollbar navigation over the cell rows and columns, menu items to handle I/O, data manipulation and module activation, and feedback fields. A dialogue form is shown in the middle of the figure, prompting the user for information to be used for formula evaluation. The formula panel allows the user to link to and edit any frame's formula, to include a variety of image processing operators in constructing the formula, and to set a few sheet-wide evaluation parameters. High-level forms components that can be seen in Fig. 5 include: (i) pushbuttons that are used to activate functions such as reading and saving the spreadsheet, and recalculation computation of all

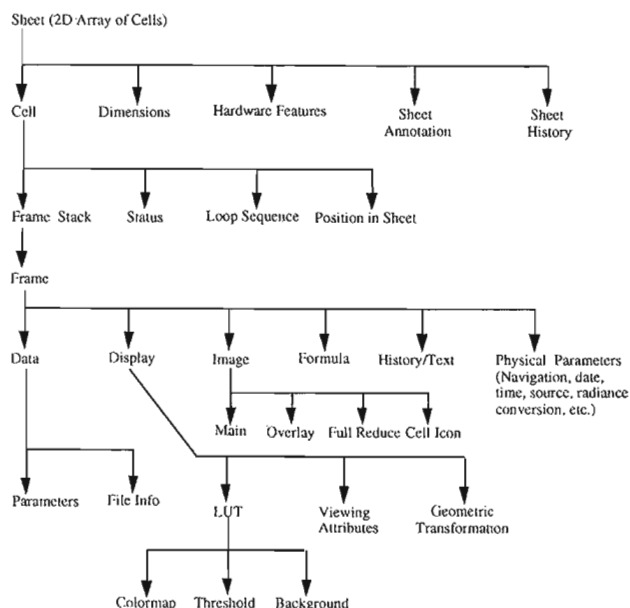


Figure 4. The IISS hierarchical object oriented data structure showing the organization of the image spreadsheet. Members of the data structure at the same level in the hierarchy have the same parent data field. Not all of the leaves in the hierarchy are unique members but may themselves be structures; for example, the Geometric Transformation field would include as members the scaling, translation and rotation parameters.

formulas that may be out-of-date, (ii) scrollbars that are used to determine which cells are in view in the image array window, (iii) input text fields to allow the user to specify the frame-id to link to and construct a frame formula, and (iv) lighted buttons that are used as on/off toggles to control user selectable options.

Direct cell interaction includes both popup menus for accessing all module commands, and keystroke accelerators for a subset of these. This unobtrusive interface was selected to maximize image viewing area at all times. It also eliminates the need for excessive mouse movement in order to control module functionality; the user is given control of a cell simply by moving the mouse cursor to within its borders. The interface forms need not be displayed on the same screen as the main spreadsheet window since GL supports distributed display of graphics across a network on any workstation supporting GL. A low cost 8-bit Silicon Graphics Personal IRIS, or IRIS Indigo can be used for the GUI and a Silicon Graphics workstation with high performance graphics can be used for the display of the IISS. Thus the screen space on a high performance monitor can be reserved for displaying the image array. Note that in this approach each display unit may require a separate input device such as a keyboard or mouse.

Dataflow or block diagram based visual interfaces to applications use nodes to represent processing modules and interconnections between nodes to indicate the movement of data from node to node. The dataflow approach to describing and executing algorithms has been used previously for image processing applications.<sup>14</sup> The Cantata tool in Khoros is a more complete implementation of a visual programming environment for signal and image processing in-

corporating conditional and iteration constructs, shared memory data transport between nodes, and distributed processing of nodes across remote machines.<sup>15</sup> A dataflow visual programming approach for multidimensional data analysis and visualization has been implemented in several commercial systems such as the IRIS Explorer™ application environment by Silicon Graphics,<sup>16</sup> and the Application Visualization System™ (AVS) model from Advanced Visual Systems Inc. (formerly Stardent Corporation).<sup>17</sup> Dataflow-based visual programming environments offer intuitive and flexible interfaces for providing data management, image processing and derived product calculation capabilities. Such visual language models can be used to help the user keep track of the complex processing steps needed to analyze large multichannel datasets. As the features supported by the IISS evolve, the complexity of formulas and operations between cells and frames may necessitate a visual dataflow interface.

#### D. Data visualization tools (*Superbrowse* Capability)

An inherent feature of the IISS is the capability of very quickly scanning or browsing through immense datasets. This IISS *superbrowse* capability allows the rapid scanning, in addition to careful examination, of data in the spatial, temporal, spectral, and gray scale domains. The IISS modules can display data in 2-D, 2.5-D, and full 3-D rendered modes as well as full 24-bit color binocular stereo with roam, zoom, and animation of time series for all of the above. *Superbrowsing* of huge datasets is realized in the independent program modules and the fully integrated features of the IISS. It also relies on the high performance of the workstation and mass storage system described below in Sec. II.

The spatial browser uses the *zoom* feature for providing high performance interactive roam, zoom, and reduce operations on 24-bit arbitrary sized images (that occupy up to 230MB of RAM). Smooth, fast zoom, and roam operations have been demonstrated on Landsat TM scenes as large as 8940×8420 pixels×24 bits that requires at least 226MB of RAM. Other large datasets for which interactive image rescaling and image translation has been found to be very useful, are from the GOES satellite VISSR instrument that produces visible images every half-hour at 12 000×15 200 pixel spatial resolution with six bits per pixel brightness quantization. A single GOES/VISSR image requires 182MB of memory just for the visible channel. Functional specifications for spatial browse include: (1) asynchronous interactive roam and zoom, (2) spatial synchronous interactive roam and zoom, (3) multichannel cell synchronous interactive roam and zoom, (4) multiple views of the same data, and (5) a fit-to-window view of the frame's entire image data (i.e., full reduce).

The temporal browser uses the *loop* feature to allow effective viewing of long time series data (i.e., 900+ frames) and bulky multichannel data (i.e., 224 channels) as an image sequence in groups of cells. Supported features include: (1) asynchronous frame animation, (2) spatial synchronous frame animation, (3) intercell frame animation, (4) multichannel synchronized frame animation, (5) time selected frame animation, and (6) stereo frame animation. Resolution may be dynamically changed while looping.

The spectral browser uses the synchronized *loop* feature in a multiple cell context. This system will allow

browsing of hyperspectral data (i.e., 224 channels) repeated in three cells of a 2×2 spreadsheet as the components of an RGB image with the fourth cell presenting the merged results as a full 24-bit color image.

The grayscale dynamic range browser will use the *lut* module, which is currently being designed, to examine datasets which have more than the standard 8 bits (256 levels) of grayscale information. It will have the capability to adjust the dynamic range of the data in the case of datasets with more than 8 bits per measurement and demonstrate new methods for interactively displaying datasets with more than 12 bits (>4096 levels) per measurement.

The volume browser will use the *flyby* module, developed at Goddard to visualize surface data, and a volume visualizer like the VIS-5D program developed at the Univ. of Wisconsin<sup>18</sup> to browse through 3-D and 4-D observed and simulated (numerical model) datasets. The volume browser will also be used to develop data fly-through animations. Rendering can be done in 2 1/2-D and future support for full 3-D is planned. Twenty-four bit color binocular stereo presentation using the *im\_stereo* module has now been integrated into the IISS in order to enable the unique capability for animation, panning, and zooming of stereo-pairs of image data. The *flyby* module, which has been incorporated into the IISS, is used to display and manipulate surface data that typically require one to four data fields (i.e., height field alone or three data fields to define a 24-bit textured image that is mapped onto a height field). The *flyby* module can also be extended to visualize datasets mapped on a globe. Hon<sup>19</sup> presented useful methods for efficiently visualizing remotely sensed data on a spherical surface using simulated mirrors.

The ability to group several cells together that contain related datasets and examine the datasets in a synchronized fashion has been found to be an extremely powerful interactive tool. Simultaneous synchronization of spatial location (*roam* and *zoom*), time sequencing (*loop*), display coordination, data probing, and cell interdependent formula evaluation pose a challenging design requirement in establishing consistent group dynamics. The end result is that group synchronization enables both qualitative and quantitative intercomparison of large volumes of data to be performed more efficiently.

#### E. Analytical tools

##### 1. Formula evaluation

Each cell and frame in the IISS supports the composition and analysis of formulas using operators acting upon datasets contained in one or more frames. Standard mathematical functions, such as addition, subtraction, multiplication, division, mean, standard deviation, etc., may be performed easily on individual image cells or on groups of two or more images. The IISS formula language is inherently an image processing language. Some of the features of the formula language are based upon IAX developed at IBM UK Laboratories,<sup>20</sup> PICO developed at Bell Laboratories,<sup>21</sup> and the syntax used in the symbolic mathematics package Mathematica<sup>22</sup> or commercial spreadsheets such as Lotus 1-2-3 and Microsoft Excel. IAX for example was designed to meet such requirements as providing a simple, concise and powerful language for programming new image processing tasks and prototyping algorithms, allowing specific





Figure 5. IISS graphical user interface forms examples showing the main interactive environment. On the left is the Options control panel, providing access to primary spreadsheet functions, and on the right is a dialogue form for editing a frame's formula. The form on the right showing pushbuttons, lighted buttons, scrollbars, text-input fields, etc. would be accessed only when needed in order to maximize the visual data displayed by the IISS.

application oriented routines to be easily specified using IAX primitives, facilitating efficient internal evaluation of operations, and enabling users to write their own extensions as compiled code.

The formula notation consists of methods for addressing elements such as cells or frames, methods for defining ranges of indices to access more than one element conveniently, a set of supported operators and functions. The following bracket notation is used in the construction of formulas:

- `[[ ]]` Indexing, element extraction, membership operation, e.g., `A1[[ 2, 12, 100 ]]` addresses pixel element in column 100 of row 12 in frame 2 of Cell A1
- `[ ]` For function parameters or arguments with the first letter of the function capitalized, e.g., `Invert [A1[[1]]]`
- `{ }` To denote lists, e.g., `{A1, A2, A3}`
- `( )` For grouping expressions, e.g., `(A1[[1]]+A2[[2]])/(B1[[1]]-B3[[2]])`.

Ranges can be specified explicitly using operators such as `Row [ ]`, `Column [ ]`, `Frame [ ]`, or `Cell[ ]`. Ranges

can also be defined implicitly using colon or double dot notation and wild card symbols. Rows of the spreadsheet are referenced by letters and columns by numbers starting with cell A1 in the upper left corner. Frames in a cell are numbered from one. Pixel addresses using rows and columns within a single frame are indexed from (0,0) representing the lower left corner of the image.

Many operators are overloaded to accommodate indexing over all pixels in a frame or image. For example, a simple addition expression such as

$$B1[[2]] = A1[[1]] + A2[[1]]$$

actually implies summation over all corresponding pixel elements in corresponding frames in cell A1 and A2 with the result being placed into corresponding frames in cell B1. Note that frames are explicitly specified in this notation. In Mathematica-like notation the expression above would be

$$B1 = \{A1[[f, x, y]] + A2[[f, x, y]], \{x, 0, 255\}, \{y, 0, 255\}\}.$$



Expressions need not explicitly identify the frames on which the operations are to be applied, as in the example above. Expressions that are based on the *current frame* (currently viewed frame) rather than an absolute frame index, need to be resolved carefully since the current frame index is constantly being changed as the user browses through the frames. For example, to look at several false color combinations interactively, the formula,

$$B2[[1]] = RGB[A1[[Current]],$$

$$A2[[Current]], B1[[Current]]],$$

will correctly show a spectral combination in frame B2[[1]] for the current frames in cells A1, A2, and B1. When the frames are advanced in cells A1, A2, or B1 then cell B2 will *automatically update* to reflect that the *current frames* have changed in other cells. This formula is then of the auto-update type with a special implied indication that operations use the current frame variable which can be dynamically updated. The auto-update mechanism in evaluating formulas can be considered to be a data-driven model of execution order.<sup>15</sup>

The dynamic update or data-driven computational model is usually the default mode of operation in standard spreadsheets since the calculations are usually rapid. However, in the IISS environment operations can be time consuming due to the volume of data and complexity of algorithms, so the default mode of operation is *evaluate upon user request* which is the demand driven model of execution. The IISS can be in auto-update (data-driven), manual-update (demand-driven), or view-driven mode depending upon the degree of real-time responsiveness required by the user. Some operations are more convenient to use in one mode than in the other; for example, RGB[ ] in auto-update and Fourier transform in manual update mode. In view-driven mode, which combines the features of auto-update and manual-update, only those frames necessary to display a currently visible frame are evaluated automatically.

There can be an inconsistency between the formula expression and the displayed data since we can delay calculation of a frame's formula. A frame which is inconsistent in this manner is called *out-of-date*, while a consistent frame is called *up-to-date*. The goal of automatic evaluation can therefore be expressed as keeping all frames up-to-date. A frame's status is readily accessible via the graphical user interface.

A frame becomes out-of-date when its formula is edited, or when a variable referenced within its formula changes value. The most common occurrence of variables is the use of frame references within a formula. For example, in the formula,  $B2[[1]] = A1[[1]] + 25$ , the frame reference A1[[1]] stands for its associated image data. In the datastream terminology, frame A1[[1]] is upstream of B2[[1]], and conversely B2[[1]] is downstream of A1[[1]]. These relations are transitive; so in the example any frames which depend on B2[[1]] would have to be downstream of B2[[1]] and by transitivity would also be downstream of A1[[1]]. Upstream and downstream frames can also be referred to as precedent and dependent frames, or backward-chain and forward-chain frames, respectively.

In either auto-update, view-driven, or manual-update mode the objective is to do the minimal amount of recalculation necessary to display a frame. Minimal recalculation is absolutely indispensable from a performance stand-

point when working with image data and is even a central issue in numerical spreadsheets involving complicated budgets or optimization problems. A frame dependency network is constructed as part of the minimal recalculation algorithm. When a frame has just been evaluated, all downstream frames become out-of-date and are candidates for recalculation. In the auto-evaluation mode the calculations begin immediately, while in the manual mode recalculation needs to be initiated by the user. In either case, the evaluation of a frame follows the same algorithm: (i) verify that all frames upstream of the current frame are up-to-date, (ii) reevaluate precedent frames that are out-of-date in a recursive fashion, (iii) evaluate the current frame, (iv) mark all downstream frames as being out-of-date, and (v) recalculate all downstream (dependent) frames recursively, when in auto-evaluation mode.

The checking and reevaluation of upstream frames in this algorithm leads to a recursive set of calls to the evaluation function, termed backward chaining. Backward chaining allows us to assert that if a frame is up-to-date, all of its upstream frames are up-to-date as well. Further, backward chaining ensures that a frame with several out-of-date frames upstream will only be reevaluated once which imposes a partial ordering on the evaluation sequence. The backward chaining approach can be extended to cover cases of circularity in the frame dependency network (recursive frame references). By definition, all frames in a dependency cycle are always out-of-date and evaluation results in each of the frames being evaluated only once, even in auto-evaluation mode.

The composition of formulas should support frequently required constants such as Pi ( $\pi=3.1415$ ) and E ( $e=2.71828$ ). Arithmetic (minus, power, divide, multiply, add), bit wise (left shift, right shift, and, or, xor, complement), and logical operators (not, and, or, equal to, not equal to, <, >, ≤, ≥) should be supported uniformly for scalars, vectors and 2-D matrices (images).

A large set of mathematical and numerical functions such as absolute value, rounding, trigonometric functions, exponentiation functions, convolution operators, transform operators and statistical operators, should be supported for scalars, vectors and matrices as appropriate. Functions that are more specific to images and involve image analysis algorithms<sup>23</sup> include the following. Typical arguments to these functions would include specification of one or more frames, and definition of parameters that the function requires.

Flip[ ]	flip the image about an axis
Histogram[ ]	calculate the image histogram
HSB[ ]	display images in hue, saturation, brightness color space
LocalMean[ ]	local space varying mean
LocalStdDev[ ]	local space varying standard deviation
Median[ ]	apply a median filter
Moments[ ]	various <i>n</i> th order moments
Overlay[ ]	overlay one image on top of another
Pad[ ]	extend the size of an image by padding
RGB[ ]	display false color combination in red, green and blue channels
ROI[ ]	select region of interest
Rotate[ ]	rotate the image
Scale[ ]	scale factors for the image

Surface [ ]        display datasets as a height field  
 Threshold [ ]     threshold the image  
 Translate [ ]      translate the image  
 Transpose [ ]     transpose (x,y) coord of the image

The complex example shown below could be done in one step or each step could be distributed to a different frame. The following algorithm,

```
Histogram[
  ROI[
    Scale[ Median[ A1[[3]], 3, 3 ],
          2.5, 1.5 ],
    0, 0, 100, 100 ]
]
```

applies a 3×3 median filter to frame A1[[3]], then scales the resulting image by a factor of 2.5 in the x direction and 1.5 in the y direction, extracts a 101×101 pixel subregion starting at (0,0) and then shows the histogram for this region of interest. So the result of the formula is not another image frame but a graph that would be displayed in its own frame.

Formula expressions also support IISS functions for copying or duplicating information between cells and

frames, for adjusting rows and columns, and accessing data values in frames. The default mechanism for copying data across frames is to conserve memory and share pointers rather than actually copying data. The physical memory is freed when there are no frames that require the data. Operators for conveniently accessing a group of data elements such as a linear range of pixels are also necessary. The Element[ ] operator can be used to retrieve the displayed gray value (*gray*), 24-bit color values (*rgb*), or the underlying original data used to determine the value at a particular location (*data*). The general syntax of the Element[ ] operator is Element[*cell*, *frame*, {*red*, *green*, *blue*, *rgb*, *gray*}, {*display*, *data*}, *row*, *col*] where *cell* represents a list of cell addresses, *frame* a list of frame numbers, then a data type, *row* a list of row numbers, and *col* a list of column numbers. A specific use of the Element[ ] operator to find the gray values for pixels in rows 5 through 15 in column 255 in the current frame of cells B2, A3, and B1 is

```
Element[ {B2, A3, B1}, {Current}, {gray}, {data},
         {5, 6, 7,..., 15}, {225} ].
```

The result is an 11×3 matrix or more generally a list of three lists of gray values:

```
{ {B2[ [ current frame, 5, 255 ] ], B2[ [ current frame, 6, 255 ] ],..., B2[ [ current frame, 15, 255 ] ] },
  {A3[ [ current frame, 5, 255 ] ], A3[ [ current frame, 6, 255 ] ],..., A3[ [ current frame, 15, 255 ] ] },
  {B1[ [ current frame, 5, 255 ] ], B1[ [ current frame, 6, 255 ] ],..., B1[ [ current frame, 15, 255 ] ] }.
```

In order to enable the user to input a variable number of parameters for functions a notation that identifies each parameter can be used as in Mathematica. When parameters are identifiable then an added benefit is that the order in which the parameters are specified becomes flexible and thus simpler for the user to use functions. The above Element[ ] example could be specified as

```
Element[ frame→{Current}, cell→{B2, A3, B1},
        row→{5, 6, 7,...,15}, col→{225}, type
        →{gray} ].
```

In addition to using row and column based access to cells, some newer numerical spreadsheet designs allow cells to be referenced by combinations of variable names, and contain a separate window showing all of the formulas entered in the spreadsheet.<sup>24</sup> One of the advantages of this approach is that complex relationships among frames (variables) are accessible immediately through mnemonic names rather than through a long sequence of searches through individual frames. Providing such a feature in the IISS would be important when frames display the results of an interrelated flow of processing. A cascade of calls to image analysis operators with feedback would be an example of a situation wherein the ability to graphically view all of the frame formulas simultaneously would be advantageous.

## 2. Image processing tools

Many remote sensing related image processing techniques will be available in the IISS as a toolbox and via formula evaluation including: destriping, interactive and automatic bad-line removal, spike removal, interactive contrast stretching using level slice, histogram equalization, local enhancement, high and low band-pass filters, Laplacian illumination-based edge enhancement, and perspective rendering of surfaces. For example, the problem of correcting raw satellite data from multiple sensors and hundreds of spectral bands will be a crucial step in obtaining images that can be compared across channels and over long time periods required for climatological studies. Comparing the results of applying different geometric and radiometric correction algorithms to a set of images would be facilitated in the IISS environment. Different steps of the correction could also be readily previewed and the accuracy over a sequence of images could be verified at the full resolution for very large images (such as Landsat TM) by roaming and zooming through the image to locate the key feature points, then alternating several of the images in a sequence or viewing them side-by-side.

Some of the standard image enhancement operations have been implemented in a variety of software packages such as AOIPS,<sup>25</sup> VICAR (Video Image Communication and Retrieval/JPL), ELAS (Earth Resources Laboratory Applications Software/NASA Stennis Space Center), LAS (Land Analysis System formerly known as Landsat-D As-

assessment System/NASA Goddard Space Flight Center), View (Lawrence Livermore National Laboratory), Khoros (University of New Mexico), NCSA XImage (University of Illinois), and the ImageVision Library (Silicon Graphics, Inc.) all of which can be used as a guide in the design of the IISS functions. The IISS system will support the user's own special enhancement and analysis algorithms to be accessible from within the IISS environment. Some specialized algorithms that the IISS can be effectively used with include those for estimating rainfall,<sup>26</sup> for classification of cloud-free and cloud-filled pixels,<sup>7,27</sup> for estimating cloud-top height from stereo data,<sup>28</sup> for computing vegetation indices,<sup>29</sup> or creating cloud free composites.

## II. IISS HARDWARE SYSTEMS

The current generation of moderate cost superworkstations have multiple processors, a very large amount of general purpose memory, high-performance data buses, mass storage typical of a minisupercomputer, and pipelined or other advanced architectures for graphics and image operations. A set of hardware specifications are given based on ideal spreadsheet performance requirements, for comparison with current and future potential system configurations. The specifications and performance of the current hardware implementation is first presented and then followed by a brief description of hardware capabilities expected to be widely available in the next few years.

It is revealing to realize that the floating point performance of digital computers has increased exponentially by more than a trillion times over the past five decades.<sup>30</sup> The fastest microprocessors now have a clock speed comparable to a Cray supercomputer after doubling in speed (as measured in instructions per second) every 18 months over the past decade. The capacity of solid-state dynamic random access memory (DRAM) chips has been doubling every 18 months for over two decades and is currently 16Mbits per chip.<sup>31</sup> Recording densities of storage media has been doubling every 3.5 years so that currently one terabyte of data can fit on six D2 format large cassettes instead of 6667 nine-track tapes. These exponential rates of change in hardware performance coupled with comparable decreases in price allow us to infer that the high performance superworkstations of today (like the one shown in Fig. 6) will become affordable desktop personal computers in the near future.

### A. Hardware performance requirements

The use of screenspace can be enhanced when dual 24-bit color monitors are available. The IISS workspace can then be displayed on one monitor with the second monitor used for the Graphical User Interface (GUI) and other information such as an overview image with a box showing the region of current interest. Standard image operations, like zoom, reduce, scroll, animate, etc., must be executed nearly instantaneously. It must be possible to roam quickly and smoothly around a number of subimage cells simultaneously. The subimage cells would be derived interactively from up to ten 4096×4096 pixel 24-bit images. While a 30 frame/s animation of these images is in progress, the user should have the ability to smoothly roam, zoom, and enhance. At least 100 GB of on-line storage is essential with

access to remote storage devices across a high performance network interconnection to supercomputers and other databases.

The workstation must have exceptional input/output (I/O) bandwidth in order to be able to move large amounts of image data from local and remote storage to the screen (framebuffer). The bandwidth of the system bus connecting main memory (RAM), CPU, display memory, and I/O bus must be very high (at least 150MB/s) in order to support fullscreen 24-bit animations at 30 frames per second. The RAM must be very large so that reasonable sized IISS arrays and animations may be accessed at the 150MB/s rate. For example, full sized multispectral Landsat scenes, full GOES images and moderate quality thirty second animations can be processed interactively with 256MB of memory. High-performance access to the local mass storage system on a hard disk at a rate of at least 20MB/s is necessary. This would allow 200MB of data to be loaded in 10 s, and would permit 512×512 pixel×8-bit animation at 60 frames/s directly from mass storage. The same performance across a local or wide area network would allow these operations to be performed directly using the Goddard National Space Science Data Center (NSSDC) databases and eventually the Earth Observing System Data and Information System (EOSDIS). Real-time data compression and decompression techniques with efficiencies of 100:1 would be necessary for achieving long (for example, 20 min) animations of observed and model datasets.

CPU resources need to be high in order to perform simultaneous manipulations on numerous image cells and the calculation of several products based on multiple cells interactively. An ideal workstation would dedicate an individual, fast RISC CPU for each cell of the IISS in a multiprocessor environment.

### B. Hardware capabilities

A Silicon Graphics Computer Systems (SGI) supercomputing workstation SGI 4D/340VGX with the following major specifications, as illustrated in Fig. 6, was chosen as the platform to demonstrate the IISS. The system has four fast CPUs for a total of 120 MIPS and 36 MFlops, a large, main CPU memory of 256MB, a high-performance (20MB/s) data transfer rate between RAM and image framebuffer, fast graphics (more than 1 000 000 triangles per second), a large amount of mass storage (10GB at 3.6MB/s or faster SCSI-2 disk drive systems), and parallelizing FORTRAN and C compilers to take advantage of multithread symmetric multiprocessing.

The IISS is displayed on a high resolution color monitor (1280×1024) with double-buffered 32-bit color (for 64 bits of color), 8-bit stencil, 24-bit z-buffer, 32-bit texture, 4-bit overlay, and 8-bit window plane layers for a total of 140 bits. Standard image operations, like zoom and reduce, are nearly instantaneous for reasonable sized images. Smooth and fast zoom and roam operations have been demonstrated using three Landsat TM images each as large as 8940×8420 pixels×8 bits requiring at least 226 MB of RAM. Animations of 512×512 pixels×8 bits at 70 frames/s and color frames of 512×512 pixels×24 bits at 37 frames/s with full interactive control have been demonstrated. The workstation has acceptable input/output (I/O) and system bandwidth. The data transfer bandwidths for the workstation are as follows:



- System bus from memory (RAM) to CPU at 64 MB/s,
- RAM to I/O bus at 34MB/s,
- RAM to framebuffer at 20MB/s for 8-bit images and 30MB/s for 24-bit images.

The RAM is large (0.25GB) so that extended IISS arrays can be displayed interactively and long animations (900 frame 8-bit and 300 frame 24-bit color 512×512 pixel) accessed at 30 frames/s. Moderate performance access to the local mass storage system on hard disks at a rate of 2MB/s has been demonstrated on the current SCSI disk drives. Striping of multiple SCSI-2 disks has allowed us to increase the rate to 3.6MB/s. Real-time data compression and decompression techniques have not yet been implemented. To access remote storage, the high performance UltraNet™ network (125MB/s) interconnection to Goddard databases and supercomputers has been installed. With the four processor SGI 4D/340 VGX workstation an individual high-performance RISC CPU can be assigned to each of four cells simultaneously, or equally partitioned among the total number of active cells, or formula evaluations.

### C. Next generation hardware performance expectations

The next generation of workstations will use data paths that are 64 bits wide instead of 32 bits wide. It is expected that transfer rate requirements of 150 MB/s will be available. The 64-bit architecture will also enable larger memory configurations of at least 4GB. Fullscreen 24-bit animations of 30 s and about 5 min of broadcast quality digital video could then be stored in RAM. Low cost striped SCSI-2 fast and wide hard-disk drives using RAID (redundant array of independent disks) architectures should reach the 30MB/s data transfer rate level. Increased data densities and high-performance I/O systems will make accessing 100GB mass store digital libraries and redundant disk arrays more practical. Multigigabyte RAM disks may also become cost effective allowing an intermediate mass storage system which is of higher performance than hard disk-based systems. Data compression and storage systems using magnetic tape or optical media will allow complete remote sensing databanks to be kept on line. The display unit resolution should increase to 2048×2048 with a separate hardware lookup table provided for each of the three primary colors instead of the single lookup table used in current systems.

CPU speeds will increase up to ten times or more over the current system providing greater capability to do interactive processing of all spreadsheet cells. Using coarse grain parallelism, one processor can be assigned to each cell even for large spreadsheets. To achieve image processing operations at video rates will require a special Image Engine analogous to the Geometry Engine available on the SGI 4D/340 VGX, or the utilization of a massively parallel supercomputer such as the Maspar via a high speed network. New high speed networks becoming available include nonproprietary ones such as fiber distributed data interface (FDDI) rings, synchronous optical networks (Sonet), Fast Ethernet, high-performance parallel interface (Hippi) switches, or asynchronous transfer mode (ATM) technology that range from 100Mb/s to several gigabits per second.<sup>32</sup>

## III. RESULTS

The IISS concept has been implemented by integrating the data visualization modules into the spreadsheet matrix using the data structures as described above. Many of the IISS features are carried out by the formula evaluation mechanism. Numerous interactive IISS examples have been demonstrated on the SGI 4D/340 VGX IISS workstation using real datasets, to investigate the performance requirements and interactive control features necessary for displaying data in various data formats, and to prototype zoom, reduce, roam, animation, and surface rendering functions. The results from these prototype implementations have been integrated into the design for a functionally complete interactive spreadsheet.

A variety of remote sensing datasets have been used to illustrate some of the IISS features currently available or under implementation. The datasets include GOES/VISSR images of hurricane Hugo, NOAA/AVHRR global mosaics of water vapor and outgoing long wave radiation, Nimbus-7/TOMS daily global ozone data for a twelve year period, complete Landsat TM scenes of the Chernobyl area and of the Kuwait area, GMS infrared imagery for comparative rain estimation, and DMSP/SSM/I microwave radiometric data for global rain rate estimation. The wavelength regimes covered by this group of instruments (including AVIRIS) are presented in Fig. 7 for reference. It provides a quick overview of the different channel spectral responses for the various instruments.

Figure 8 shows a 3×3 cell matrix configuration displaying monthly averages of TOMS ozone data.<sup>33</sup> Eight of the cells contain 12 frames each representing monthly averages for the years 1982 through 1989 and the lower right hand cell, C3, has 149 frames of monthly averages for the period from November 1979 through March 1991. The IISS allows the independent or synchronized looping of frames in any combination of cells. In this example the middle and upper left cells have been grouped together as indicated by the yellow border around the cells and the yellow cell icons in the scrollable grid matrix at the top of the Options panel. The two cells have been grouped together in order to manipulate the data contained in the two frames in synchrony.

Figure 9 gives an example of an IISS application using Landsat TM data. This figure shows four channels of Landsat TM data of the region surrounding Chernobyl, USSR, before and after the nuclear reactor accident on April 26, 1986, and various products which can be produced from these data. Rows 1, 3, and 5 display Landsat imagery between 0807 and 0820 Universal Time Code (UTC) for June 6, 1985, April 29, 1986, and December 2, 1986. These images were taken the year before, three days after, and several months after the accident. Channels 3, 4, 5, and 6 are displayed in columns 1 through 4. The wavelengths of these channels are given in Fig. 7. Rows 2, 4, and 6 show some products which can be obtained from these channels. The relative darkness of rows 5 and 6 is a result of the low sun angle found at this latitude in December.

Columns 1 and 2 in rows 2, 4, and 6 are color composites of channels 3, 4, and 5 in which vegetation cover and bodies of water are enhanced. Note that vegetation appears red in column 1 and green in column 2. Seasonal variation in vegetation is clearly evident when viewing these data. A vegetation index derived using Landsat TM

# INTERACTIVE IMAGE SPREADSHEET (IIS) WORKSTATION & SERVERS

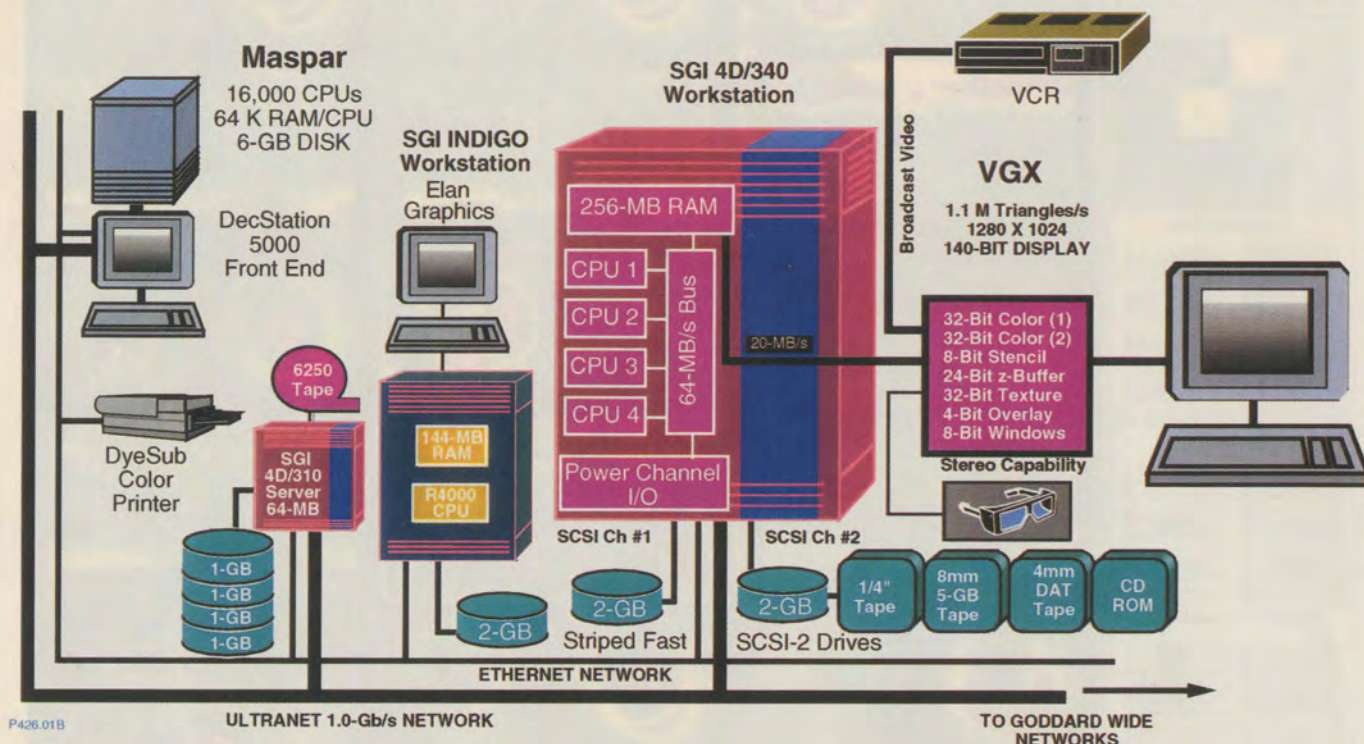


Figure 6. IIS superworkstation hardware configuration illustrating multiprocessor architecture, high performance graphics system, large memory, large mass storage system, and a high performance network.

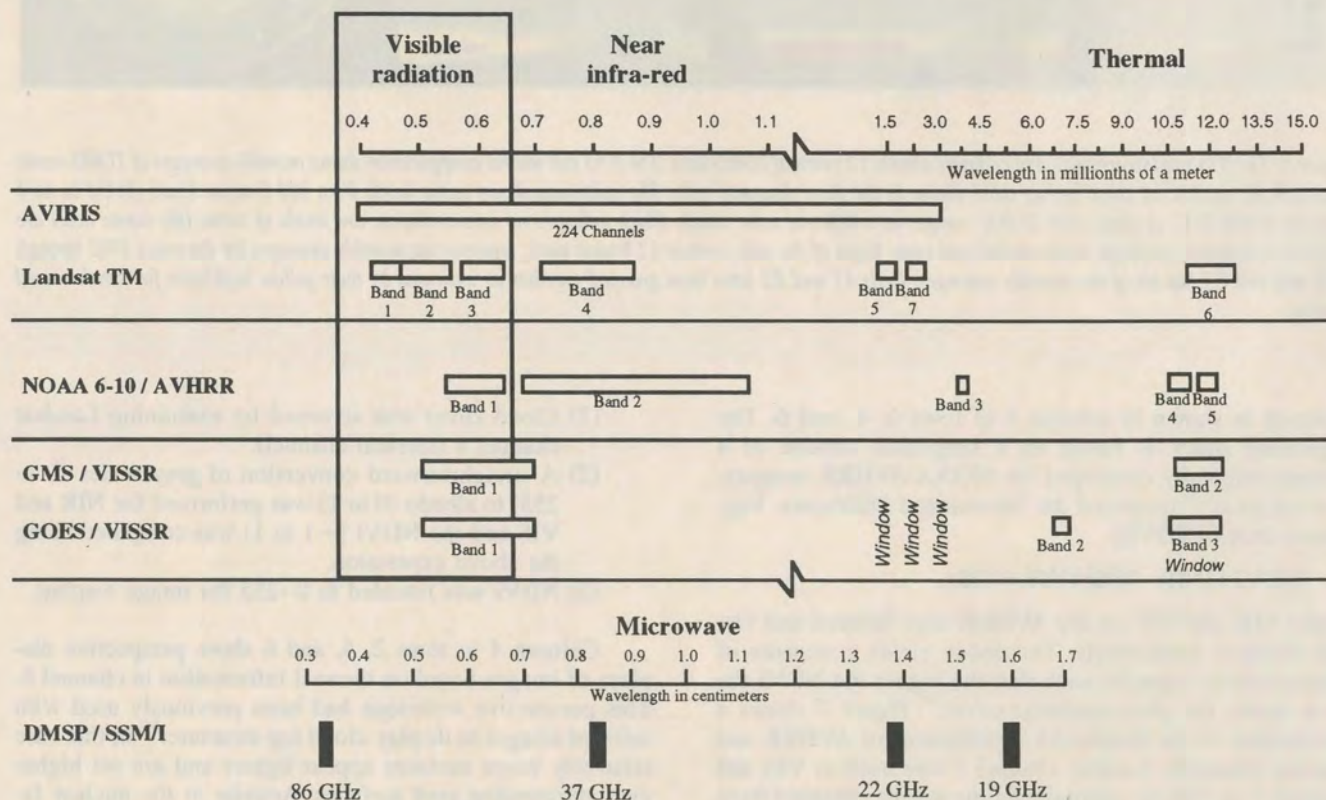


Figure 7. Wavelengths of various observation bands used by AVIRIS, Landsat TM, NOAA/AVHRR, Geostationary Meteorological Satellite (GMS)/VISSR, GOES/VISSR, and DMSP/SSM/I instruments.



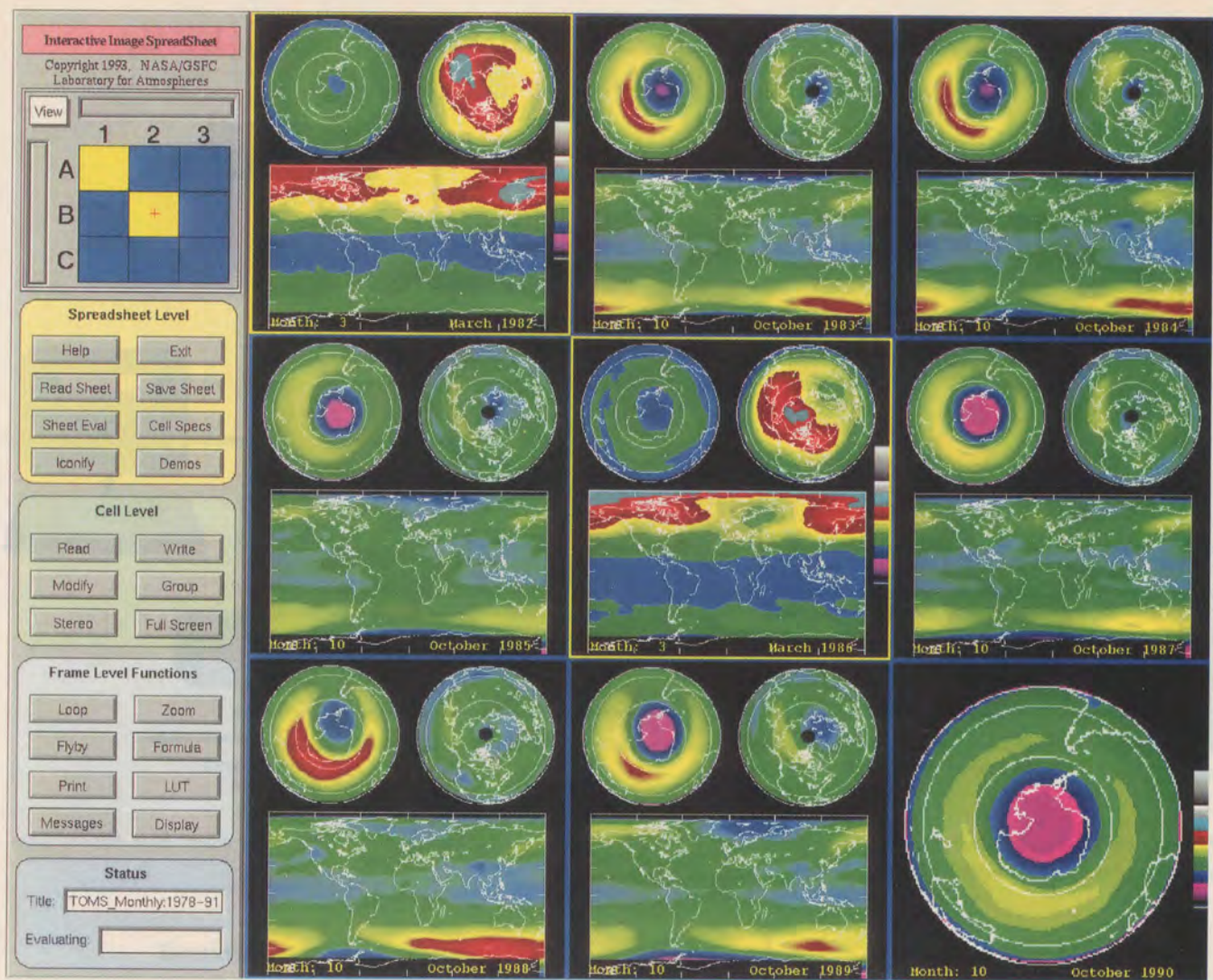


Figure 8. The IIS used to organize and compare almost 12 years of TOMS data. The  $3 \times 3$  cell matrix configuration shows monthly averages of TOMS ozone data with the associated color lookup table shown in the three diagonal cells. The colormap shows ozone levels from 100 Dobson Units (D.U.) in dark magenta to 650 D.U. in white with 25 D.U. ranges in a different color shade. Black indicates no measurements, low levels of ozone (the ozone hole) are depicted in magenta and high levels are red and cyan. Eight of the cells contain 12 frames each, representing monthly averages for the years 1982 through 1989 and cell C3 has all of the monthly averages. Cells A1 and B2 have been grouped together as indicated by their yellow highlights for synchronized viewing.

channels is shown in column 3 of rows 2, 4, and 6. The vegetation index is based on a simplified version of a scheme originally developed for NOAA/AVHRR imagery. Deering *et al.*<sup>29</sup> proposed the Normalized Difference Vegetation Index (NDVI):

$$\text{NDVI} = (\text{NIR} - \text{VIS}) / (\text{NIR} + \text{VIS}),$$

where NIR and VIS are the AVHRR near infrared and visible channels respectively. This index yields a measure of photosynthetic capacity such that the higher the NDVI the more active the photosynthetic cover.<sup>34</sup> Figure 7 shows a comparison of the bandwidth distribution for AVHRR and Landsat channels. Landsat channel 3 was used as VIS and channel 4 as NIR to approximate the results obtained from AVHRR. The following steps were used to convert the 8-bit (0 to 255) Landsat data to NDVI:

- (1) Cloud cover was screened by examining Landsat channel 6 (thermal channel).
- (2) A straightforward conversion of gray scales (0 to 255) to albedo (0 to 1) was performed for NIR and VIS and the NDVI (-1 to 1) was computed using the above expression.
- (3) NDVI was rescaled to 0-255 for image display.

Column 4 in rows 2, 4, and 6 show perspective displays of images based on thermal information in channel 6. This perspective technique had been previously used with infrared images to display cloud top structure.<sup>35</sup> In this case relatively warm surfaces appear lighter and are set higher than surrounding cool surfaces. Activity at the nuclear facility can be deduced by examining the southern part of the cooling lake in the right hand corner of these images (only

a small part shown here). The plant was operating in June 1985 and warm water discharge is evident in the southern end of the lake. When the plant was not operating in April 29, 1986 (three days after the accident) the water in the cooling pond appears uniformly cool. The perspective in column 4, row 6 again shows warm water discharge indicating that one of the undamaged reactors was brought back on line by December.

In Fig. 10 the IISS is shown with a 2×2 cell matrix configuration displaying color composited SSM/I data in various 2-D and rendered perspective 3-D views. The top frame of the upper left cell displays the Northern hemisphere, the upper right cell shows the Southern hemisphere, the lower right cell, a rendered perspective view of the Southern hemisphere using texture mapping with lights, and the lower left cell B1 shows a close-up perspective view rendered using only texture mapping of North America. The height field used for the rendering is related to precipitation. The 85.5 GHz (vertically polarized), 85.5 GHz (horizontally polarized) and 37.0 GHz (vertically polarized) channels are used for red, green and blue, respectively.<sup>36</sup>

#### IV. CONCLUSIONS AND FUTURE WORK

A new scientific visualization approach to organizing, browsing, and analyzing large volumes of image data has been developed by extending the traditional numerical spreadsheet style of computation to the image domain. The Interactive Image SpreadSheet (IISS) environment integrates quantitative analysis with a variety of intuitive, novel, and highly responsive visualization tools to enable the quick inspection and validation of data and to explore

the development of new analysis algorithms. The data viewed in each cell in the IISS has the capability to be independently translated, scaled, animated, or enhanced. When the cell is part of a group all of the transformations are applied to each member of the group synchronously. A large image dataset can be viewed using multiple frames without duplicating storage requirements.

The powerful formula evaluation capability enables the user to construct and test complex algorithms without having to worry about the details of input and output operations or programming in a low level language. Formulas can consist of a mixture of algebraic relationships between frames and function operators involving frames. The necessity to generate and keep track of a large number of intermediate calculated images, in addition to the large volume of original data, in order to evaluate and compare the strengths of different algorithms is facilitated by the 2-D cell and 3-D frame-stack organization provided by the IISS. A minimal recalculation strategy is used to update out-of-date frames using a frame dependency network to eliminate any unnecessary computations and ensure responsiveness whenever possible. Recalculations are initiated automatically and are demand driven when the user requests the display of an out-of-date frame while in the view-driven mode.

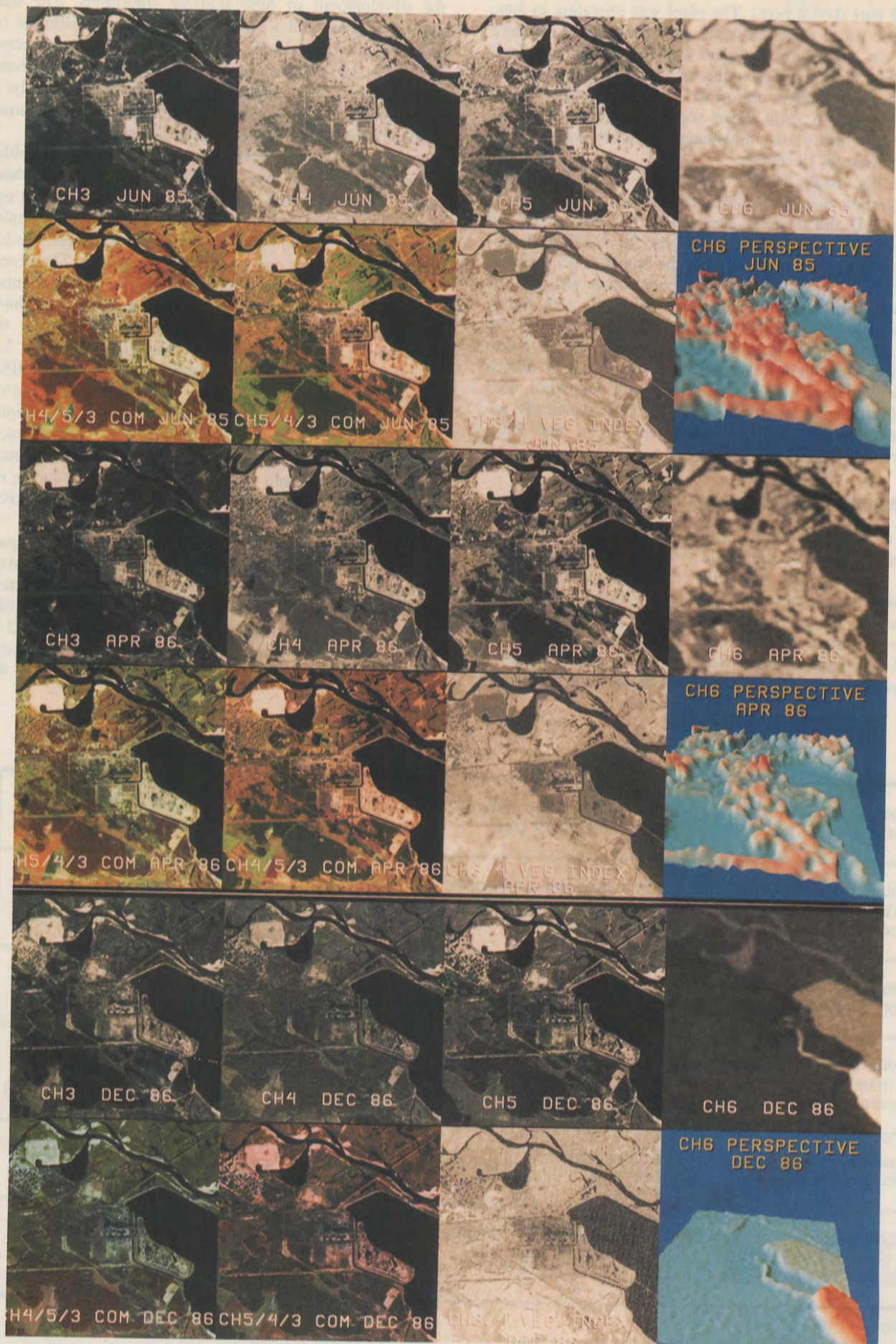
The IISS is being prototyped at the Goddard Space Flight Center on both a high performance graphics/image processing workstation as well as a low cost workstation. The high performance workstation is a multiprocessor Silicon Graphics Inc. 4D/340 VGX with four MIPS 32-bit R3000 processors (rated at 120 MIPS and 36 MFlops), a fully configured memory of 256MB, a graphics engine that

Landsat channel 3 (visible) Jun 6, 1985 0819 UTC	Landsat channel 4 (near IR) Jun 6, 1985 0819 UTC	Landsat channel 5 Jun 6, 1985 0819 UTC	Landsat ch 6 (thermal) Jun 6, 1985 0819 UTC
Color composite of channels 3 (blue), 4 (red), and 5 (green) for Jun 6, 1985.	Color composite of channels 3 (blue), 4 (green), and 5 (red) for Jun 6, 1985.	Vegetation index derived from channels 3 and 4 for Jun 6, 1985.	Perspective derived from thermal signature (channel 6-Jun 6, 1985)
Landsat channel 3 (visible) Apr 29, 1986 0820 UTC	Landsat channel 4 (near IR) Apr 29, 1986 0820 UTC	Landsat channel 5 Apr 29, 1986 0820 UTC	Landsat ch 6 (thermal) Apr 29, 1986 0820 UTC
Color composite of channels 3 (blue), 4 (red), and 5 (green) for Apr 29, 1986.	Color composite of channels 3 (blue), 4 (green), and 5 (red) for Apr 29, 1986.	Vegetation index derived from channels 3 and 4 for Apr 29, 1986.	Perspective derived from thermal signature (channel 6-Apr 29, 1986)
Landsat channel 3 (visible) Dec 2, 1986 0807 UTC	Landsat channel 4 Dec 2, 1986 0807 UTC	Landsat channel 5 Dec 2, 1986 0807 UTC	Landsat ch 6 (thermal) Dec 2, 1986 0807 UTC
Color composite of channels 3 (blue), 4 (red), and 5 (green) for Dec 2, 1986.	Color composite of channels 3 (blue), 4 (green), and 5 (red) for Dec 2, 1986.	Vegetation index derived from channels 3 and 4 for Dec 2, 1986.	Perspective derived from thermal signature (channel 6 Dec 2, 1986)

9(a)

Figure 9. (a) Annotation for image spreadsheet of Landsat TM data and various products. (b) Example image spreadsheet matrix of 4×6 cells of Landsat TM data and various products. Descriptions of the cell contents are given in (a).





9(b) cont.



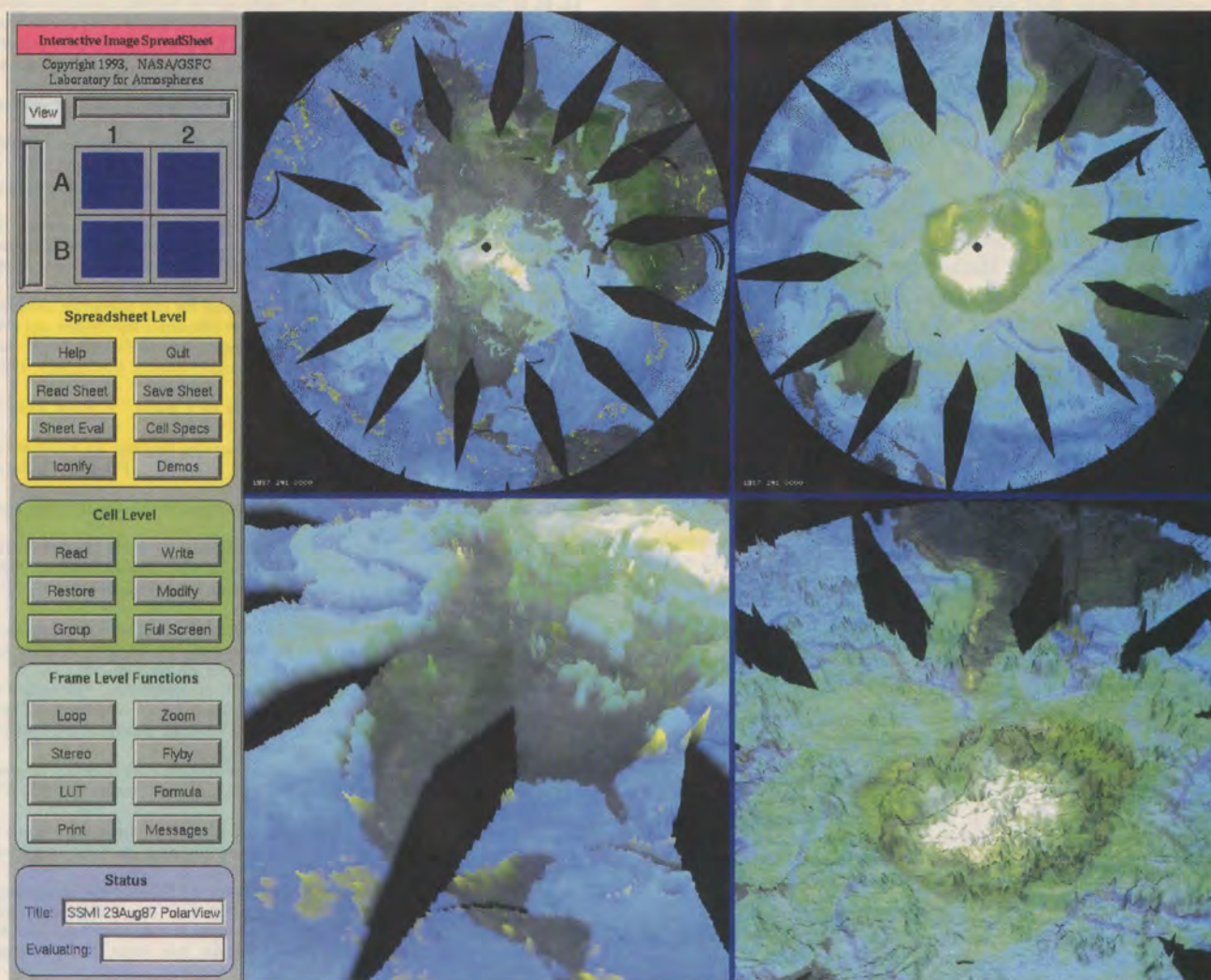


Figure 10. The  $2 \times 2$  IIS cell matrix displays color composited SSM/I data in various 2-D and rendered 3-D perspective views using the *flyby* module. Cells A1 and A2 show north and south polar stereographic projections, respectively. Cells B1 and B2 show perspective renderings of a portion of the data in cells A1 and A2, respectively, using a height field which is the 85.5 GHz horizontally polarized channel that is related to precipitation.

can render 1.1 million triangles per second, stereo capability, and a 10 GB high performance (3.6MB/s) mass storage system. Work with very large remote sensing satellite datasets has emphasized the need for even higher performance systems which are now becoming commercially available. Most of the functionality of the IIS has also been demonstrated on a low-end low-cost Silicon Graphics Indigo workstation to ensure the accessibility of the software to a large group of investigators.

Many IIS functions have been tested using a number of multispectral remotely sensed datasets that simulate new satellite instruments including those on the EOS platforms. Some of the visualization and quantitative functionality that have been developed include: (i) high performance spatial, temporal, spectral and volume browsing including 24-bit color binocular stereo with simultaneous zoom, roam, and animation capability, (ii) display of arbitrary sized image cell arrays in the IIS format with the goal of "mining" large image digital archives for interesting phenomena, (iii) synchronized visual browsing of data in multiple cells that have been grouped together, and (iv) algorithm develop-

ment using image formula evaluation with an initial set of mathematical functions and image processing operators. The IIS system has made possible the effective manipulation and analysis of very large datasets that are currently available. For example, full-sized maximum resolution multispectral (7 channel) Landsat TM scenes, full earth maximum resolution 182MB GOES images and up to 900 frame grayscale animations using 236MB of memory have been visualized in a highly interactive manner on the primary development workstation.

Several features of the IIS are still under development. The spreadsheet concept requires interactive responses including interactive formula evaluation. The formula evaluation capabilities are being more efficiently integrated with the Khoros image processing system and other image processing libraries to take advantage of shared memory data-transport mechanisms, parallel algorithm implementation, and distributed processing on supercomputers if necessary. The prototype Dynamic Range browser and Volume browser modules are being modified to conform to the data structure and event processing of the IIS.

A useful feature that needs to be explored is the option to nest spreadsheets so that the contents of a frame are not an image but another image spreadsheet. It is anticipated that future hardware improvements will substantially extend the capabilities of the IISS and drive further development of new visualization techniques and data analysis methods.

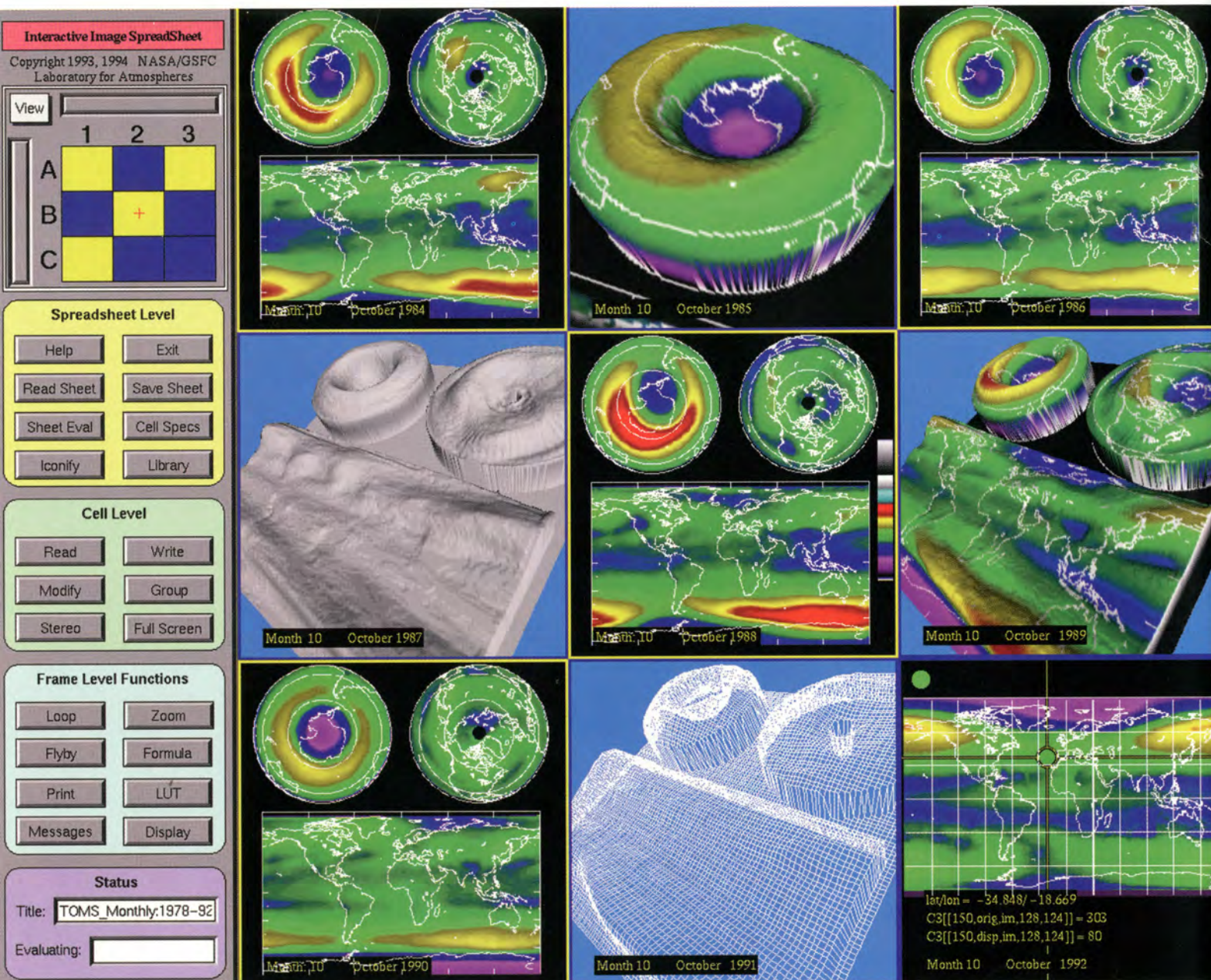
## ACKNOWLEDGMENTS

The authors wish to thank Dennis Chesters for providing the TOMS data, John Barker for the resampled, contrast enhanced Chernobyl Landsat TM image products, and Hal Pierce and Lafayette Long for their assistance in the preparation of graphics. The development of the IISS has been supported by funding from the Energy and Hydrologic Systems Research Program at NASA Headquarters.

## REFERENCES

- W. Hibbard, "A highly parallel approach for satellite archive processing," in *Eighth International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography and Hydrology*, 5–10 Jan. 1992, Atlanta, GA (American Meteorological Society, Boston, MA, 1992), pp. 82–83.
- V. V. Salomonson and J. L. Barker, "EOS Moderate Resolution Imaging Spectroradiometer: Phase C/D status and comments on calibration and georeferencing approaches," in *Guidance and Control 1992*, Proceedings Annual Rocky Mountain Guidance and Control Conference, 8–12 February 1992, Keystone, CO, edited by R. D. Culp and R. P. Zietz (American Astronomical Society, San Diego, CA, 1992), Vol. 78, pp. 665–676.
- D. M. Butler *et al.*, *From Pattern to Process: The Strategy of the Earth Observing System*, EOS Science Steering Committee Report, Vol. II, NASA, p. 27.
- T. V. Papathomas, J. A. Schiavone, and B. Julesz, "Application of computer graphics to the visualization of meteorological data," *Comput. Graph.* **22**(4), 327–334 (1988).
- A. F. Hasler, J. Dodge, and R. H. Woodward, "A high performance interactive image spreadsheet," *Seventh International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography and Hydrology*, 14–18 Jan. 1991, New Orleans, LA (American Meteorological Society, Boston, MA, 1991), pp. 187–194.
- A. F. Hasler, K. Palaniappan, and D. Chesters, "Visualization of Multispectral and Multisource Data Using an Interactive Image Spreadsheet (IISS)," in *Eighth Int. Conf. on Interactive Information and Processing Systems for Meteorology, Oceanography and Hydrology*, 5–10 Jan. 1992, Atlanta, GA (American Meteorological Society, Boston, MA, 1992), pp. 85–92.
- K. Palaniappan, A. F. Hasler, and M. Manyin, "Exploratory analysis of satellite data using the Interactive Image Spreadsheet (IISS) Environment" in *Ninth International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology*, 17–22 Jan. 1993, Anaheim, CA (American Meteorological Society, Boston, MA, 1993), pp. 145–152.
- T. R. Licklider, "Ten years of rows and columns," *Byte* **14**(13), 324–331 (1989).
- W. J. Orvis, *1-2-3 for Scientists and Engineers* (Sybex, Alameda, CA, 1991), 2nd ed.
- C. Lewis, "NoPumpG: Creating interactive graphics with spreadsheet machinery," in *Visual Programming Environments: Paradigms and Systems*, edited by E. P. Glinert (IEEE Computer Society, Los Alamitos, CA, 1990), pp. 526–546.
- S. H. Gutfreund, "ManipIcons in ThinkerToy," *Proceedings OOPSLA*, Orlando, FL (ACM, New York, 1987) pp. 307–317.
- P. T. Nguyen and D. Ho, "Multiple source data processing in remote sensing," in *Digital Imaging Processing in Remote Sensing*, edited by J. P. Muller (Taylor & Francis, London, 1988), pp. 153–176.
- M. H. Overmars, *Forms Library—A Graphical User Interface Toolkit for Silicon Graphics Workstations*, Version 2.1 (Nov. 10, 1992). Dept. of Computer Science, Utrecht University, P. O. Box 80.809, 3508 TB Utrecht, Netherlands. Available by anonymous ftp from sol.cs.ruu.nl (For additional information contact markov@cs.ruu.nl).
- G. Fisher, "An overview of a graphical multilanguage applications environment," *IEEE Trans. Software Eng.* **14**(6) 774–786 (1988).
- J. Rasure and C. Williams, "An integrated data flow visual language and software development environment," *Visual Lang. Comput.* **2**, 217–246 (1991).
- K. Gorey, "IRIS Explorer: The example," *IRIS Universe: The Magazine of Visual Processing*, Issue 17, 34–37 (1991).
- C. Upson, T. Faulhaber, D. Kamins, D. Laidlaw, D. Schlegel, J. Vroom, R. Gurwitz, and A. van Dam, "The Applications Visualization System: A computational environment for scientific visualization," *IEEE Comput. Graph. Appl.* **9**(4), 30–42 (1989).
- W. Hibbard and D. Santeck, "The VIS-5D system for easy interactive visualization," in *Seventh International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography and Hydrology*, 14–18 Jan. 1991, New Orleans, LA (American Meteorological Society, Boston, MA, 1991) pp. 129–134.
- D. Hon "A new method for visualizing data on a sphere," *Comput. Phys.* **5**, 505–513 (1991).
- P. H. Jackson, "IAX-An algebraic image processing language for research," in *Digital Image Processing in Remote Sensing*, edited by J. P. Muller (Taylor & Francis, London, 1988), pp. 79–103.
- Holzmamm, G. J. *Beyond Photography: The Digital Darkroom* (Prentice Hall, Englewood Cliffs, NJ, 1988).
- S. Wolfram, *Mathematica*, 2nd ed. (Addison-Wesley, Redwood City, CA, 1991).
- J. A. Richards, *Remote Sensing Digital Image Analysis* (Springer, Berlin, 1986).
- B. Howard "Analyzing data from all the angles," *PC Magazine*, Oct. 27, 141–158 (1987).
- A. F. Hasler and M. L. des Jardins, "AOIPS/2: An interactive system to analyze and display meteorological datasets for nowcasting," in *Advances in Space Research* (Pergamon, New York, 1987), Vol. 7, No. 11, pp. 375–388.
- R. F. Adler, A. J. Negri, P. R. Keeln, and I. M. Hakkarinen, "Estimation of monthly rainfall over Japan and surrounding waters from a combination of low-orbit microwave and geosynchronous IR data," *J. Appl. Meteor.* **32**(2), 335–356 (1993).
- R. W. Saunders and K. T. Kriebel, "An improved method for detecting clear sky and cloudy radiances from AVHRR data," *Int. J. Remote Sensing* **9**(1), 123–150 (1988).
- A. F. Hasler, J. Strong, R. H. Woodward, and H. Pierce, "Automatic analysis of stereoscopic image pairs for determination of hurricane cloud top height and structure," *J. Appl. Meteor.* **30**, 257–281 (1991).
- D. W. Deering, J. W. Rouse, R. H. Hass, and J. A. Schall, "Measuring forage production of grazing units from Landsat MSS data," *Proceedings of the 10th International Symposium on Remote Sensing of the Environment*, Ann Arbor, MI (Environmental Research Institute of Michigan, Ann Arbor, MI, 1975), pp. 1169–1178.
- W. J. Kaufmann III and L. L. Smarr, *Supercomputing and the Transformation of Science* (Freeman, New York, 1993).
- G. S. Shostak, "The ultimate motion imaging systems," in *Proceedings of the 26th Annual SMPTE Advanced Television and Electronic Imaging Conference*, 7–8 February 1992, San Francisco, CA (Society of Motion Picture and Television Engineers, White Plains, NY, 1992), pp. 7–17.
- V. Cerf, "Data Communications," *IEEE Spectrum* **30**(1), 42 (1993).
- P. Guimaraes and R. McPeters *TOMS Ozone Image Data 1978-1991*, Published on CD-ROM by the Ozone Processing Team, NASA/Goddard Space Flight Center. Available from National Space Science Data Center, NASA/Goddard Space Flight Center, Code 633, Greenbelt, MD 20771.
- P. J. Sellers, "Canopy reflectance photosynthesis and transpiration," *Int. J. Remote Sensing* **6**(8), 1335–1372 (1985).
- A. F. Hasler, H. Pierce, K. R. Morris, and J. Dodge, "Meteorological data fields in perspective," *Bull. Amer. Meteor. Soc.* **66**, 795–801 (1985).
- A. J. Negri, R. F. Adler, and C. D. Kummerow, "False-color display of Special Sensor Microwave/Imager (SSM/I) data," *Bull. Am. Meteor. Soc.* **70**(2), 146–151 (1989).





Enhanced capabilities of the IIS are shown using TOMS monthly average ozone data from 1984 to 1992. Five of the cells contain 2-D imagery using the colormap in Figure 8. The other four cells, A2, B2, B3 and C2, contain 3-D perspective renderings of surfaces representing monthly average ozone levels. Cell C3 shows a CED projection with original and displayed data being probed in latitude and longitude coordinates as well as pixel coordinates.