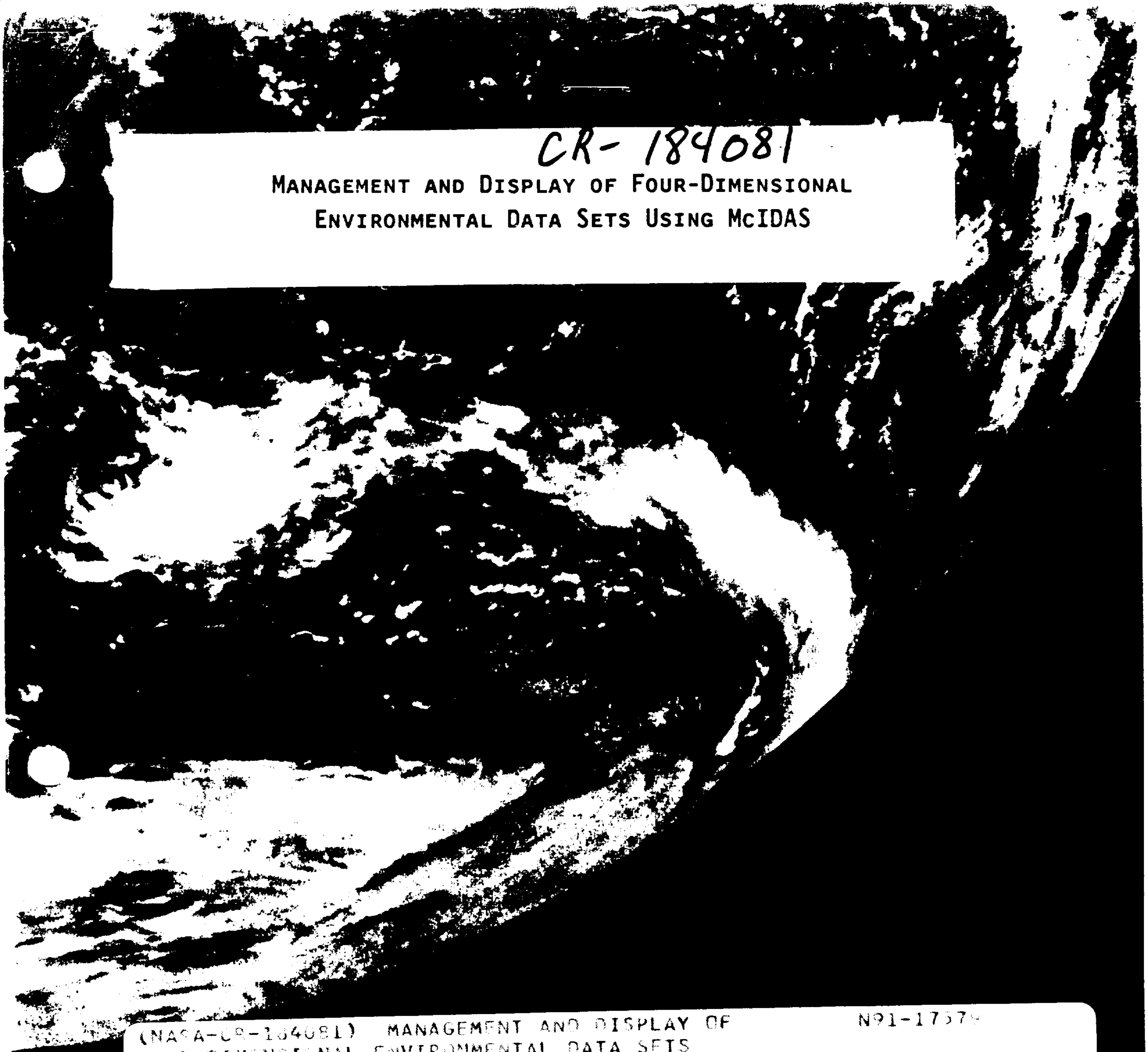
CR- 184081

MANAGEMENT AND DISPLAY OF FOUR-DIMENSIONAL ENVIRONMENTAL DATA SETS USING McIDAS



(NASA-CR-184081) MANAGEMENT AND DISPLAY OF FOUR-DIMENSIONAL ENVIRONMENTAL DATA SETS USING McIDAS Final Report, 26 Feb. 1986 - 31 Aug. 1990 (Wisconsin Univ.) 126 p CSCL 09B

N91-17579

Unclas

G3/61 0325642

A REPORT

from the space science and engineering center
the university of wisconsin-madison
madison, wisconsin

MANAGEMENT AND DISPLAY OF FOUR-DIMENSIONAL ENVIRONMENTAL DATA SETS USING McIDAS

A Final Report to

National Aeronautics and Space Administration (NASA)

prepared for

George C. Marshall Space Flight Center (MSFC)
Marshall Space Flight Center
Alabama 35812

for

Management and Display of Four-Dimensional Environmental Data Sets Using McIDAS

Contract #NAS8-36292

University of Wisconsin Account #144X669

for the period of

2/28/86 thru 8/31/90

submitted by

William L. Hibbard
David Santek
Verner E. Suomi

Space Science and Engineering Center
at the University of Wisconsin - Madison
1225 West Dayton Street
Madison, Wisconsin 53706
(608) 262-0544

November 1990

I. INTRODUCTION

Over the past four years, great strides have been made in the areas of data management and display of 4-D meteorological data sets. When the current contract went into effect in early 1986, the displays consisted of wind trajectories and moisture mesh surfaces from upper air data over a topographical map. Now the software can produce multi-parameter displays of transparent surfaces, trajectories as streamers, density surfaces, 2-D contours and text, in stereo or mono (with color). Also, we have made the move from single image processing on the mainframe computer to real-time interaction on the Stardent computer.

This report summarizes the work accomplished under this contract with details contained in the appendices.

II. WORK SUMMARY

The Statement of Work for this contract read as follows:

TASK 1: Inventory of 4-dimensional data sets. Survey of available and planned 4-D meteorological data sources. Evaluate the data types for their impact on the data management and display system.

TASK 2: Data management. Analyze the requirements for data base management generated by the 4-D data display system. Evaluate the suitability of the existing data base management procedures and file structures in light of the new requirements. Where needed, design and implement new data base management tools and file structures.

TASK 3: Data integration. Assure the quality of the basic 4-D data sets. Investigate interpolation and extrapolation techniques for the 4-D data. Combine 4-D data from various sources to make a uniform and consistent data set for display purposes.

TASK 4: Design data display software to create abstract line graphic 3-D displays. Create realistic shaded 3-D displays. Develop animation routines for these displays in order to produce a dynamic 4-D presentation.

TASK 5: Workstation design. Implement a prototype dynamic color stereo workstation. Produce a complete functional design specification based on interactive studies and user feedback.

TASK 1: 4-D INVENTORY

Inventory of 4-D data sources

Over the course of the contract, many scientists and research institutions were queried about their use of or availability of 4-dimensional data sets. Primarily, the search was limited to meteorological data, though data from other earth sciences have been included. The effort relied on catalogs, journal articles, and referrals to track down the source of a particular data type. A questionnaire was designed to provide a concise format for the inventory: whom to contact, what kind of data is available, how to acquire (price, format), and any reference or catalog that would be useful. Naturally, even the most exhausted searches would not be complete, but also we

excluded individuals that were not willing to provide their data to outside users. Appendix A contains the inventory for the 4-D data sets.

Data sources used

Over the past three years, the 3-D software has been applied to data from 21 different sources. The development of a general 3-D grid structure and display program made this task manageable. Usually, the only new software that needed to be written, were programs to put the data in the McIDAS (such as a tape read program). For imagery, though, it was handled on an individual basis, since most of the effort was in the area of model output. The data types fall into three general categories: imagery, observations, and models.

Imagery

GOES images
McGill volumetric radar
NSSL Doppler radar
NASA/MSFC volumetric radar
LIDAR
CT scan medical data

Observations

Rawinsonde balloon soundings
VAS satellite soundings
Wind Cave data
Global topography
Global soil and vegetation type
HIS aircraft soundings

Models

NSSL kinematic microphysical cloud model
NASA/MSFC LAMPS model
Robert Schlesinger's 3-D cloud model
UW/CIMSS 4-D data assimilation model
NASA/GSFC GMASS model
NMC global model
RAMS model
Warzyn Engineering hydrology model

More information on these data sets can be found in Appendix A and in Hibbard (1989b).

TASK 2: DATA MANAGEMENT

Requirements

The management of 4-D data sets requires more than one data structure for efficient access and organization. The different file structures are divided into categories similar to the way the 21 data sources were divided (under TASK 1):

Images, unevenly spaced data (observations and trajectories), and regularly spaced data (2-D and 3-D grids from model output).

Image data and 2-D grids are similar in structure (2-D matrices), with the differences being the source and volume of data, and the display. Remote sensors (satellite, radar, lidar) produce massive amounts of data spatially, temporally, and spectrally.

Regularly spaced data (grids) are usually produced from numerical models. The output consists of many physical parameters at a sequence of time steps in the form of 2-D and 3-D grids. These data sets may also be large (as is imagery), but the access pattern requires a different structure.

Observed data (rawinsonde and satellite soundings) and trajectories do not fit into a matrix format. No assumptions are made about the space or time distribution of the data.

Evaluation of existing data management

The McIDAS provides three data structures for the storing of weather data: Image, Grid and MD (Meteorological Data) files.

Images are stored as 2-D arrays as 1, 2, or 4 byte quantities per pixel. For multi-spectral data, interleaving of the pixels is done, thus adding a third dimension. Images of different times are kept in individual files. A separate file contains a directory entry for all images on the system so that all users have read access to all images on the system, but not necessarily write access. Since the naming convention is numeric, a time sequence of images would normally be stored in numeric order. This structure is very general, and is being used for the radar, lidar, and the resulting rendered 3-D graphics.

The grid file structure provides for only 2-D grids. There are limitations in size, 10240 maximum number of grid points per grid, and 159 grids per grid file. Also, the data values at the grid points are stored as scaled integers. Since the naming of data sets on McIDAS is currently numeric, large numbers of 2-D grids are stored in successive grid files.

The MD file structure is a key driven system for randomly spaced data. The layout is a matrix, with each cell containing the data at a particular time and location. It is usually organized so that each row corresponds to a time, and each column a location. This works well for surface hourly reports and upper air data. The data structure is organized well, but may be inefficient in some access modes. Trajectories could be stored in MD files, but since trajectories are distributed more non-uniformly, performance and storage would be degraded.

New data structure and McIDAS commands

The current McIDAS data structures did not provide an efficient means for storing and manipulating 3-D grids and trajectories. The 2-D grid structure was used as a model to develop a 3-D grid structure. The naming of utility commands for accessing and listing the data are similar to their 2-D counterparts. But there are some important differences between the two. 3-D grids are stored as floating point numbers and there is no size limitation. The maximum number of grids per grid file was increased to 319 grids; but for large 3-D grids (40,000 points), only about 200 grids can

be stored due to limitations in the McIDAS file system. Some of these changes have brought about discussions regarding changing the 2-D structure.

A new type of file was designed for trajectories. They are stored as lists of X, Y, Z, T coordinates with a maximum of 1000 trajectories or 30,000 coordinates. Currently, there is one trajectory file that can store up to 10 trajectory sets. This has not been a problem since trajectories are usually a derived parameter and can be regenerated when needed from the original data of U, V, and W 3-D grids.

Along with the new data structures, new commands for McIDAS were written. A concise and general package of programs were written to manage and display 3-D grids and trajectories. A summary of these can be found in Appendix C.

Management of 4-D and 5-D data sets

Organization of multi-parameter time sequences of 3-D grids is necessary for not only the user, but also for the software. To generate long time animation sequences, the grids should be ordered such that this task can proceed somewhat automatically. We order the grids by grouping all parameters for each time, with an integral number of groups per grid file if the data set spans several files.

TASK 3: DATA INTEGRATION

Data quality assurance

Most of the data we have worked with are processed in some way. For the modelers, any anomalies that would appear in the 3-D images would be useful in detecting problems in the model. We have used data from Doppler radar, which is processed before we have access. The problem of data quality for real-time rawinsondes was not looked into as we forged ahead into investigating better graphical techniques and animation.

Interpolation and extrapolation

Originally, our major source of 4-D data was rawinsondes or VAS satellite soundings, which is very coarse in the time dimension. Interpolation becomes important to generate smooth time animations of the data. As we gathered more data sets, model output became the dominant source, and the need for interpolation went away. In cases where interpolation was necessary, a simple linear method was used to double the amount of times for a more coherent animation. For image data, the problem is much more difficult and was not addressed.

Combine data from different sources

The combination of data from different sources has not been a problem. Most data sets (especially model output) are highly integrated already. Once the data are in a 3-D grid or trajectory file, though, combinations of different data are possible.

TASK 4: 4-D DISPLAY SOFTWARE

Design display software

The display software evolved based on the feedback we received from meteorologists. It was developed on an IBM mainframe computer which allowed us to animate up to 128 frames of individually generated 3-D images. Many of these features were ported to the Stardent which permits real-time interaction with the data.

A rectangular box is the basis for the display, to which we can add objects and line drawings. The base of the box depicts geography: base maps and topographical relief. Color is used to delimit ocean from land and also to accent the land relief. This orientates the user in terms of earth location. For small scale studies (e.g., thunderstorm), this will be a shaded plane. The vertical axes of the box are labeled with height (in kilometers).

The depiction of 3-D fields can be added to this initial box. Scalar parameters can be represented with transparent (or opaque) contour surfaces, grid-mesh surfaces, or as a transparent fog. Wind speed maxima shown as contour surfaces represent a "jet-core" well. A time animation of potential temperature depicted as a grid-mesh may show wave features propagating. Cloud water rendered as a transparent fog gives an effect of cloud density. Though they are all scalar quantities, different display techniques are used to bring out the desired information and also to allow combinations of different parameters. 2-D slices through the 3-D scalars can be drawn either at a constant height level or along the topographical surface.

The motion of air particles are shown as line segments. These trajectories can be drawn in several ways. They can be long and tapered showing a long time span in a single image for tracing particles back to some beginning point. Usually they are drawn shorter, with length proportional to speed, and animated. These can be solid or fade to transparent.

Image data can be textured mapped to a 3-D perspective. This has been used for lidar data and for GOES satellite imagery.

3-D display techniques

The use of stereo gives a vivid 3-D effect for viewing the images generated on McIDAS. Unfortunately, stereo viewing is not practical for videotaping or for inclusion in publications. Techniques were developed that would heighten the 3-Dimensionality of the images. One of these is depth cued trajectories. The trajectories are shaded so that the ones closer are brighter. Tapered trajectories (the leading edge is wider than the tail) have been used in some cases. For animated sequences, a slight rocking motion is added. By rocking the objects a degree or so (excluding the outer box), the eye detects easily what is forward of another object. We found that rocking along the axis of motion (horizontal for motion going horizontal, and along the vertical for up/down moving objects) proved the best.

TASK 5: WORKSTATION

The McIDAS workstation was the starting point for a display device for the 4-D weather data. A stereo terminal was developed using two large screen projectors, which presented different images for the left and right eye through polarization. This

required modifications to the existing McIDAS terminal which is described in Hibbard, et al., 1987. This was an effective tool for presenting animated color sequences of 4-D data in stereo; but, it could not be viewed elsewhere through still pictures, or more importantly, video tape. This required more development in the software to provide more visual cues (see 3-D Display Techniques above) and to rely less on stereo.

The ultimate goal was to develop an interactive 3-D terminal (Hibbard 1988, Hibbard and Santek, 1989b, 1989c). This was developed on a Stellar (now Stardent) GS-1000 graphics supercomputer. This system can produce 3-D animations in real time, of large gridded data sets. Through the use of a mouse, the scientist can view his data by selecting parameters, controlling animation, and manipulating the viewing angle. The user is presented with a window that contains his data rendered as 3-D objects and a control panel. The window contains imagery in the same format as described in Design Display Software. But, the image contained in the window can be rotated, zoomed, panned by holding down a mouse button and moving the mouse. The control panel is new, but very intuitive. By clicking buttons (with the mouse) parameters can be selected (turned on and off) and animation is started and stopped. Sliders provide a way to select levels of a particular parameter. For example, to change from a 50 m/s contour surface to 55 m/s level, the pointer is positioned over the appropriate slider and is moved with the mouse button depressed until 55 m/s is displayed above the slider. Color and transparency for the surfaces are also controlled in this way.

The interactive workstation has many advantages over the original workstation. Stereo, or some software techniques (such as rocking), are not needed to provide visual cues to heighten the 3-D effect: the user now controls the perspective at will. The time needed to produce a video tape has been reduced from weeks to just minutes. The user can interactively change the perspective or the combination of parameters immediately, without having to wait for a sequence of images to be generated and displayed.

III. PUBLIC APPEARANCES

The product, namely the display of 4-D data sets, has been well received wherever it has been shown. This includes classroom situations, international conferences, and many other places unknown to us, due to the massive distribution of video taped displays of animated 4-D images. "Live" demonstrations were an important part in presenting our work as it would provide us with immediate feedback of comments and suggestions. The demonstrations in the exhibit area during the International Conferences on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology would draw hundreds of meteorologists. We also had video tapes accepted at SIGGraph (Special Interest Group on Graphics) conferences where upwards of 20,000 attend (Hibbard and Santek, 1988, 1989d).

Our first large public production was at the 3rd IIPS held in New Orleans, LA in January, 1987. We brought along two McIDAS workstations and Aquastar IIIC projectors to present 3-D animated sequences in polarized stereo. Ten different sequences were shown using data from radar (standard and Doppler), rawinsondes, satellite, and numerical models (thunderstorm, regional, and global). The response was positive, with many favorable comments and suggestions resulting. Many commented that they remembered some meteorology by viewing this demo. Some suggested the addition of line graphics to the images. This forum also sparked the interest of

researchers with data. This gave us an opportunity to work with various types of data for the years to come, which aided in the design of data management and display software.

The following year at the 4th IIPS, a ten minute video tape was continually shown, with someone present for explanation. Some suggestions from the previous conference were incorporated and new data sets were presented. Of note is the use of more color and depth cues (to eliminate the need for binocular stereo) and the application of the software to dual Doppler radar and more numerical models (LAMPS and GMASS).

At the 5th and 6th IIPS (held in 1989 and 1990), we presented a truly interactive 3-D display with a Stellar (Stardent) Graphics Supercomputer. At times we had the researcher presenting his own data, describing how it helped interpret the model output. At the 1990 conference we displayed the current NMC global model output in addition to case studies of very large data sets.

Also at the IIPS conferences, one or two papers were presented each year describing the current effort and future plans. Other conferences where video tapes produced by us were presented are:

- SPIE: Boston, MA 1988
- 4th IIPS: Anaheim, CA 1988 (Prof. Patricia Pauley)
- Polar Low Conference: Madison, WI 1988 (Louis Uccellini)
- Palmen Symposium: Helsinki, Finland 1988 (Uccellini)
- SIGGRAPH: Atlanta, GA 1988
- SIGGRAPH: Boston, MA 1989

During the fall of 1987, the Synoptic Laboratory class spent some time on McIDAS studying the 3-D representation of their case study of an extratropical cyclone. The class, taught by Profs. Lyle Horn and Patricia Pauley, were given a series of questions to answer based on the 3-D images. The response was favorable, noting that the 3-D images provided additional information that could not be extracted from traditional plots. Many faculty of the Meteorology Department also viewed the display and were interested in developing case studies for their courses. In the fall of 1989, the Synoptic Laboratory class revisited, but this time it was presented on the Stardent. Once again it proved valuable to the class, especially in showing the inter-relationships between parameters.

IV. FUTURE EFFORT

The future development effort will expand on the work accomplished over the last three years. Three main areas emerge as being important for continued work in this area:

- Data management
- Visualizing model data
- Visualizing remotely sensed data

In the area of data management, a more object orientated approach will be investigated. This will tie together large and varied data sets in a more manageable way. The user accesses the data through these named objects, rather than grid

numbers or area numbers. This allows the data to be ordered in a fashion that is commensurate with the desired display.

Visualizing model data was emphasized in much of the work that was done. A general display package will be developed that will allow ANY scientist to import his 4-D data and display on a Stardent workstation. User defined procedures will be added to the analysis, to permit complex calculations to operate on the data at display time. Also, real-time interaction with parameters as the model executes is not too far away into the future as the speed of supercomputers increases.

Visualizing remotely sensed data is becoming increasingly important. Here the volume of data is much larger than that of model data. Interactive tools are needed to manage, display, and operate on the data. Also, methods for combining with model output are needed for intercomparisons of parameters.

V. APPENDICES

Appendix A.	4-D Data Inventory (separately bound)
Appendix B.	Journal and Conference Papers
Appendix C.	New McIDAS Commands

jws/c:\4d\final.doc

APPENDIX B

The following is a list of papers written under this contract. Copies of the papers follow.

Hibbard, W., 1986a: 4-D Display of Meteorological Data. Proceedings, 1986 Workshop on Interactive 3D Graphics, Chapel Hill, SIGGRAPH, 26-33.

Hibbard, W., 1986b: Computer Generated Imagery for 4-D Meteorological Data. Bull. Amer. Met. Soc., 67, 1362-1369.

Hibbard, W., R. Kraus, D. Santek, and J.T. Young, 1987: 4-D Display of Weather Data on McIDAS. Proceedings, Third International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, New Orleans, AMS, 89-93.

Hibbard, W., 1987: 4-D Display of Satellite Cloud Images. Proceedings, Digital Image Processing and Visual Communications Technologies in Meteorology, Cambridge, SPIE, 83-85.

Santek, D., L. Leslie, B. Goodman, G. Diak, and G. Callan, 1987: 4-D Techniques for Evaluation of Atmospheric Model Forecasts. Proceedings, Digital Image Processing and Visual Communications Technologies in Meteorology, Cambridge, SPIE, 75-77.

Hibbard, W., 1988: A Next Generation McIDAS Workstation. Proceedings, Fourth International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Anaheim, CA, AMS, 57-61.

Hibbard, W. and D. Santek, 1988a: Presidents' Day Storm, Visualization/State of the Art: Update. SIGGraph Video Rev., No. 35.

Hibbard, W. and D. Santek, 1988b: Visualizing Weather Data. Workshop on Graphics in Meteorology, ECMWF, Reading, England, 63-65.

Hibbard, W., D. Santek, and G. Dengel, 1988: Visualization of Four-dimensional Meteorological Data. SIGGraph Video Review, No. 37.

Santek, D., W. Hibbard, K. Quinn, and T. Melka, 1989: 4-D Display of Geohydrological Model Results. Proceedings, Fifth Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Anaheim, CA, AMS.

Hibbard, W. and D. Santek, 1989a: Visualizing Large Data Sets. Proceedings, Fifth Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Anaheim, CA, AMS.

Hibbard, W. and D. Santek, 1989b: Visualizing Large Data Sets in the Earth Sciences. Computer, Vol. 22, No. 8, 53-57.

Hibbard, W. and D. Santek, 1989c: Interactivity is the Key. Chapel Hill Workshop on Volume Visualization, University of North Carolina, Chapel Hill, NC, 39-43.

Hibbard, W. and D. Santek, 1989d: Interactive Earth Science Visualization. SIGGraph Video Rev., No. 43.

Hibbard, W. and D. Santek, 1989e: Visualization of Earth Science Data. Volume Visualization/State of the Art. SIGGraph Video Rev., No. 44.

Hibbard, W. and D. Santek, 1989f: 4D model of Presidents' Day Storm, 4D model of cloud water density, and 4D model of sea level pressure contours. Planetscapes (interactive video disk), Optical Data Corporation, Warren, NJ.

Pauley, P., L. Keller, W. Hibbard, and D. Santek, 1989: 3-D Animations of a Developing Extratropical Cyclone for use in General Meteorological Education. 2nd International Conference on School and Popular Meteorology and Oceanographic Education, AMS, 166-167.

Tripoli, G., W. Hibbard, D. Santek, 1989: Four-Dimensional Interactive Analysis: A Tool for the Efficient Understanding of Large Data Sets. Preprints, 12th Conference on Weather analysis and Forecasting. Monterey, CA, AMS, J10-J12.

Hibbard, W. L. Uccellini, D. Santek, K. Brill, 1989: Application of the 4-D McIDAS to a Model Diagnostic Study of the Presidents' Day Cyclone. Bull. Amer. Met. Soc., 70, 1394-1403.

Meyer, P.J. and M. Seablom, 1989: Applications of the Four-Dimensional McIDAS to LAMPS Model Output. Presented at the Fifth Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Anaheim, CA, AMS. (Presentation but no paper)

Pauley, P., W. Hibbard, and D. Santek, 1989a: A Four-Dimensional Case Study for Synoptic Laboratory. Workshop on Synoptic Meteorology Instruction. National Center for Atmospheric Research, Boulder, CO, 122-132.

Pauley, P. M., W. Hibbard and D. Santek, 1989b: The Use of 4-D Graphics in Teaching Synoptic Meteorology. Proceedings, Fifth Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Anaheim, CA, AMS.

Hibbard, W., D. Santek, and M-F Voidrot-Martinez, 1989: The Interactive 4-D McIDAS Workstation. Second Workshop on Meteorological Operational Systems, ECMWF, Reading, England.

Hibbard, W., D. Santek, M-F Voidrot-Martinez, D. Kamins and J. Vroom, 1990: UNIX and X Windows: The Right Choice for Interactive Systems. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Anaheim, CA.

Santek, D., W. Hibbard, M-F Voidrot-Martinez, D. Kamins and J. Vroom, 1990: A UNIX and X Windows Implementation of McIDAS. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Anaheim, CA.

Hibbard, W. and D. Santek, 1990: The VIS-5D for Easy Interactive Visualization. Proceedings of the First IEEE Conference on Visualization, p. 28-35.

Proceedings, 1986 Workshop on Interactive 3D Graphics, Chapel Hill, SIGGRAPH

4-D Display of Meteorological Data

William L. Hibbard

Space Science and Engineering Center
University of Wisconsin-Madison

Abstract

The Man-computer Interactive Data Access System (McIDAS) developed at the University of Wisconsin-Madison Space Science and Engineering Center (UW-SSEC) collects large quantities of meteorological data in real time for storage, analysis and display on multi-frame video terminals. Software is being developed on the McIDAS system which produces 3-D images from a variety of meteorological data for stereo display in short animation sequences. These animation sequences are produced in a few minutes. The user controls the space and time extents, contents and information density of the display.

1. Introduction

The atmosphere is a multivariate moving 3-D volume. Traditionally, meteorologists have worked with large numbers of 2-D plots of different parameters, at different times, levels and orientations within the atmosphere, and integrated these into a mental model of the atmosphere. Recently there has been interest by a number of groups to produce integrated 4-D displays of meteorological data, including the National Center for Atmospheric Research (Grotjahn and Chervin, 1984), the Goddard Space Flight Center (Hasler, Pierce, Morris and Dodge, 1985), Colorado State University (Meade, 1985), AT&T Bell Laboratories (Schiavone et. al.,1986), Lawrence Livermore National Laboratory (Grotch, 1985), and the University of Wisconsin Space Science and Engineering Center (Hibbard, Krauss and Young, 1985).

The McIDAS system has been developed over the last 12 years at the UW-SSEC to give meteorologists access to a variety of weather data in real time through multi-frame video terminals (Chatters and Suomi, 1975). This system consists of a host, which is an IBM mainframe, from one to twenty or more video terminals designed and built at UW-SSEC, interfaces to a variety of meteorological data sources, and roughly 500,000 lines of software, mostly FORTRAN. Large quatities of weather data from observing systems such as satellites, weather balloons, radar and ground observations, and from weather models, are brought into the McIDAS host within minutes of being generated. The data are stored and analyzed on the host, and transmitted to the terminal for display. The terminal can display animated sequences of images and graphics.

This paper describes an effort to develop four-dimensional displays of meteorological data with the McIDAS system at the UW-SSEC. This effort has focused on the creation of software which analyzes meteorological data and generates time sequences of 3-D images depicting the atmosphere. The images are usually generated in pairs for both left and right eye perspective and displayed on a McIDAS terminal with anaglyphic stereo and

ORIGINAL PAGE IS
OF POOR QUALITY

animation. The stereo display has been implemented by coloring the left and right eye images green and red, and viewing through green and red filter glasses. We are currently building a large screen display using two projectors with polarized filters, to be viewed through polarized filter glasses. The image generation software uses perspective, shading, hidden surface removal, transparency and other visual cues consistently with the stereo geometry to produce a vivid illusion of a moving 3-D model of the atmosphere. A typical sequence contains from 3 to 7 image pairs and is animated repeatedly at between 1 and 15 frames per second.

2. Image Design

The atmosphere is very complex. It is understood in terms of multiple interacting variables over a space-time continuum. Weather observing systems and weather models produce huge quantities of data, which are increasing every year. Forecasters, model developers and students all have a need to access this data and comprehend it. They need to see meteorological data in ways that make the 3-D structures and motions clear for a variety of different parameters. Our images are being designed to accomplish these goals.

Our images are also designed to match the capabilities of our existing McIDAS terminals. These terminals include up to 128 video frames and 64 graphics overlay frames, all of 480 by 640 pixels. The video pixels have 6 bits of intensity encoded in 3 bits by a statistical scheme which is suited to smoothly varying intensities. The video frames are organized in pairs, with the two 6 bit intensities driving a color table which maps the 12 video bits to 5 bits each of R, G and B out. The video output is overlayed by a graphics frame with 3 bits per pixel. The 3 graphic bits are decoded as a transparent level and a table of 7 colors defined by 5 bits each of R, G and B. A cursor of selectable shape, size and color is drawn over the video and graphics to help the user interact with displayed information. User input comes via keyboard, joysticks, mouse, and graphics tablet. A modified version of the color table circuit is being built for the polarized two projector display.

Figures 1 through 7 (1 and 3-7 are from Hibbard, 1986) illustrate the images generated by our software. Figure 7 consists of four small stereographs. To see the stereo effect, look at the left image with your left eye and the right image with your right eye. It may help to place a piece of cardboard perpendicular to the page separating the images.

All of our images depict a rectangular box of atmosphere, with the box edges drawn in and a map on the bottom of the box, including boundaries or topography or both. The box helps create the illusion of 3-D space and gives a position reference for items in the box. The tick marks on the back of the box are labelled in kilometers above sea level. The topographical map is useful not only to show the relation between weather and topography, but also to avoid depicting weather phenomena beneath ground level (in figure 4, some of the phenomena are actually underground). The resolution of our topography dataset is about 10 miles, so topography is generated only for large scale maps.

A primary goal of our effort is to show the 3-D structure of the atmosphere. This problem is attacked through the consistent use of stereo display, perspective, shading, hidden surface removal and other depth cues. Since the atmosphere is a continuum, meteorologists are interested in information at all depths of the display, without nearer items obscuring farther ones. We address this problem through the use of transparent and mesh surfaces and groups of small objects, such as the wind streamlines and trajectories in figures 4 and 5. These techniques are effective if their densities are tuned correctly. For

ORIGINAL PAGE IS
OF POOR QUALITY

example, the shapes of the transparent surfaces in figure 3 are reasonably clear, but this breaks down when there are 3 or more transparent layers over an opaque layer. The software which generates the streamlines and trajectories tries to tune their densities so that they are evenly distributed and not so dense as to obscure each other. The trajectories are denser near the ground level where lower wind speeds make them shorter.

Because illumination, surface shape and transparency all contribute to perceived brightness, care must be taken so their effects are not visually ambiguous. For example, the orientations of the opaque streamlines in figure 4 are easier to see than the orientations of the semi-transparent trajectories in figure 5, where the transparency tends to wash out the subtle brightness cues to orientation.

Atmospheric data exist in the form of scalars (temperature, pressure and moisture), vectors (winds), discrete data (lightning and precipitation) and images (satellite and radar). Display techniques must be tuned to the mathematical form of the data, as well as its complexity and texture. Figures 1 to 7 illustrate the use of opaque surfaces, transparent surfaces, grid mesh surfaces, line graphics on surfaces, streamlines, trajectories, clouds, density slices and drop shadows. The contour surfaces can be applied to a wide variety of scalar quantities including wind speed, moisture, temperature, wind vorticity, etc. We have found that appropriate 3-D graphics for a parameter are not necessarily similar to the standard 2-D graphics for that parameter. For example, pressure contour lines in 2-D are familiar to anyone who watches TV weather shows, but pressure contour surfaces in 3-D are nearly flat horizontal surfaces and not very interesting. However, contour surfaces of the ratio between observed pressure and standard atmosphere pressure bring out vertical detail of high and low pressure areas. The density of objects in the display should be tuned not only for visual clarity, but also to reflect the complexity of the data being displayed. For example, the density of wind trajectories should be increased (and their thickness reduced) for more complex wind fields. For most shaded objects diffuse reflection works well, but the cloud images need a special shading function suited to cloud-like texture.

Texture makes the stereo more effective by creating tie points for the viewer's eyes to line up the left and right images. The natural texture of the topography and clouds helps to resolve their stereo depths. When contour surfaces are very flat they work poorly with the stereo.

The movement of the atmosphere is important. We depict motion through time sequences of images which are animated repeatedly. In order to produce the sequence in a reasonable amount of computing time, it is desirable to use a short sequence of images. The time steps represented in the animation will be minutes or hours, and the display rate should be between 1 and 15 frames per second. In order that the motion be unambiguous, objects in the images should significantly overlap from one frame to the next. For example, the lengths of the trajectories in figures 5 and 7 are tuned so that they overlap between frames. The way the tails of the trajectories fade into transparency also helps to enhance their sense of motion. When contour lines or surfaces are animated, the distance a contour line or surface moves between frames should be less than the distance between contours within a frame. This makes the motion less ambiguous.

The input to our image generation software consists of large quantities of meteorological data, which may contain errors, gaps in coverage and bizarre geometries. In order to be credible with the user, the software must be robust enough to avoid producing crazy images in response to these data pathologies. This is a particular

ORIGINAL PAGE IS
OF POOR QUALITY

problem for the cloud image generation illustrated in figure 1, which uses pattern recognition to extract clouds from satellite imagery. The contour surface generation software used for figures 2 and 3 is also vulnerable to problems with complex input data.

Large amounts of data can cause clutter and confusion between different parameters in an image. However, images which ordinarily look cluttered will clear up when viewed with stereo and animation. The differences in depth and motion help to resolve different items in the display. Variations in texture, brightness and color also tend to resolve clutter (Marr, 1982). The eye is very good at distinguishing objects which differ in size, orientation, shape or brightness. For example, in figure 5 the trajectories are 20 percent brighter than the map and the grid mesh is brighter still. Also in figure 5, the bright grid mesh lines contrast well with the dark map lines.

The time and distance scales of our displays are selected by the user. If the scales do not match, the motions may be too slow or too fast or the angles may be misleading. In almost all cases, the vertical scale must be greatly exaggerated, because of the relative thinness of the atmosphere. This exaggeration makes it possible to see vertical detail but leads to unrealistic angles and shapes. For example, the large clouds in figure 1 would be very flat viewed from space. The fact that they look cloud-like is satisfying but misleading, and apparently due to a fractal scaling law by which cloud geometry is similar at different scales if the aspect ratio is changed with size (Lovejoy and Schertzer, 1986).

3. Image Generation

Our software generates images of the types shown in figures 1 through 7 in accordance with the image design ideas presented above. The software is meant to be efficient, in order to be interactive, while producing images of sufficient quality to present the weather information clearly.

Weather image generation usually consists of three basic steps: analyzing the weather data, rendering the map, and rendering the weather data over the map. The analysis step is separate since its results may be used for images at different perspectives, times, densities, etc. The map rendering step is separate since a map background can be used for several images in an animation sequence.

The analysis step varies widely, depending on the nature of the weather data being displayed. For unevenly distributed observations, it is usually necessary to interpolate these to a uniform grid. This is done using an efficient statistical technique (Hibbard and Wiley, 1985). For wind streamlines and trajectories, such as shown in figures 4 and 5, paths are traced through a 3-D or 4-D grid of wind vectors. For cloud images, such as shown in figure 1, cloud heights are derived from cloud infrared temperatures, and cloud edges are detected with a pattern recognition technique. For contour surfaces, such as shown in figures 2 and 3, equal level surfaces are traced through a 3-D grid.

The topographical and contour surfaces shown in figures 2 and 3 are Gouraud shaded triangles. The topographical map usually consists of 5,000 to 10,000 triangles. This density works well with the texture of the land and the coastlines, and preserves reasonable computing times. The contour surfaces are composed of hundreds or thousands of triangles, depending on the area of the surface. These triangles are larger than those used for the topography, generally matching the density of the weather data sets. The use of larger triangles decreases the time and memory needed for generating the surfaces, but occasionally produces odd

looking surfaces.

The wind streamlines and trajectories are shaded tubes. These are rendered as triangles which extend across the width of the tube and are shaded with a "cylindrical" shade function. This shade function approximates diffuse reflection on the surface of a cylinder using forward differences for scan conversion of the cylinder. The function is parameterized to work with arbitrarily oriented cylinder and light source directions. For the trajectories which fade to transparency, this function is modified to also interpolate the alpha value.

The shaded clouds of figure 1 are generated from GOES (Geostationary Operational Environmental Satellite) visible and infrared imagery, by remapping into perspective in latitude, longitude and altitude coordinates. A simplified version of the remapping algorithm is given in the appendix. The software which generated figure 1 combines this algorithm with a cloud edge detecting algorithm. The software also optimizes many of the calculations through the use of forward differences and special cases. The remap algorithm is much more efficient than the technique of rendering every square of 4 pixels in the GOES image as a polygon in the perspective image. The remap algorithm can currently only accommodate a limited range of perspective rotations, because of the assumption that a GOES image line maps into a series of line segments running roughly horizontally across the perspective image. However, this could be remedied by adding an initial step to rotate the GOES image, or by using a more complex interpolation method in the second phase of the algorithm.

The horizontal slices of radar echos in figure 6 are generated by a remapping technique. Radar data are analyzed to images at a series of levels in the atmosphere, where pixel value is simply echo intensity. These images are then remapped into a perspective image. The remapping of these radar slices is much easier than the cloud remap. Each pixel of the perspective image maps, via simple perspective transforms, to a pixel in each horizontal slice image. At each pixel of the perspective image, we examine the mapped pixel from each horizontal slice and pick the nearest non-clear one, which is used to generate the shade of the pixel in the perspective image. Drop shadows from the lowest level are generated by a similar technique. The perspective transform for the lowest level is altered to assume an altitude of zero, and non-clear pixels cause the map surface to be dimmed.

Our software is organized to render into a band of lines, so that in general an entire triangle can be rendered at once, saving on the overhead of managing polygons. The number of lines in a band can be adjusted, but is usually set at 80. The shade at each pixel is a simple 8 bit intensity. When the image is loaded into a 6 bit video frame, the low 2 bits are dropped. A Z-buffer algorithm is used for hidden surface removal, with only 8 bits of depth. So far this low amount of Z resolution has not caused serious problems, partially because we scale depths to use all 256 levels. A Z-buffer algorithm is also used for rendering semi-transparent objects. This is done with two images, one for opaque objects and one for semi-transparent objects. The opaque image has a shade and a depth at each pixel, and the transparent image has a shade, a depth, and an 8 bit alpha at each pixel. First, all opaque objects, including the map, are rendered into the opaque image. Then, the semi-transparent objects are rendered into the transparent image, with a modified depth comparison. A new transparent pixel is combined with the transparent image only if its depth is less than the Z-buffer depth of the opaque image. The new pixel is combined according to the usual rules for alpha and shade, and the transparent depth becomes the minimum of the old transparent depth and the new pixel depth. This

always produces correct shade for pixels which see two or less transparent layers plus an opaque layer, and produces correct shade with probability 2/3 for pixels which see three transparent layers plus an opaque layer. However, with three, four, or more transparent layers, the effects of the layers on the shade are so visually ambiguous that errors in shade calculations are not readily apparent.

Our image generation software has no anti-aliasing. However, for objects shaded by diffuse reflection, the effects of aliasing can be reduced by placing the light source in the same direction as the viewer. This causes the edges of shaded objects, where the aliasing is most noticeable, to be relatively dim.

These image generation techniques are implemented on an IBM 4381. The computing times vary depending on the data set being rendered, but some typical CPU times can be stated. The map in figure 1 is rendered in 7.5 seconds, and the map in figure 3 is rendered in 11 seconds. The steamlines and grid mesh in figure 4 are analyzed in 15 seconds and rendered in 5 seconds. The surfaces in figure 3 are analyzed in 6 seconds and rendered in 19 seconds. The surface in figure 2 is rendered in 11 seconds. The clouds in figure 1 are analyzed and rendered in 22 seconds. Figure 5 is part of an animation sequence of 7 stereo pairs. The analysis for the sequence, which processes 5 data sets spanning 9 hours, takes 70 seconds. The trajectories and grid mesh surface are rendered in 8 seconds per image. The radar slices in figure 6 are rendered in 21 seconds. Of course, an animation sequence of stereo image pairs for any of these image types will require one to several minutes for analysis and rendering.

4. Interactivity

Interactivity is the key to useful meteorological display. The user should have control over the contents, spatial and time extents, perspective point and information density of the display. The nature of interactivity depends on the system's response time to the user's controls. The faster the response is, the more the user can experiment and play with the data.

Meteorologists use graphical displays to support decisions, some of them important. Thus, they must have faith in the accuracy of the displays. However, meteorologists tend to not trust depth information in 3-D displays. The work described in this paper is partially motivated by a desire to increase the accuracy of perceived depth information in meteorological displays. Ultimately, the best way to increase the user's trust in depth information is to provide control over the perspective point of the display with fast response time. Depth information in an 3-D image is converted to more trustworthy screen geometry information when the Z-axis is rotated into the screen plane. Hardware rendering systems are being built which can solve this problem for images of reasonable complexity (Clark, 1982 and Fuchs et. al., 1985).

For meteorologists, the issues of trust and interaction are complex. Some meteorologists reject computer analysis and graphics of their data, on the grounds that they can only understand their data by studying it closely enough to draw the graphics by hand. Other meteorologists see this as impractical because of the huge quantities of weather data becoming available. The point is that the first group wants the interaction with their data which they get from drawing graphics by hand. The synthesis of these two positions will be computer analysis and graphics which to allow meteorologists to interact with their data with the same response time as a pencil.

Very fast rendering speeds should be exploited to give the user control over a variety of parameters of the display. Multiple contour surfaces in an image, such as shown in figure 3, are badly cluttered if more than 2 or 3 different contour levels are rendered together. However, the relation between many different contour levels could be seen by changing the value of a single contour surface quickly under user control. Similarly, the relation between many different types of meteorological data could be seen by selectively adding and deleting renderings of each type of data, so they can be seen in various combinations. If all of them must be displayed at once, the resulting image would be cluttered. Interactive control over the distance and time scales of the display would allow the user to take a close look at important small scale regions of a large scale display. This requires more than just an image zoom, since the density of the displayed information must be adjusted with the scale change. Of course, fast response for each of these forms of control requires special attention to the analysis and rendering algorithms which need to be accelerated.

We have found that it is important that interactivity be limited to aspects of the display worth controlling. Otherwise, the user will be overwhelmed with choices they do not understand. For example, the parameters of the shading function should generally be fixed for a given type of image. Similarly, control over the perspective view point is useful, but other parameters of the perspective transform should be hidden from the user. Default values should be supplied for most controls, so that the user can choose to ignore them.

The user can interact with the 2-D images on the McIDAS system through the use of joysticks and cursor. We plan to extend this to our 3-D images through the implementation of a 3-D cursor. This cursor should be a movable image element, such as a cross hair, which participates in the perspective and hidden surface removal with the base image. The 3-D cursor will allow the user to retrieve the values of meteorological parameters at points in the atmosphere. It will also allow the user to interact with the image rendering process, for example to direct the software to trace the path in the display of a particular air parcel. We may also implement a drop shadow of the cursor to the ground, as an aid to finding the earth location of upper air phenomena.

5. Conclusions

Meteorologists need effective access to the large amounts of data produced by weather observing systems and by numerical weather models. Computer graphics provides a natural tool for this access, particularly in three dimensions with motion. The work described here explores some of the possibilities of this tool. However, there is much greater potential for computer graphics of weather data and other forms of scientific data. During the next ten years, reasonably priced fast rendering systems will change the way in which scientists see their data.

There are a variety of ways that our displays can be improved. It would be useful to combine more types of data in a single display. The cloud images can be improved by finding sources of information about the sides and bottoms of the clouds. In fact, there is some hope that sophisticated pattern recognition algorithms can deduce some of this information from the texture of the cloud tops. An important area for improvement lies in making it easier for the user to manage and analyze the data underlying the display, particularly as data from diverse sources must be combined to understand a weather situation.

Our 4-D display effort initially included an attempt to use

commercially available 3-D rendering software. However, we had better results with our own software, largely because of the flexibility it provided. The clouds in figure 1 and the trajectories in figure 5 would be difficult to produce efficiently with commercial software packages. It is a lot of work to produce rendering software, but it is worth the effort in terms of the control and understanding one gains. Although we are very excited about the development of fast rendering hardware, our experience with commercial software packages suggests that new hardware be viewed with caution. Still, the benefits of fast interaction will justify designing images to match the capabilities of rendering hardware.

Appendix

This algorithm remaps a GOES image to a perspective image, with two phases of processing for each GOES image line. First, the line is mapped to a series of line segments, assumed to stretch roughly horizontally across the perspective image. Then, the pixels between this series and the series of line segments mapped from the previous GOES line are shaded by vertical interpolation.

Constants and variables:

nglines,ngelems - size of GOES image
nplines,npelems - size of perspective image
gline,gelem - pixel coordinates in the GOES image
lat,lon,alt - 3-D location of pixel from GOES image
qelem - element coordinate in perspective image of pixel from GOES image
pline,pelem - pixel coordinates in the perspective image
GOES-IR(gline,gelem) - infrared value at coordinates gline,gelem in GOES image
pshade(pline,pelem) - shade value at coordinates pline,pelem in perspective image
iline(pelem) - line coordinate in perspective image as a function of element coordinate in perspective image, along current remapped GOES image line
ishade(pelem) - shade as a function of element coordinate in perspective image, along current remapped GOES image line
jline(pelem) - similar to iline, for previous GOES image line
jshade(pelem) - similar to ishade, for previous GOES image line

Functions:

transform - transform from GOES image coordinates to latitude and longitude. This function is calculated precisely at a square grid of points and interpolated inside the squares
height - calculates altitude from GOES infrared temperature
shade - calculates shade based on GOES visible value and the geometry of the cloud top surface
perspective - calculates the perspective transform from latitude, longitude and altitude to line and element

Algorithm:

```
for gline=1 to nglines
  lelem=0
```

```
for pelem=1 to npelems
  iline(pelem)=0
  jline(pelem)=0
endfor
 ; first phase: map a line of the GOES image to a series of line
 ;  segments in the perspective image
for gelem=1 to ngelems
  lat,lon=transform(gline,gelem)
  alt=height(GOES-IR(gline,gelem))
  gshade=shade(gline,gelem)
  pline,qelem=perspective(lat,lon,alt)
  if 0<lelem and lelem<qelem then
     ; create a line segment in the perspective image between the
     ; destinations of two consecutive pixels from the GOES image
    dline=(pline-lline)/(qelem-lelem)
    dshade=(gshade-lshade)/(qelem-lelem)
    for pelem=lelem+1 to qelem
      lline=lline+dline
      lshade=lshade+dshade
      if 1<=pelem and pelem<=npelems then
        iline(pelem)=lline
        ishade(pelem)=lshade
      endif
    endfor
  endif
  lline=pline
  lshade=gshade
  lelem=qelem
endfor
 ; second phase: calculate the shades of pixels in the perspective
 ; image between the current and the previous series of line segments
for pelem=1 to npelems
  aline=iline(pelem)
  bline=jline(pelem)
  if 0<bline and bline<aline then
     ; interpolate shades along a vertical segment of pixels in
     ; the perspective image
    ashade=ishade(pelem)
    bshade=jshade(pelem)
    dshade=(ashade-bshade)/(aline-bline)
    for pline=bline+1 to aline
      bshade=bshade+dshade
      if 1<=pline and pline<=nplines then
        pshade(pline,pelem)=bshade
      endif
    endfor
  endif
endfor
 ; move line segments for current line to previous line
for pelem=1 to npelems
  jline(pelem)=iline(pelem)
  jshade(pelem)=ishade(pelem)
endfor
endfor
```

ACKNOWLEDGEMENTS

I wish to thank D. Santek for the software which renders the

contour lines in figure 2, and for help with the surface contour software. I wish to thank E. Brandes of the National Severe Storms Laboratory for dual doppler radar data, and A. Bellon of McGill University for volumetric radar data. I also wish to thank F. Mosher, J. T. Young, R. Krauss, and T. Whittaker for help and support. This work was supported by the National Aeronautics and Space Administration (NAS8-33799 and NAS8-36292).

REFERENCES

Chatters, G. and V. Suomi, 1975: The application of McIDAS. IEEE Trans. Geosci. Electron., GE-13, 137-146.

Clark, J., 1982: The geometry engine: a VSLI geometry system for graphics. Computer Graphics 16(3), 127-133.

Fuchs, H., J. Goldfeather, J. Hultquist, S. Spach, J. Austin, F. Brooks, J. Eyles, and J. Poulton, 1985: Fast spheres, textures, transparencies, and image enhancements in pixel planes. Computer Graphics 19(3), 111-120.

Grotch, L., 1985: The use of interactive three-dimensional graphics in a climate research program. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Los Angeles, Amer. Meteor. Soc., 213-215.

Grotjahn, R. and R. Chervin, 1984: Animated graphics in meteorological research and presentations. Bull. Amer. Meteor. Soc. 65, 1201-1208.

Hasler, A., H. Pierce, K. Morris, and J. Dodge, 1985: Meteorological data fields "in perspective". Bull. Amer. Meteor. Soc. 66, 795-801.

Hibbard, W., R. Krauss, and J. Young, 1985: 3-D weather displays using McIDAS. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Los Angeles, Amer. Meteor. Soc., 153-156.

Hibbard, W. and D. Wiley, 1985: An efficient method of interpolating observations to uniformly spaced grids. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Los Angeles, Amer. Meteor. Soc., 144-147.

Hibbard, W., 1986: Computer generated imagery for 4-D meteorological data. Bull. Amer. Meteor. Accepted for publication.

Lovejoy, S. and D. Schertzer, 1986: Scale invariance, symmetries, fractals, and stochastic simulations of atmospheric phenomena. Bull. Amer. Meteor. Soc. 67, 21-32.

Marr, D., 1982: Vision. W. H. Freeman and Company. 397 pages.

Meade, A., 1985: Interactive display of clouds, terrain and other surfaces. M. S. Thesis, Dept. of Atmospheric Science, Colorado State University.

Schiavone, J., T. Papathomas, B. Julesz, C. Kreitzberg, and D. Perkey, 1986: Anaglyphic stereo animation of meteorological fields. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Los Angeles, Amer. Meteor. Soc., 64-71.

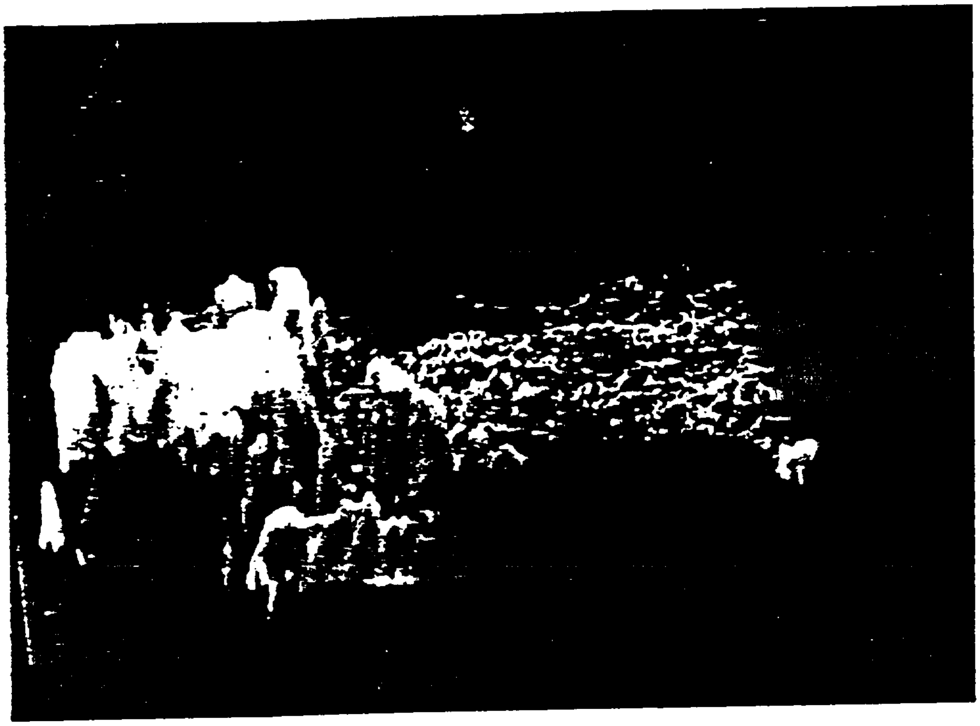


Figure 1. Cloud image generated from GOES data over the Gulf of Mexico, Louisiana, Mississippi and Alabama. The tick marks on the back of the box are labelled in kilometers above sea level.

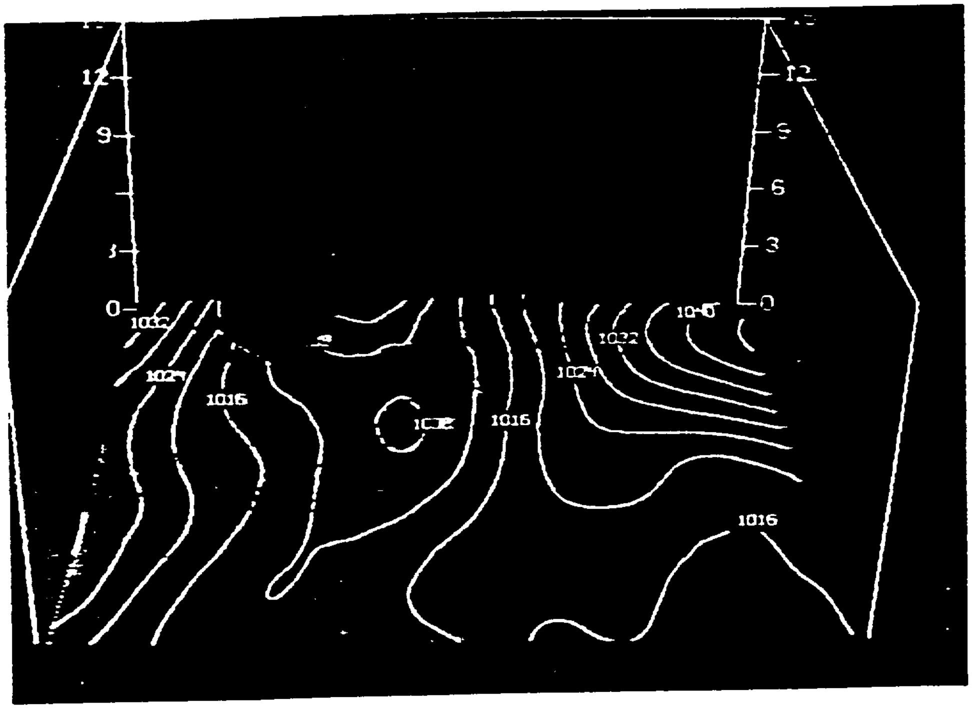


Figure 2. 55 m/s wind speed surface and sea level pressure contour lines drawn on the topographical surface, generated from LFM (Limited Fine Mesh) model data.

Figure 3. Semi-transparent 32 m/s wind speed surface and an opaque 48 m/s wind speed surface generated from balloon data.

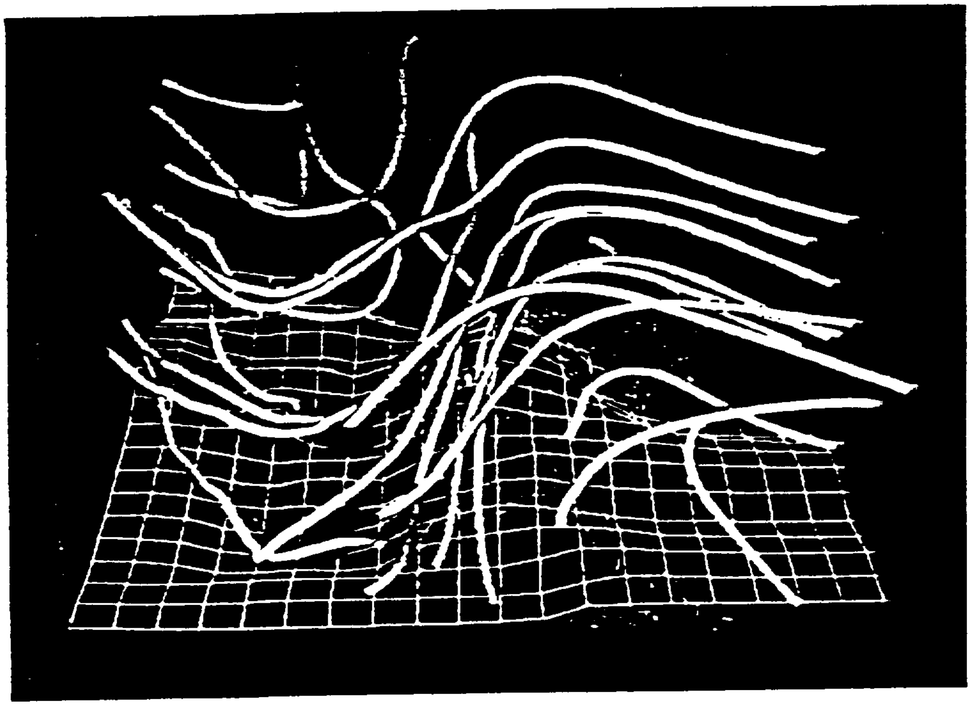


Figure 4. Wind streamlines and a 5 g/kg (grams of water vapor per kilogram of air) grid mesh mixing ratio surface, generated from balloon data.

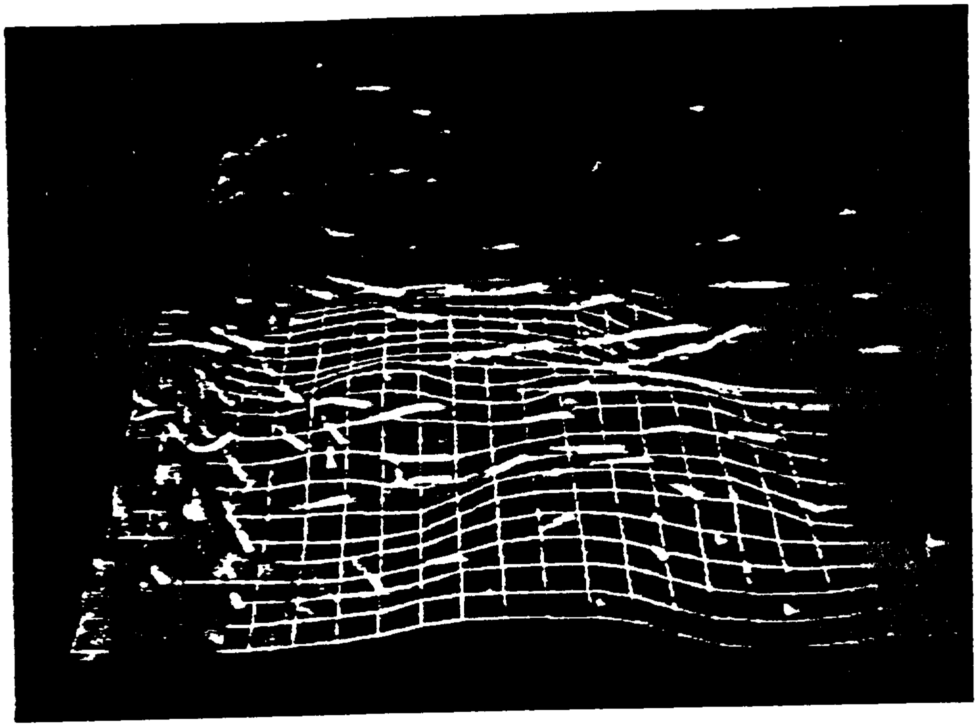
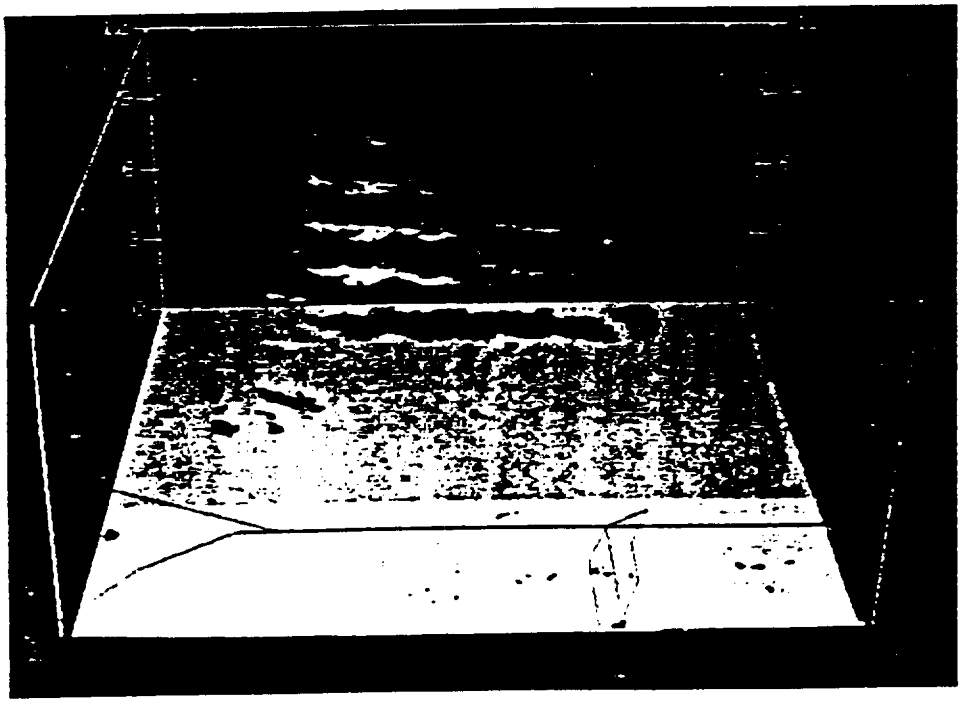


Figure 5. Wind trajectories with tails fading to transparency, and a 5 g/kg mixing ratio surface, generated from VAS (VISSR Atmospheric Sounder) satellite data.



ORIGINAL PAGE IS
OF POOR QUALITY

Figure 6. Horizontal slices of echos from a volumetric radar over a small scale map centered at Mount Royal, Quebec.

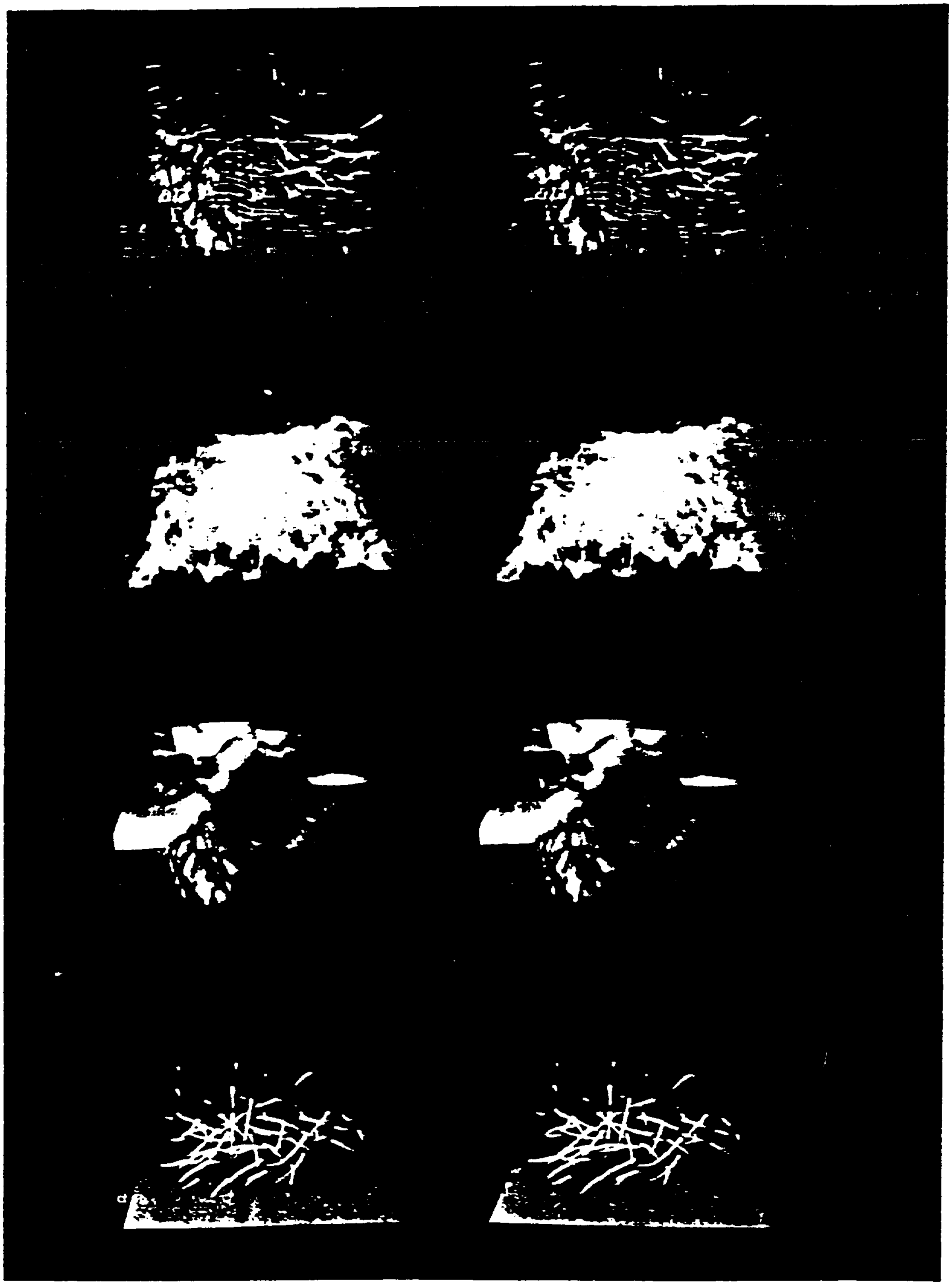
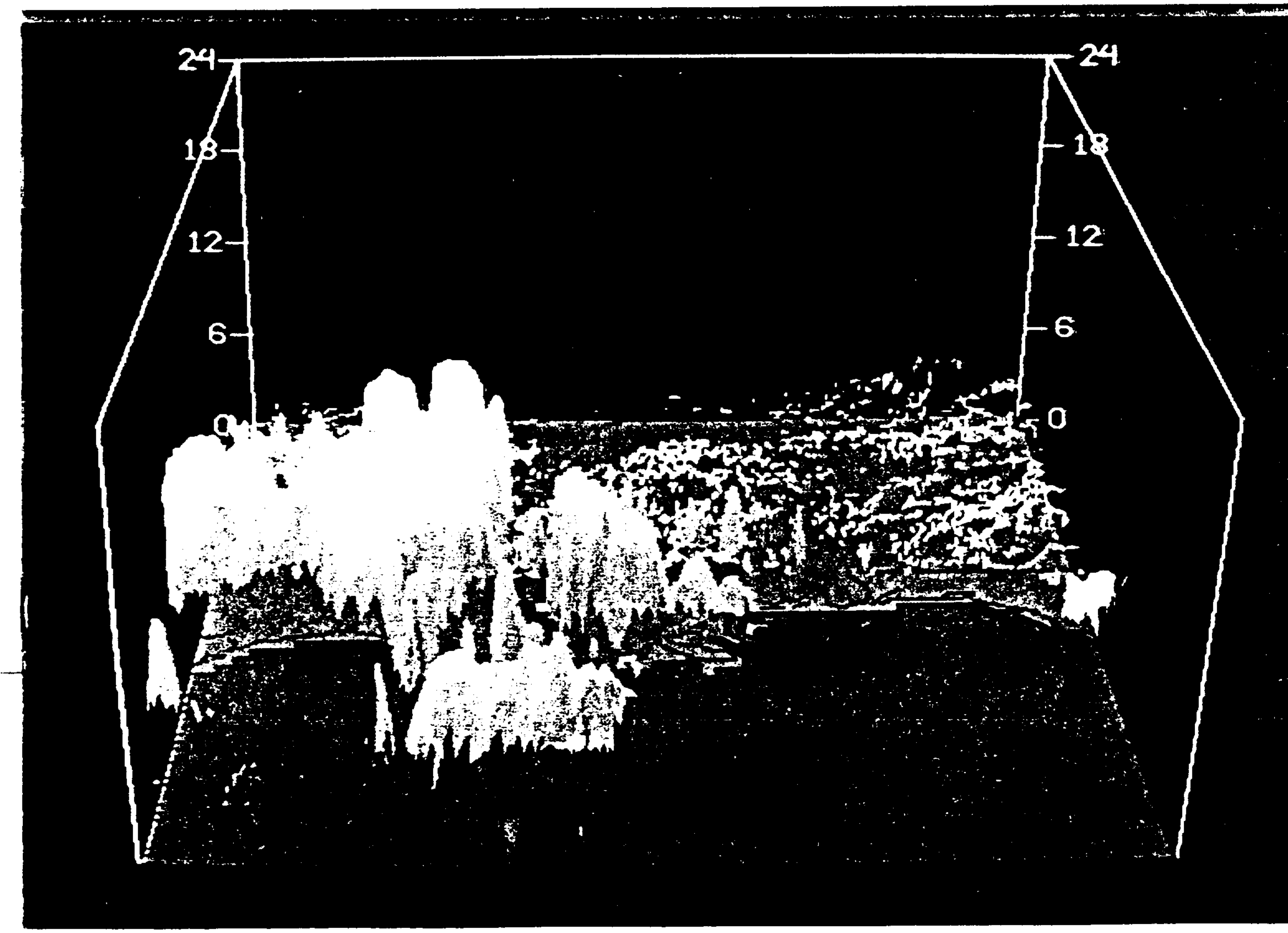


Figure 7. Four small stereographs of: (a) wind trajectories and mixing ratio surface from balloon data, (b) clouds from GOES data, (c) opaque wind speed surface from balloon data, and (d) wind trajectories in a severe thunderstorm from dual doppler radar data over a small scale map near Orienta, Oklahoma. To see the stereo it may help to place a piece of cardboard perpendicular to the page separating the images.

Bulletin of the American Meteorological Society

Volume 67 Number 11 November 1986

ORIGINAL PAGE IS
OF POOR QUALITY



Cover: Cloud image generated from GOES data of 1800 GMT 10 September 1985 over the Gulf of Mexico, Louisiana, Mississippi, and Alabama. The image was generated by remap with artificial enhancement of the sun's shade and a cloud-edge-detection algorithm. For more details see the article by Hibbard beginning on page 1362.

[Reprinted from BULLETIN OF THE AMERICAN METEOROLOGICAL SOCIETY, Vol. 67, No. 11, November 1986]
Printed in U. S. A.

Computer-Generated Imagery for 4-D Meteorological Data

William L. Hibbard
Space Science and Engineering Center
University of Wisconsin-Madison

Abstract

The University of Wisconsin-Madison Space Science and Engineering Center is developing animated stereo display terminals for use with McIDAS (Man-computer Interactive Data Access System). This paper describes image-generation techniques which have been developed to take maximum advantage of these terminals, integrating large quantities of four-dimensional meteorological data from balloon and satellite soundings, satellite images, doppler and volumetric radar, and conventional surface observations. The images have been designed to use perspective, shading, hidden-surface removal, and transparency to augment the animation and stereo-display geometry. They create an illusion of a moving three-dimensional model of the atmosphere so vivid that you feel like you can reach into the display and touch it. This paper describes the design of these images and a number of rules of thumb for generating four-dimensional meteorological displays.

1. Introduction

A meteorologist wishing to understand a weather situation must typically study a large number of map plots of various types of weather data over a range of times and vertical levels. Skilled meteorologists can integrate these plots and create moving three-dimensional pictures of the weather in their minds. It would be useful to meteorologists to have a tool that would directly present integrated three-dimensional moving displays of weather situations.

There has been interest by a number of groups recently in the development of three-dimensional weather displays, including the National Center for Atmospheric Research (Grotjahn and Chervin, 1984), Colorado State University (Meade, 1985), the NASA Goddard Space Flight Center (Hasler et al., 1985), AT&T Bell Labs (Schiavone et al., 1986), Lawrence Livermore National Laboratory (Grotch, 1985), and the University of Wisconsin-Madison, Space Science and Engineering Center (Hibbard et al., 1985).

The stereo McIDAS terminal being developed at the University of Wisconsin-Madison Space Science and Engineering Center is an important tool for creating moving three-dimensional weather displays. This terminal displays different images to the viewer's left and right eyes by using filter glasses, which may either be red and green or cross polarized. The viewer has the illusion of looking at a three-dimensional object. The stereo terminal can also animate sequences of stereo images to create the effect of motion.

In order to use the stereo terminal effectively, the displayed images should depict the moving three-dimensional structure of atmospheric parameters and phenomena as clearly as possible. This paper reports on imagery designed and implemented over the last three years as part of the larger stereo-terminal-development effort. In these images the three dimensions are the three spatial dimensions with animation used for the time dimension. The attitude is that the viewer is looking at a model of the atmosphere evolving over time. The real content of this work is the development of a set of algorithms that create these images interactively (in real time) from data on the McIDAS computer system. The paper starts with a description of the goals and problems of three-dimensional meteorological displays, followed by the general approach used and rules of thumb learned along the way. Then the design of specific images is described. These include cloud images from GOES (Geostationary Operational Environmental Satellite) data, wind streamlines from rawinsondes, wind trajectories from rawinsondes and VAS (VISSR [visible infrared spin scan radiometer] Atmospheric Sounder) soundings, contour surfaces from rawinsondes, wind trajectories from dual doppler radar, and images of multiple sections through a volumetric radar. These descriptions do not include the details of the geometry and shading algorithms used. Such information can be found in graphics texts (Foley and Van Dam, 1982 and Rogers, 1985). The conclusion describes future work and the application of these techniques to new sources of data.

2. Goals and problems of 4-D meteorological display

Description of the atmosphere requires presentation of parameters within its volume, and representation of their time evolution. To depict the three-dimensional spatial portion of this, it is necessary to use a variety of visual cues including perspective, stereo, motion parallax, depth precedence, shading, texture, brightness, color, shadows, and transparency. Perspective is reflected in the way things appear smaller at greater distances; stereo is the use of different views for the right and left eyes; motion parallax is the way the relative positions of objects at different depths appear to change as the viewer moves; depth precedence is simply the fact that near objects hide objects behind them (this is often called hidden-surface removal); and shading is the variation in brightness due to the changing surface orientation of the objects being viewed. Texture is a high-frequency variation in shading, which can provide a lot of information about depth and surface properties. Brightness and color also tend to carry depth information. Bright objects usually appear closer than dull objects, and more subtly, red objects sometimes appear closer than blue objects. Shadows and transparency also help to support the illusion of depth. These visual cues are powerful tools for communicating depth information when they are used together consistently. However, it is important to in-

ORIGINAL PAGE IS
OF POOR QUALITY

tegrate the use of different visual cues so that their effects are not ambiguous. For example, the perceived brightness of a scene is a function of illumination, shape, transparency, and shadows. If these cues are used carefully, the viewer will be able to separate their effects. Because the atmosphere is a continuum, an effective display must show information at all depths without nearer regions obscuring farther regions. Whereas most computer-graphics applications to date have been concerned with the surfaces of three-dimensional objects, for meteorology the concern is with the three-dimensional volume.

Meteorological data exist in the form of scalar fields (moisture, temperature, pressure, wind speed, etc.), vector fields (winds), images (satellite and radar images) and discrete phenomena (lightning and precipitation). Display techniques must be developed for each type of data, suited to the mathematical form as well as the density and texture of the data. It is also desirable to combine different types of data into a single display without confusion between parameters or too much clutter.

The motion of the atmosphere is depicted by animating a sequence of images. In an animated display it is desirable that the motion be clearly apparent using a relatively short sequence of images. A short sequence will conserve display-terminal frame space and can also be computed more quickly.

As with two-dimensional meteorological displays, interactivity is important. The viewer should be able to control the contents and the viewpoint of the display and to generate a new display relatively quickly. Because of the complexity of three-dimensional display algorithms, particularly for sequences of images, efficiency must be seriously addressed to achieve the desired level of interactivity.

Three-dimensional displays share a number of data-oriented problems with more-traditional displays. Bad data points can generate distracting and bizarre effects in an analysis. In fact, the greater volume of data used in generating three-dimensional displays makes them more susceptible to bad-data problems. Data density and coverage should help determine the scale of the display. However variations in density and coverage can create problems. Gaps in the data can create misleading displays while too much data can create a cluttered display.

A particular problem of three-dimensional displays is the exaggerated vertical-to-horizontal aspect ratio that they often have. For a horizontal scale of hundreds or thousands of kilometers, it is necessary to expand the vertical scale by a factor of tens or hundreds. Although this is unavoidable, it can lead to misleading angles and slopes in the display.

Another particular problem of three-dimensional display is the difficulty of presenting precise quantifiable information. This is in part due to the ambiguity of depth, which makes it easier to see relative spatial relations than absolute location.

3. Design of 4-D meteorological imagery

All of the images described here (except the first cloud image) are based inside a rectangular box whose edges are drawn in the image. This helps to define the three-dimensional space and to create a position reference for objects in the box. Th coordinates of the box are usually latitude, longitude, an height. Tick marks along two of the box's vertical edges are labeled in kilometers to help locate objects vertically. Some sort of map base is always drawn on the bottom of the box. This can be a topographical relief surface or a flat surface. Map boundaries may be drawn on the surface as a way to find the horizontal locations of objects in the box. The relief map shows the relation of weather to topography. It also hides the part of the box space which is below ground level and which should not appear to contain atmosphere.

The principal depth cues used are perspective, stereo, depth precedence (hidden-surface removal), and shading. When these are used accurately and consistently they can create a striking three-dimensional illusion. Although stereo is central to the four-dimensional weather-display effort, its effect is greatly enhanced by the use of perspective, shading, and depth precedence. Stereo is also enhanced by texture or other variations in brightness that create tie points for the eyes to line up the left and right images. Other depth cues used include motion parallax, brightness, shadows, and transparency. In order to create a convincing three-dimensional illusion, the depth cues should be used consistently.

It is important that the images contain no gross errors that can distract the viewer's attention from the real information of the display. On the other hand, there are some calculations that can be very approximate without causing a distraction. For example, smooth surfaces can be approximated by fairly large flat polygons without this being noticed as long as th shading is smooth.

Because the atmosphere is a continuum and most of its phenomena are continuous, it is desirable to display information from many depths simultaneously without nearer items obscuring farther items. This can be accomplished with the judicious use of transparent surfaces and grid-mesh surfaces. A group of small shaded objects, such as trajectories or streamlines, is also effective if their density is picked correctly. However, these effects should not be overworked to the point where the images become cluttered and confusing.

Images which appear cluttered when viewed statically without stereo may become clear when animation and stereo are added, as differences in depth and motion tend to resolve items in the image. Clutter can also be cleared if different items are given different sizes, textures, brightnesses, or colors (Marr, 1982). For example, sets of objects can be resolved if the general brightness level of one set is at least twenty percent different than the other.

In addition, animated items in the display should have significant overlap from one frame to the next; otherwise, they will be difficult to follow. For example, trajectories should be long enough to overlap between frames.

Although interactivity is important, it is also important not to burden the user with too many choices of perspective geometry, light-source placement, shading algorithm parameters, etc. These can be picked correctly once (perhaps as a function of the data to be displayed), freeing the user's attention for meteorology. For example, fixing the light sour to coincide with the viewer's eye position seems to give goo results in all situations. The user should have control over the viewpoint of the display, so that a scene may be viewed from

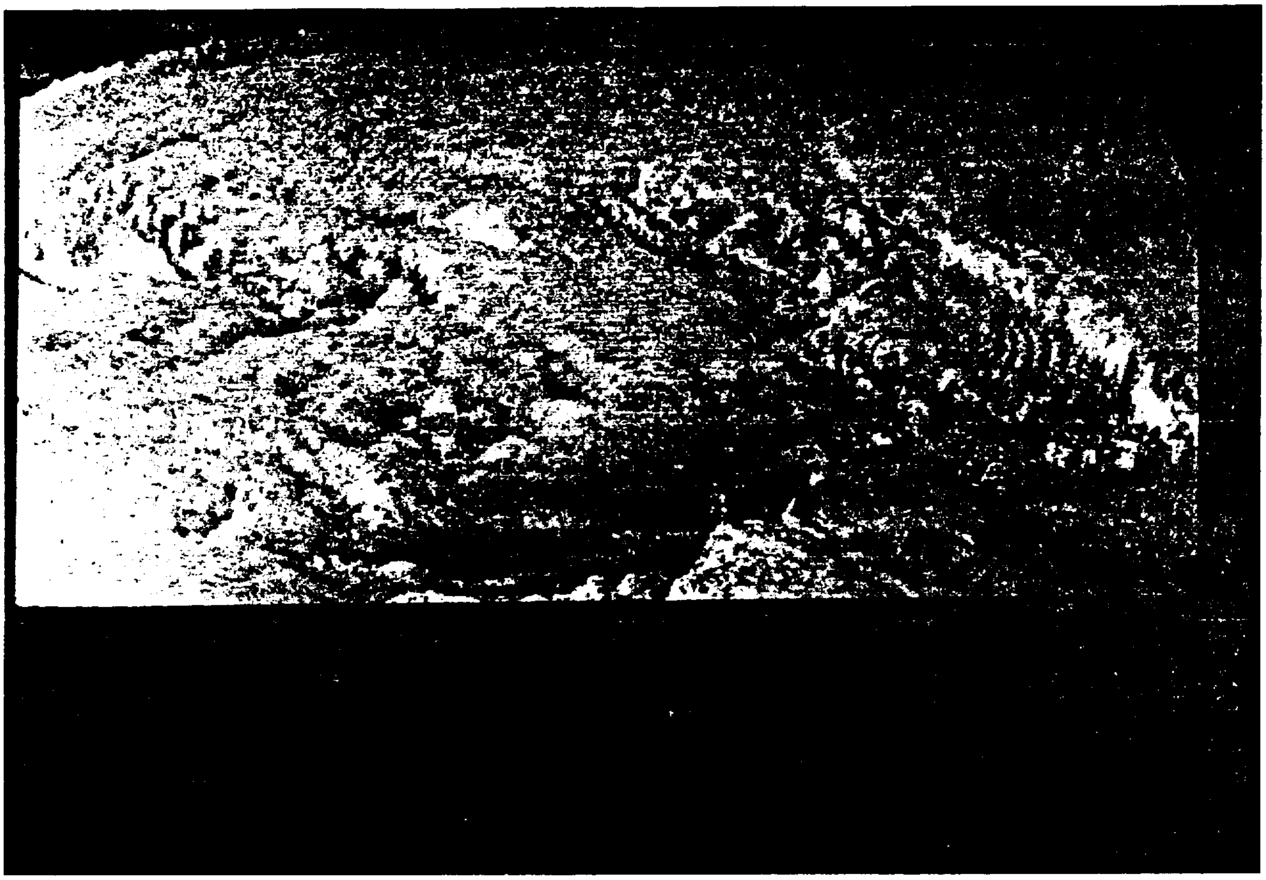


FIG. 1. Cloud image generated from GOES data of 2215 GMT 28 March 1984, by remap and vertical shift of pixels according to infrared-temperature-derived heights.

different angles and different distances. The user should also be able to have some control over the contents of a display, such as the time and geographical extents, the values of contour surfaces, and in some cases the density of information. All these forms of control are possible with the displays presented here. Interactivity is greatly enhanced by increasing the efficiency of the display generation, giving the user faster response to his choices.

4. Cloud images

This work includes two approaches to creating perspective images of clouds from GOES data. The first approach remaps GOES images to a nongeosynchronous perspective and then shifts pixels vertically in the image by a distance proportional to height, as calculated from IR (infrared) temperature (Fig. 1). Any gaps created by these vertical shifts are filled in by repeating pixel values. For high-resolution images, IR brightnesses are interpolated.

The images produced by this approach appear three-dimensional and are effective when viewed in stereo with animation. However, they suffer from several problems. The shading is generated from the GOES visible brightness and lacks texture except for low-sun angles. Where there are vertical gaps between cloud and ground and between cloud layers this approach interpolates over the gaps, connecting clouds with ground and cloud layers with each other.

The second approach to cloud images is an attempt to overcome the problems of the first. This approach produces perspective images of clouds over a topographical map (Fig. 2), which may be viewed in stereo with animation. A remapping algorithm is combined with artificial enhancement of the sun's shading and a pattern recognition algorithm to detect cloud edges. The cloud-edge detection is based primarily on the IR-brightness-variance technique of Coakley and Bretherton (1982) with surface hourly temperatures and visible brightness used to increase the accuracy of detecting low clouds. Detecting cloud edges makes it possible to treat clouds as distinct objects in three-dimensional space, so that they may be combined with other types of objects in the display.

One curious aspect of 3-D cloud displays is that they look so much like we expect clouds to look, despite the fact that they cover areas hundreds of kilometers across and the vertical scale is greatly exaggerated. Apparently, clouds obey a fractal scaling law with similar shapes at different scales if the vertical-to-horizontal aspect ratio is adjusted to match the scale (Lovejoy and Schertzer, 1986).

The beautiful cloud images produced at the Goddard Space Flight Center helped to motivate and guide this work (Hasler et al., 1985).

5. Wind trajectories and streamlines

Time sequences of stereo images of isentropic wind trajectories are generated from rawinsondes or VAS soundings combined with a grid-mesh mixing-ratio surface and a topographical map (Fig. 3). The trajectories are depicted as shaded tubes of finite diameter. With animation they appear to trace a path through the atmosphere, with their tail ends fading into transparency as they move. The finite body of the trajectories makes it possible to use shade and hidden-surface removal as cues to their shape and location. The fading into transparency enhances the sense of motion and serves as a cue to wind speed, as the degree of transparency along a trajectory is proportional to the length of time since the air parcel was at the points along the trajectory. The display clearly shows the relation between wind, moisture, and topography over time.

The analysis is relatively simple. For each observation time over the period of the display three-dimensional isentropic grids of wind velocity and height are produced. Sounding data is interpolated along each sounding to 10 evenly spaced potential-temperature levels. On each level wind speed and height are analyzed to a uniform 20 by 20 grid using a fast Barnes analysis (Hibbard and Wiley, 1985). The wind trajectories are traced along the isentropic surfaces using linear interpolation in time and space. The algorithm tries to maintain a reasonable density of trajectories. The mixing-ratio surface at each time is analyzed by finding the height of the desired mixing-ratio surface at each sounding, and interpolating these heights to the same grid. Thus the mixing-ratio surface consists of points in three dimensions all having the same mixing-ratio value, similarly to the way an isentropic surface consists of points with the same potential temperature.

The first images produced as part of this work were wind streamlines drawn as shaded tubes (they look like spaghetti) combined with a grid-mesh mixing-ratio surface and a simple map (Fig. 4). These are generated from rawinsonde balloon soundings with an analysis similar to the trajectories described above. Animation is used for motion parallax rather than time sequence. Two stereo pairs of streamline images are generated, so that by animating between the two pairs, the scene seems to rock vertically. This is very effective in enhancing the three-dimensional illusion.

6. Contour surfaces

Perspective line graphics of isentropic and mixing-ratio surfaces are generated from rawinsondes. These are grid-mesh surfaces over map outlines (Fig. 5). Because of their simplicity, long animation sequences can be generated quickly. They work best when the time interval between images is three hours and when they are animated at a rate of at least three

FIG. 2. Cloud image over a topographical map, generated from GOES data of 1800 GMT 10 September 1985 by remap with artificial enhancement of sun's shade and cloud-edge-detection algorithm.

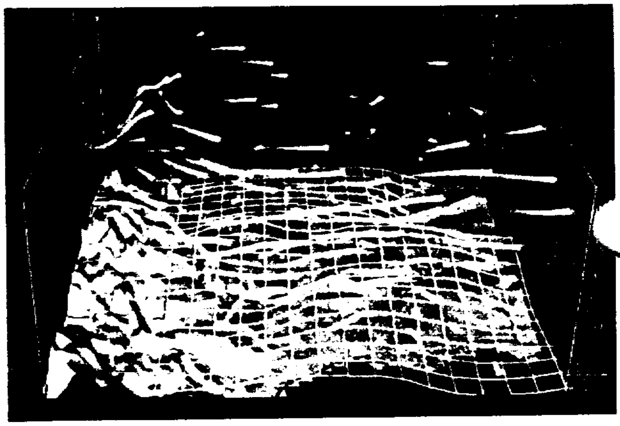


FIG. 3. Wind trajectories and a 5 g/kg mixing-ratio surface over a topographical map, generated from VAS soundings of 11 April 1984.

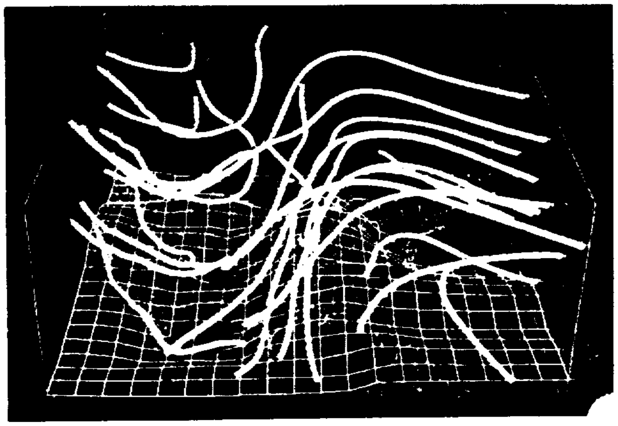


FIG. 4. Wind streamlines and a 5 g/kg mixing-ratio surface over a simple map, generated from rawinsondes of 1200 GMT 11 April 1979.

ORIGINAL PAGE IS
OF POOR QUALITY

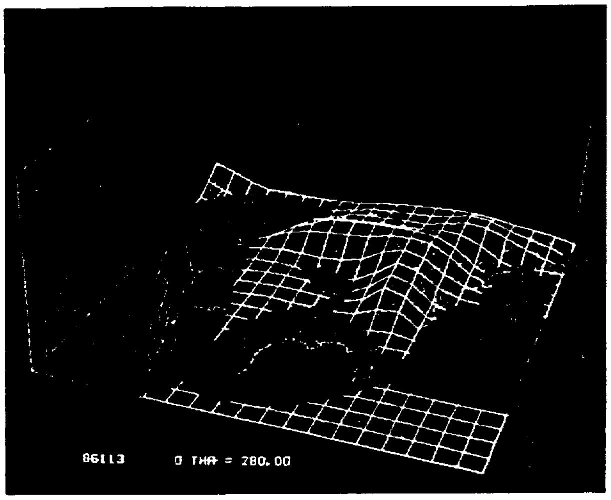


FIG. 5. 280 K potential-temperature surface over a simple map, generated from rawinsondes of 0000 GMT 23 April 1986.

FIG. 6. Wind-speed surfaces for 32 m · s^{-1} (semitransparent) and 48 m · s^{-1} (opaque) over a topographical map, generated from rawinsondes of 0000 GMT 17 April 1986.

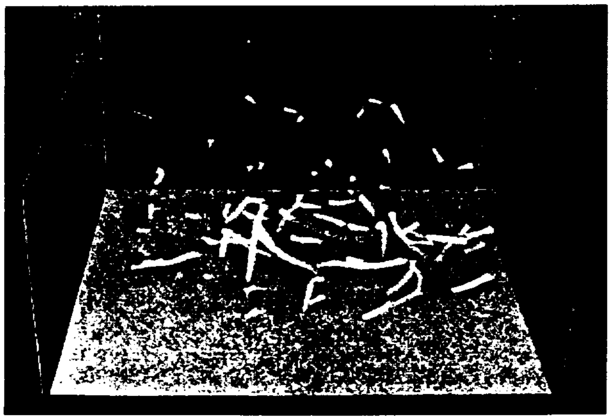


FIG. 7. Wind trajectories with trajectory thickness used to indicate echo intensity, generated in a frame of reference that moves with the storm. The storm is near Orienta, Oklahoma, on 2 May 1979.

per second. This gives the appearance of waves rolling along the moisture or isentropic surface.

The mixing-ratio or isentropic surface at each time is analyzed by finding the height of the desired surface at each sounding, and interpolating these heights to a 20 by 20 grid. The grids of heights are then linearly interpolated between sounding times.

More-complex shaded contour surfaces are generated from rawinsondes for a variety of meteorological parameters. The data are interpolated along each sounding to 10 evenly spaced height levels. On each level the data are analyzed to a uniform 20 by 20 grid and the surfaces are then located in the display through this three-dimensional grid. These are depicted as one or more shaded surfaces, some of which may be semi-transparent, over a topographical map (Fig. 6). This technique is capable of displaying oddly shaped surfaces which may contain folds or bubbles. The shapes of the surfaces are easy to understand, particularly when viewed in stereo. When viewing a transparent surface and an underlying opaque surface simultaneously, the eye is quite good at separating the shade variations of each surface and deducing their shapes. However, this breaks down when there are several transparent layers.

7. Radar data

Dual doppler radar data, consisting of three component wind vectors and echo intensities over three-dimensional 15 by 30 by 30 grids at a sequence of observing times between three and 12 minutes apart, have been analyzed and supplied by the National Severe Storms Laboratory. These data were supplied in grid sets that move with the storm. The data are displayed as a time sequence of shaded wind trajectories in a flat box, with trajectory thickness varied to represent echo intensity along the trajectories (Fig. 7). These trajectories are traced through the grids, subtracting average storm velocity from the winds, resulting in a storm relative display. The box is about 33 km on a side and 15 km high, so that the vertical-to-horizontal aspect ratio is approximately square. When these images are animated in stereo, they create a dramatic display of winds inside a thunderstorm.

Volumetric-radar data sets were supplied by the McGill University Radar Weather Observatory. These had been resampled by McGill to uniform 256 by 256 grids on 11 unevenly spaced vertical levels, with observation times 10 minutes apart. A stereo pair of images is generated at each observing time, consisting of a series of horizontal slices through the atmosphere (Fig. 8). Each slice is opaque where there are echoes and transparent elsewhere, with the ground-clutter echoes removed. The data are resampled in the vertical with the density controlled by the user. This allows the density of slices to be tuned according to size and orientation of the storm, so that high slices do not obscure low slices. The bottom of the box is flat with map boundaries and drop shadows from the lowest level of echoes. Note that the map boundaries in Fig. 8 are too coarse for the 256 by 256 km scale of the box. Finer map boundaries would be used with these images where radar data were available regularly.

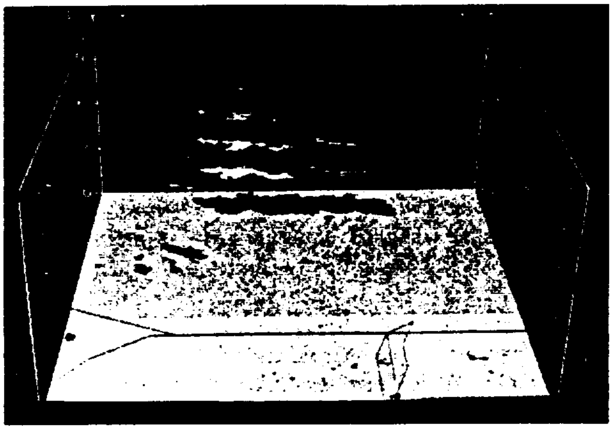


FIG. 8. Horizontal sections through a volumetric radar, with ground echoes removed, over a flat map with coarse boundaries. The data is centered near Mount Royal, Quebec, at 1710 GMT on 9 July 1981.

TABLE 1. CPU times and wall-clock times for representative sequences of image pairs for the image-generation techn[illegible] presented in this paper.

Image-generation technique	Number of image pairs	CPU time (secs)	Wall time (mins)
cloud images by remap and vertical shift	1	73	3.4
clouds over a topographical map, with artificial shade and edge detection	3	160	8.1
wind trajectories and mixing-ratio surface from VAS soundings, over a topographical map, covering nine hours and five sounding periods	7	220	13.3
wind streamlines and mixing-ratio surface from rawinsonde balloon soundings, over a simple map, with motion parallax	2	35	3.2
grid-mesh potential-temperature surface over a simple map	9	48	2.0
two semitransparent wind-speed surfaces from rawinsonde balloon soundings, over a topographical map	1	70	3.3
wind trajectories from dual doppler radar	7	97	9.8
horizontal sections of a volumetric radar	3	148	8.1

8. Conclusions

The image-generation techniques presented above integrate large quantities of four-dimensional meteorological data for animated stereo display. These techniques combine perspective, shading, depth precedence, transparency, brightness, shadows, and motion parallax with the stereo-display geometry in a consistent way to create the vivid illusion of a moving three-dimensional model of the atmosphere. Figure 9 contains small stereographs of four of these types of images.

Many of the elements in these images, such as fading trajectory tubes and clouds with edge detection, would be difficult or impossible to generate using commercially available 3-D graphics software. It is also useful to have the flexibility to experiment with image-generation techniques without the constraints of a large software package that was originally created for a different application.

While the displays presented in this paper create a striking four-dimensional illusion when viewed with stereo and animation, they are still a long way from reaching the potential of four-dimensional display for meteorologists. There are several ways they can be improved in the near future. First, elements from these displays will be integrated into combined displays. For example, a semi-transparent isotach contour surface may be combined with the shaded tube wind trajectories to help indicate the speeds of the trajectories. Or, in the study of severe weather, it might be useful to combine a semitransparent moisture surface with grid-mesh, potential-temperature surfaces at several different temperatures, and to add sea-level-pressure isobars drawn on the topographical surface. It would also be interesting to combine cloud images with volumetric-radar echoes, perhaps with the clouds modified to be semitransparent. With such a large number of potential combinations, more research and field testing will be essential in determining the degree of utility for any particular one. Second, the image-generation techniques of this paper will be applied to gridded output data of a sophisticated data-assimilation model that is being developed on the McIDAS system and is being run on a regular basis (Mills and Hayden, 1983). Third, the use of four-dimensiona[illegible]plays would benefit from making the numerical value [illegible] geographical locations of the displayed meteorological data more accessible. The location problem can be addressed by displaying views of the same scene from several different perspective points, or by using the four-dimensional display in conjunction with simple two-dimensional line graphics. The numerical values and locations of data can also be accessed using a three-dimensional cursor. The user would place this cursor at a point of interest in the display and retrieve its location in latitude, longitude, and height as well as the values of meteorological parameters at that point. The three-dimensional cursor should be implemented in such a way that it participates in the perspective and hidden-surface removal with the rest of the image elements.

The cloud-image generation with cloud-edge detection is the most complex software presented here, and attacks the most difficult problem. These cloud images are based on information from GOES data about the tops of clouds, and on surface hourly temperatures. They could be improved by adding information from surface hourly cloud-cover observations and VAS soundings to fill in the sides and bottoms of the clouds. These data could also help improve the accuracy of cloud-top heights and cloud-edge detection. The current cloud images have a tendency to all look like cumulus clouds. This could be addressed by adding a cloud-typing algorithm (Kittler and Pairman, 1985) to adjust the shade and texture of the clouds. These changes may help to improve the tendency of the current technique to display thick cirrus cloud[illegible] thunderstorms similarly.

Most of this work was done using red and green filters with the stereo display, so that it has been impossible to effectively

FIG. 9. To see the stereo effect with these four image pairs, look at the left image with your left eye and the right image with your right eye. It may help to place a piece of cardboard perpendicular to the page so that your left eye cannot see the right image and your right eye cannot see the left image. (a) Wind trajectories and a 5 g/kg mixing-ratio surface over a topographical map, generated from rawinsonde balloon soundings of 1200 GMT 23 April 1986. (b) Cloud image over a topographical map in the region of the Baja peninsula, generated from GOES data of 2100 GMT 24 April 1986 by remap with artificial enhancement of sun's shade and cloud-edge-detection algorithm. (c) Wind-speed surface for 40 m $\cdot$ s^{-1} over a topographical map, generated from rawinsonde balloon soundings of 0000 GMT 18 April 1986. (d) Wind trajectories with trajectory thickness used to indicate echo intensity, generated in a frame of reference that moves with the storm. The storm is near Lahoma, Oklahoma, on 2 May 1979.

experiment with the use of colored images. However, once a cross-polarized display is regularly available, color will be exploited as a visual cue.

One of the major features of this work has been the attention paid to efficiency. Table 1 lists the CPU time and the wall-clock time to compute representative sequences of image pairs for each of the image-generation techniques presented. The *CPU time* is the number of seconds to generate the sequence of image pairs on the IBM-4381 McIDAS, including the time for data analysis but not including the time for loading the images into the video terminal. The *wall time* is the number of minutes the user must wait, from the time the first command is entered until the sequence of image pairs is ready for viewing on the video terminal, assuming the system is not very busy. Note that the wall time is roughly twice the CPU time, plus a minute for each image pair to be loaded into the video terminal. Also note that the grid-mesh potential-temperature surfaces use the graphics mode of the video terminals and do not need image loads. Times for these techniques will vary depending on the time and distance scales being displayed, and on the values of the meteorological data. For example, cloud images are generated more quickly when there are fewer clouds.

A revolution is occurring in computer displays as computing power approaches the point where it is economical to produce images in a fraction of a second. This will allow the user to specify the perspective point and the contents of the display with a device like a mouse or a joystick and to get immediate response. This is currently possible for simple graphics and under development for complex and smoothly shaded images similar to those presented here (Fuchs et al., 1985 and Clark, 1982).

Acknowledgments. I wish to thank T. Whittaker for his help and for his own graphics software which provided the model for much of this work; J. T. Young, F. Mosher, and R. Krauss for their help and support; D. Santek for help with the contour surfaces; D. Bittorf for help with the cloud images; E. Brandes of the National Severe Storms Laboratory for supplying dual-doppler-radar data; A. Bellon of McGill University for volumetric-radar data; and J. Waite-Schuster for help with the manuscript. I also wish to thank C. Anderson, J. Benson, G. Dengel, R. Dengel, L. Garand, M. Hoerling, P. Meyer, C. Norton, E. Pope, D. Rotzoll, B. Smith, E. Suomi, V. Suomi, A. Weickmann, K. Weickmann, J. Weinman, D. Wylie, and the referees for many helpful suggestions. This work was supported by the National Aeronautics and Space Administration (NAS8-33799 and NAS8-36292).

References

Clark, J., 1982: The geometry engine: a VLSI geometry system for graphics. *Computer Graphics,* **16,** 127–133.

Coakley, J., and F. Bretherton, 1982: Cloud cover from high resolution scanner data: detecting and allowing for partially filled fields of view. *J. Geophys. Res.,* **87,** 4917–4932.

Foley, J., and A. Van Dam, 1982: *Fundamentals of Interactive Computer Graphics.* Addison-Wesley Publishing Company, 664 pp.

Fuchs, H., J. Goldfeather, J. Hultquist, S. Spach, J. Austin, F. Brooks, J. Eyles, and J. Poulton, 1985: Fast spheres, textures, transparencies, and image enhancements in pixel planes. *Computer Graphics,* **19,** 111–120.

Grotch, L., 1985: The use of interactive three-dimensional graphics in a climate research program. *Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology.* Los Angeles, American Meteorological Society, 213–215.

Grotjahn, R., and R. Chervin, 1984: Animated graphics in meteorological research and presentations. *Bull. Amer. Meteor. Soc.,* **65,** 1201–1208.

Hasler, A., H. Pierce, K. Morris, and J. Dodge, 1985: Meteorological data fields "in perspective." *Bull Amer. Meteor. Soc.,* **66,** 795–801.

Hibbard, W., R. J. Krauss, and J. T. Young, 1985: 3-D weather displays using McIDAS. *Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology.* Los Angeles, American Meteorological Society, 153–156.

Hibbard, W., and D. Wiley, 1985: An efficient method of interpolating observations to uniformly spaced grids. *Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology.* Los Angeles, American Meteorological Society, 144–147.

Kittler, J., and D. Pairman, 1985: Contextual pattern recognition applied to cloud detection and identification. *IEEE Trans. Geosci. Remote Sensing,* **GE-23,** 855–863.

Lovejoy, S., and D. Schertzer, 1986: Scale invariance, symmetries, fractals, and stochastic simulations of atmospheric phenomena. *Bull. Amer. Meteor. Soc.,* **67,** 21–32.

Marr, D., 1982: *Vision,* W. H. Freeman and Company, 397 pp.

Meade, A., 1985: Interactive display of clouds, terrain and other surfaces. M. S. thesis, Dept. of Atmospheric Science, Colorado State University.

Mills, G., and C. Hayden, 1983: The use of high resolution satellite temperature and moisture profiles to initialize a mesoscale numerical weather prediction model—A severe weather event case study. *J. Climate Appl. Meteor.,* **22,** 649–663.

Rogers, D., 1985: *Procedural elements for computer graphics.* McGraw-Hill Book Company, 433 pp.

Schiavone, J., T. Papathomas, B. Julesz, C. Kreitzberg, and D. Perkey, 1986: Anaglyphic stereo animation of meteorological fields. *Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology,* Miami, American Meteorological Society, 64–71.

●

ORIGINAL PAGE IS
OF POOR QUALITY

3rd Int. Conf. on Interactive Info. and Processing Systems for Meteor.,
4.3 Ocean., and Hydro., New Orleans, AMS, 1987

4-D DISPLAY OF WEATHER DATA ON MCIDAS

William Hibbard, Robert Krauss, David Santek and J. T. Young

Space Science and Engineering Center
University of Wisconsin
Madison, Wisconsin

1. INTRODUCTION

Meteorology is the science of the constantly moving four-dimensional atmosphere around us, but the meteorologist must currently rely on static two-dimensional plots of weather information for forecasting and research. A meteorologist wishing to understand a weather situation typically studies a large number of map plots of various types of weather data over a range of times and vertical levels. Skilled meteorologists mentally integrate these plots and create moving three dimensional pictures of the weather in their minds, but it would be useful to have a computer terminal which would directly present such integrated three dimensional moving displays of weather situations, omitting the need for time consuming study of two-dimensional plots and maps. To shed new light on atmospheric interactions and the behavior of predictive models, we need a faster and more efficient look at the atmosphere as a four-dimensional continuum.

The Space Science and Engineering Center (SSEC) of the University of Wisconsin, with its Man-computer Data Access System (McIDAS), has long been a leader in satellite and weather data processing (Suomi, 1983). The extensive use of animated sequences of two-dimensional images and graphics in McIDAS has proved very helpful in weather analysis. With a large data base of real-time observations as well as archive and model output now available, McIDAS capabilities are being expanded to handle four-dimensional data sets in x,y,z, and t.

To complement the expansion to four-dimensional data sets, a McIDAS based terminal is being developed at SSEC that provides the user a means of viewing animated sequences of 4-D weather data in stereo (Hibbard, Krauss and Young, 1985). This terminal uses two large screen projectors, creating different images for the left and right eye via polarization. An illusion of three dimensionality is created in the display, and by using time-lapse loops of a series of images, motion in the time domain is created. Image generation techniques have been developed at SSEC to take maximum advantage of this new terminal, integrating large quantities of four-dimensional meteorological data from balloon and satellite soundings, satellite images, doppler and volumetric radar, and conventional surface observations (Hibbard, 1986). The images have been designed to use perspective, shading, hidden surface removal and transparency to augment the animation and stereo display geometry. They create an illusion of a moving three-dimensional model of the atmosphere so vivid that you feel you can reach into the display and touch it. Other groups currently exploring aspects of multi-dimensional data display include the National Center for Atmospheric Research (Grotjahn and Chervin, 1984), Colorado State University (Meade, 1985), the NASA Goddard Space Flight Center (Hasler, Pierce, Morris and Dodge, 1985), and AT&T Bell Labs (Schiavone, Papathomas and Julesz, 1986).

No pictures are included with this paper, as still photographs and non-stereo viewing dramatically reduce the effect of the displays. Live demonstrations of this work will be presented using video projectors and a large screen in the University of Wisconsin exhibit as a part of this conference. Hibbard (1986) contains some images of stereo pairs in black and white.

2. HARDWARE

A deliberate decision was made at the beginning of the stereo terminal development program to minimize new hardware design and construction until we had a well-developed idea of what the stereo terminal actually should do in the McIDAS environment. The first two years started with developing the software that gave access to the database and constructed the 4-D data files needed to display the data. Then we experimented with the display of the data, trying to creatively find the best ways of communicating to the meteorologist via the stereo images. We are still in this stage of development, and will remain there for a while longer, as the section on future work suggests below.

The first experiments were done with two McIDAS terminals linked together, using two CRT displays and a half-silvered mirror. This was the minimal configuration we could use to provide color stereo. It required software to load one of the images backward, so the reflection would coincide with the direct view image. In addition, a link between the two terminals was required so they would change frames in synchrony. A lot of software development work could be done this way, but it tied up two terminals at a time and was therefore somewhat impractical. Moreover, only one or two people at a time could see the display because of the restricted field of view the half-silvered mirror introduced. We wanted a more efficient way to generate stereo with a single terminal that could be permanently available and be viewed by a number of people at once. To display color stereo with a single McIDAS terminal required more extensive terminal hardware modifications, and we have recently completed them. The color stereo displays in the exhibit hall are produced using a single McIDAS terminal.

One major constraint on the modification of a single McIDAS terminal for stereo development work was that the terminal continue to work like a standard terminal when used for non-stereo display. All the additions for stereo generation had to be transparent to the non-stereo user. In addition, we wanted to minimize the amount of modification, since this would only be an interim step to the final specification of a McIDAS stereo display terminal.

The current McIDAS terminal design is built on a multibus chassis with two bus masters, an Intel 8085 which drives the display hardware, and a PC/AT interface which serves as the link to the user and to the mainframe and provides additional local processing capability. Normally, a terminal has a single RGB board with its own internal timing and a lookup table which accepts 12 bits of digital image input and outputs 5 bits each of R,G, & B to a DAC for video generation.

What was done in this interim modification was to add a second video chain to a standard McIDAS terminal. This required four internal hardware changes:

1) The multibus P2 backplane was rewired to switch image memory to an additional new video board.

2) An external timing reference for the video was added, since the two video boards could no longer operate independently using their own timing references.

3) The logic controlling the boards was modified so they operated in parallel as far as access to memory and output to the video chain was concerned, but their ack's were designed to be serial, so both boards had to complete one operation before the next operation could begin.

4) The signal flow to the lookup tables was inverted for the second board so it would always receive the image for the opposite eye. The graphics planes were also disabled on the second video board since they would be needed for drawing the stereo cursor. The non-stereo hardware-generated cursor was retained to operate in parallel on both boards, however, so a consistent line-element reference would always be present in the two output images, if needed.

The result of these changes is that when using the second video channel to drive a monitor or projector with polaroid filters, one can see full color stereo using passive polaroid filter glasses, and one now has a stereo cursor as well, which can even participate in depth buffer [illegible]'s if the depth buffer is available in terminal memory. The first channel of video, however, is still identical to what an unmodified terminal would produce. The only drawback to this approach is that one does not have overlaid stereo graphics planes, since the graphics planes have been appropriated for the software generated stereo cursor.

3. SOFTWARE DESIGN CONSIDERATIONS

The atmosphere is a four-dimensional continuum. To depict the spatial portions it is necessary to show the location of data in the atmosphere using a variety of visual cues including perspective, stereo, motion parallax, depth precedence, shading, brightness, color, shadows and transparency. It is also necessary to integrate the use of different visual cues so that their effects are not ambiguous. For example, the perceived brightness of a scene is a function of illumination, shape, transparency and shadows. If these cues are used carefully, the viewer will be able to separate their effects and the scene is pleasing to view and easy to comprehend. If the cues are inconsistent, the result is confusion. Because the atmosphere is a continuum, an effective display must show information at all depths without nearer regions obscuring farther regions. Most computer graphics applications to date have been concerned with the surfaces of 3-D objects. In meteorology the concern is the three-dimensional volume. The programming challenge is to display representations of the data within that volume effectively so the information contained within the data is meaningfully perceived by the meteorologist.

An extensive software package has been written at SSEC for the generation and viewing of 3-D weather parameters. This includes assembling a 4-D data base from a variety of sources and rendering images from this data base through the use of highly tuned algorithms that enhance the 3-D effect. It is our desire to create 3-D images of atmospheric parameters, animated with time, that will show as clearly as possible the structure and evolution of weather systems. The images we currently generate include cloud images from GOES (Geostationary Operational Environmental Satellite) data, wind streamlines, wind trajectories, and contour surfaces from rawinsonde balloon soundings, VAS (VISSR Atmospheric Sounder) soundings, and model output at any scale (thunderstorm to global). Also, from dual doppler radar data, images are created depicting wind trajectories and multiple sections through a volumetric radar.

Meteorological data available in the McIDAS environment exist in the form of scalar fields (moisture, temperature, pressure, wind speed, etc.), vector fields (winds), images (satellite and radar images) and discrete phenomena (lightning and precipitation). Display techniques must be developed for each type of data, suited to the mathematical form as well as the density and texture of the data. It is also desirable to combine different types of data into a single display, but without confusion between parameters or too much clutter.

Nearly all of our 3-D images are based within a rectangular box, which is drawn as part of the image. This provides a visual reference for the objects inside, making it easier to visually extract relative positional information in space. The coordinates of the box are usually, but not limited to, latitude, longitude, and height. In the case of doppler radar or a thunderstorm model, the base of the box could be storm relative instead of earth coordinates. For large scale data, a base map can be drawn on the bottom of the box to aid in locating objects in the horizontal. A topographical relief map can also be added for relating weather parameters to the topography, and can provide a better land/ ocean boundary through brightness variation. The vertical axis is labelled in kilometers or pressure coordinates to help in determining the heights of volumes in the box.

The key to presenting 3-D images is creating the illusion of depth in the pictures. The use of stereo, perspective, hidden surface removal, and shading can be very effective in providing depth cues to the viewer. In addition, transparency, shadows, and motion will tend to heighten the illusion of reality. It is important to use these depth cues effectively and efficiently. Interactivity and quick regeneration of images are also prime concerns if this type of data display is to be used in an operational or research environment. It is important that the viewer's attention is not distracted by visual errors in the display due to bad data or to inadequate approximations.

The command structure for creating images is modular and provides the user with total control of parameters for assembling and viewing data sets. The meteorologist, however, does not have to concern himself with choices of shading algorithm parameters, light source placement, etc., as these should not have to be modified in most cases, and are buried one level deeper in the command structure.

As with two-dimensional meteorological displays, interactivity is important. The viewer should be able to control the contents and the view point of the display and generate a new display quickly. Because of the complexity of three-dimensional display algorithms, particularly for sequences of images, efficiency must be seriously addressed to achieve the desired level of interactivity. This is the main reason existing McIDAS terminals are merely interim steps. We need about two orders of magnitude more computing power in the terminal to get true interactivity.

Three-dimensional displays share a number of data oriented problems with more traditional displays. Bad data points can generate distracting and bizarre effects in an analysis. In fact, the greater volume of data used in generating three-dimensional displays makes them more susceptible to bad data problems. In most cases data density and coverage determines the scale of the display. However, variations in density and coverage can create problems. Gaps in the data can create misleading displays while too much data can create a cluttered display.

A particular problem of 3-D displays is the exaggerated vertical to horizontal aspect ratio that they often have. For a horizontal scale of hundreds or thousands of kilometers, it is necessary to expand the vertical scale by a factor of tens or hundreds. Although this is unavoidable, it can lead to misleading angles and slopes in the display.

Another particular problem of 3-D display is the difficulty of presenting precise quantifiable information. This is in part due to the ambiguity of depth, which makes it easier to see relative spatial relations than absolute location. For these cases an interactive 3-D pointing device in the image is essential. Like the 2-D version of McIDAS terminals, use of a cursor relates the image coordinates in the display to the true spatial coordinates in the data. If the stereo cursor partakes in depth buffer access, its usefulness is greatly enhanced because it can now also point behind objects.

4. CREATING 4-D METEOROLOGICAL IMAGERY

The first decision to be made in generating an image is the geographical extent of the 3-D box and the user's view point. An image of this basic "box", which can include topography, is generated, to which objects are then added. The default viewpoint works well for a synoptic scale box, but can be modified to view the scene from a different angle. Next, one must not only choose the parameter to examine, but also the value of the surface to generate and possibly a degree of transparency if more than one parameter or contour is to be displayed.

The decision tree is different for each case. For example, to find the upper level jet on a 2-D map, one would simply look on a 250 mb chart for the area of strongest winds. For a 3-D display, the question asked would be, "What wind speed defines the jet?". An image containing a volume showing the extent of the 50 m/s wind speed may be created. Or, a surface of 40 m/s may better define the jet with a jet-core speed of 75 m/s. Here the 40 m/s surface would be drawn semi-transparent and 75 m/s volume would be opaque. In this example the 3-D image would show the jet as it is positioned in the atmosphere, whereas, the 250 mb chart is just a slice through the atmosphere at a particular level. A time sequence could then show the jet meandering across the country changing in intensity and height.

Another example could be verifying model forecasts. We could take the 2-D grid output every 12 hours from the LFM for a 48 hour forecast run and generate 3-D grids of wind speed. For the next 48 hours we would take each current LFM initial condition as verification every 12 hours and also create 3-D grids for comparison. A sequence through the forecast period would be made by using different levels of transparency for forecast and verification jet core wind speed value.

Also, 2-D contours can be drawn on the topographical surface, such as surface pressure, precipitation totals, etc. This level of control allows the meteorologist to combine grids of different types or different contour levels, along with 2-D contours, to create more meaningful 4-D data sets and displays that show relationships of different kinds of data.

Time sequences of stereo images of isentropic wind trajectories are generated from rawinsonde balloon soundings or VAS soundings, combined with a grid mesh mixing ratio surface and a topographical map. The trajectories are depicted as shaded tubes of finite diameter. With animation they appear to trace a path through the atmosphere, with their tail ends fading into transparency as they move. The finite body of the trajectories makes it possible to use shade and hidden surface removal as cues to their shape and location. The fading into transparency enhances the sense of motion and serves as a cue to wind speed, as the degree of transparency along a trajectory is proportional to the length of time since the air parcel was at the points along the trajectory. The display clearly shows the relation between wind, moisture and topography over time.

Perspective line graphics of potential temperature and mixing ratio surfaces are generated from rawinsonde balloon soundings. These are grid mesh surfaces over map outlines. Because of their simplicity, long animation sequences can be generated quickly. They work best when the time interval between images is 3 hours and when they are animated at a rate of at least 3 per second. This gives the appearance of waves rolling along the moisture or isentropic surface.

More complex shaded contour surfaces are generated from rawinsonde balloon soundings for a variety of meteorological parameters. The data is interpolated along each sounding to 10 evenly spaced height levels. On each level the data are analyzed to a uniform 20 by 20 grid and the surfaces are then located in the display through this 3-D grid. These are depicted as one or more shaded surfaces, some of which may be semi-transparent, over a topographical map. This technique is capable of displaying oddly shaped surfaces which may contain folds or bubbles. The shapes of the surfaces are easy to understand when viewed in stereo. When viewing a transparent surface and an underlying opaque surface simultaneously, the eye is quite good at separating the shade variations of each surface and deducing their shapes. However, this breaks down when there are several transparent layers.

Most recently, atmospheric model data has piqued our interest because of the smoother fields, shorter time intervals, and lack of observational artifacts and data gaps. We can spend less time massaging the data and more time experimenting with generation of images. NMC model data is available only at 12 hour steps, but still has proved to be useful if the spatial scale is large enough. An assimilation model running on McIDAS provides us with better spatial and temporal scale data than the NMC model. The assimilation model has 15 vertical levels with 60 km grid spacing. By using one hour time intervals, animated sequences of isotach surface or vortex tubes show continuity through time. Other surfaces of interest include vertica motion and pressure deviation. Since a pa[illegible]lar pressure surface is relatively flat in [illegible] 15 km high box, we have found that display of th deviation of pressure from a climatological mean shows troughs and ridges better. .

The 3-D grids and displays developed are not restricted to any scale or data type. We have taken output from a thunderstorm model (Schlesinger, 1974) and generated surfaces of different parameters. Of particular interest is liquid water content which, when contoured at th .5 g/m**3 concentration, shows the time evolutio of a thunderstorm complete with anvil, low-level outflow and overshooting tops. Doppler radar data can also be displayed as trajectories. Whe these images are animated in stereo, in a coordinate system moving with the storm, they create a dramatic display of winds inside a thunderstorm. We are working now to combine the doppler winds with volumetric radar showing water concentratio

6. FUTURE WORK

In an earlier report (Hibbard, Krauss, an Young, 1985) we discussed the need to address four major aspects of stereo terminal development: 1) the human interface; 2) the display technology; 3) data base management; and 4) computer and software technology. Display technology is not yet sufficiently advanced to produce true volumetric displays, so we have chosen the traditional approach of present[illegible] separate color image to each eye. We have addressed the data base management problem to th extent that we can assemble and process a wide variety of data to generate the stereo images. More sophisticated methods of storage and access may be desirable in the future. Most of our recent work has been in software development and experiments relative to the generation of images. We invite you again to view our exhibit at this conference and offer suggestions. We still need to address the questions of what the optimum user-computer interface really is. Now that we have developed a range of display options, we ca address how the meteorologist will use them in concert to enhance his perception of what the true physical situation in a data set is and how that physical situation is evolving, and develop a McIDAS terminal environment to aid this perception.

A key aspect of interactivity is the ability of the scientist to access all the data at a given point in the displayed McIDAS image by pointing at that location with a cursor. This capability must be preserved in the stereo display. The location problem can be addressed by displaying views of the same scene from several different perspective points, or by using the four dimensional display in conjunction with simple two dimensional line graphics. The numerical values and locations of data can also be accessed using a 3-D cursor. The user would place this cursor at a point of interest in display and retrieve its location in latitu longitude and height as well as the values of meteorological parameters at that point. The 3-D cursor must be implemented in such a way that it participates in the perspective and hidden surface removal with the rest of the image elements.

Line graphics, because they do not appear in nature, tend to detract from the physically real appearance of the stereo display. There is a place for line graphics, but we suspect that it is in displaying computed theoretical quantities, or used to mark or annotate various features. Line graphics can be used for abstract mathematical quantities, to distinguish them from the physical surfaces or volumes in the images. Line graphics might also be useful in drawing cross-sections of the atmosphere, introducing 2-D plots into the 3-D display at key positions to aid in analysis.

While most surfaces can be interpreted using a single light source and appropriate shading, even when the surfaces are highly convoluted, it is clear that use of both color and texture, along with shadows and multiple light sources, can aid in helping the viewer to comprehend curvature of highly unintuitive surfaces. Most of the work to date was done using red and green filters with a single monitor stereo display. Now that a cross polarized display is regularly available, color will be exploited as a visual cue. Texture can also aid in making a surface appear more realistic, and in discriminating one surface from another when they intersect in three dimensions.

We mentioned that the user interface has yet to be addressed in any great detail in our development program. Part of the delay has been because we were not clear just what was to be manipulated and how the scientist would use the displays in his analysis. As we get scientists at SSEC more involved with using the stereo displays for actual data analysis over the next two years, we will be able to improve the interfaces to speed up the generation and presentation of images. We also intend to experiment both with voice I/O as a substitute for keyboard use (so the user will not have to take his eyes off the display), and introduce a certain amount of "intelligence" into the terminal to guide the user in appropriate ways and suggest useful alternatives depending on context.

The development of faster CPU's and cheaper memory will clearly help design of a more powerful stereo terminal. It is unlikely, however, that a general purpose processor will be able to provide the fast image generation capability required for a terminal to be truly interactive. A revolution is occurring in computer displays as computing power approaches the point where it is economical to produce images in a fraction of a second with custom hardware. This will allow the user to specify the perspective point and the contents of the display with a device like a mouse or a joystick and get immediate response. This is currently possible for simple graphics and under development for complex and smoothly shaded images similar to those presented here (Fuchs, et. al., 1985; and Clark, 1982).

We intend also to introduce a 6-D "joystick" to indicate both position and direction in the stereo display. This will be done using a simple current carrying coil moving in a 3-D magnetic field surrounding the terminal operator. The field is roughly two meters on a side, and the position and attitude of the coil is periodically sensed by means of the current it generates as it moves. The coil can be worn on a finger, so the operator need only point and give a verbal command to initiate terminal action.

7. CONCLUSIONS

Currently, work on stereo terminal specification is proceeding on schedule and is getting more sophisticated in terms of types of data accessed and complexity of image generation. Scientists' interest appears to be increasing as they see some of the benefits of adding an extra dimension to the display of their data. With the introduction of displays of model output we hope to heighten this interest even further. Within two years we expect to have a wide variety of possible displays and a user interface capable of guiding a novice user to produce meteorologically meaningful displays.

8. ACKNOWLEDGMENTS

We wish to thank Tom Whittaker, Don Wylie and Russ Dengel for their help and suggestions over the past two years.

9. REFERENCES

Grotjahn, R. and R. Chervin, 1984: Animated graphics in meteorological research and presentations. Bull. Amer. Meteor. Soc. 65, 1201-1208.

Hasler, A., H. Pierce, K. Morris, and J. Dodge, 1985: Meteorological data fields "in perspective". Bull. Amer. Meteor. Soc. 66, 795-801.

Hibbard, W., R. J. Krauss, and J. T. Young, 1985: 3-D weather displays using McIDAS. Proceedings, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Los Angeles, Amer. Meteor. Soc., 153-156.

Hibbard, W., 1986: Computer Generated Imagery for 4-D Meteorological Data. Bull. Amer. Meteor. Soc. (accepted for publication).

Meade, A., 1985: Interactive display of clouds, terrain and other surfaces. M. S. Thesis, Dept. of Atmospheric Science, Colorado State University.

Schiavone, J., T. Papathomas, B. Julesz, C. Kreitzberg, and D. Perkey, 1986: Anaglyphic stereo animation of meteorological fields. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Miami, Amer. Meteor. Soc., 64-71.

Schlesinger, R. E., 1974: A Three-Dimensional Numerical Model of an Isolated Deep Convective Cloud: Preliminary Results. J. Atmos. Sci., 32, 934-957.

Suomi, V. E., R. Fox, S. S. Limaye, and W. L. Smith, 1983: McIDAS III: A Modern Interactive Data Access and Analysis System. J. Clim. and Appl. Meteor., 22, 766-778.

Proceedings, 1987 Digital Image Processing and Visual Comm. Tech. in Met., Cambridge, SPIE

4-D display of satellite cloud images

William L. Hibbard

University of Wisconsin, Space Science and Engineering Center
1225 West Dayton Street, Madison, Wisconsin 53706

ABSTRACT

A technique has been developed to display GOES satellite cloud images in perspective over a topographical map. Cloud heights are estimated using temperatures from an infrared (IR) satellite image, surface temperature observations, and a climatological model of vertical temperature profiles. Cloud levels are discriminated from each other and from the ground using a pattern recognition algorithm based on the brightness variance technique of Coakley and Bretherton. The cloud regions found by the pattern recognizer are rendered in three-dimensional perspective over a topographical map by an efficient remap of the visible image. The visible shades are mixed with an artificial shade based on the geometry of the cloud-top surface, in order to enhance the texture of the cloud top.

1. INTRODUCTION

A variety of techniques for displaying satellite cloud images with depth have been developed on the Man-computer Interactive Data Access System (McIDAS). The first of these was a simple stereo display of simultaneous images from two different GOES satellites, usually from one satellite over the eastern United States and another over the western United States. Then a technique was developed to recreate this stereo effect from a single GOES satellite. The cloud-top heights are estimated from the temperature of the IR image, and these heights are used to simulate the image seen from another satellite by shifting pixels horizontally according to height.

These simple techniques rely on binocular stereo as the only visual cue to cloud height, which makes it impossible to see small height variations and textural detail. In order to address this problem we developed a technique for rendering GOES images in a perspective view. As before, the cloud height is estimated from IR temperature. The GOES image is remapped into a perspective projection and then pixels are shifted vertically according to height. This can be viewed in stereo by remapping the right and left eye views to two slightly different perspectives. These images work well in some cases, but still lack shading texture when the sun angle is high. They also depict cloud edges as connected to the ground.

In this paper I will describe an improved technique for rendering GOES images in perspective. Cloud heights are estimated using temperatures from the IR image and a climatological model of vertical temperature profiles. Clouds are discriminated in the image using a pattern recognition algorithm based on a brightness variance technique, with surface temperature observations used to help discriminate low clouds from ground. These clouds are rendered in three-dimensional perspective over a topographical map by an efficient remap of the visible image. In order to enhance the texture of the cloud top, th natural sun shading in the visible image is enhanced by mixing with an artificial shade derived from the geometry of the cloud-top surface.

2. DISCRIMINATING CLOUDS

The input to the discrimination process consists of a GOES visible and IR pair of images Because of computer memory limitations, these images are processed in bands of 80 lines rather than all at once. As the program reads the images, it selects a sector that just covers the region of interest specified by the user (which is a rectangle of latitude and longitude). This sectorizing process includes a blow-up or blow-down, if necessary, to a resolution comparable to the resolution of the perspective image being produced. After the images have been read, the input IR data are smoothed. Each IR pixel is averaged with its four neighbors, and the weighting of the average is adjusted according to whether the value of the pixel lies inside or outside the range of values of its neighbors, so that noisy pixels are smoothed more than other pixels.

Next, the IR radiances are converted to heights. First, the IR radiances are converted to temperature by a simple formula. These temperatures are then compared to a climatological model of vertical temperature profile which is parameterized by date and latitude, and the height is estimated. These profiles are interpolated between the latitudes at the top and bottom of the input image. They are also modified by the surface hourly temperatu observations (at the time of the GOES data) in order to improve the accuracy of height estimates for low clouds. The visible radiances are also brightness normalized by a simple multiplicative factor depending on latitude and date to remove the effects of sun angle.

The variance technique of Coakley and Bretherton (1) is designed to find the IR radian intervals associated with cloud layers in an IR satellite image. In this case the technique is applied to an image whose pixel values are the heights derived from the IR radiances. This height image is divided into sectors of 80 pixels by 80 pixels, and the analysis is done for each sector independently. A histogram of the heights in the sector is constructed and the height difference (=D) between its fifth and ninety fifth percentile is calculated. The average height and the variance of heights is calculated for each of the 1600 two by two boxes in a sector, and a histogram is formed of the average heights of those boxes whose variance is less than 0.05*D. The idea is that boxes with small variance will tend to lie on flat cloud levels rather than on slopes between clouds. The histogram gives frequency as a function of height, and the clusters in this histogram will give height intervals associated with cloud layers.

Our cluster detection algorithm was developed by trial and error, and several of its parameters can be controlled by the user when they are dissatisfied with the default values. The cluster finder starts by applying a low pass filter to the histogram. This filter can be controlled by a user parameter. Moments are calculated for the histogram and a ramp function is added to accentuate the skew of the histogram. This step is a good example of the trial and error development of the algorithm. We originally added this step to remove skew from the histogram, but found that the final results were better when we accentuated skew. Then the algorithm finds the peaks of the histogram, defined as maxima within a certain range of heights, where the range can be controlled by a user parameter. For each peak, we find an associated height interval and calculate its area (sum of frequencies), the mean frequency and the variance of frequencies within the interval. These values are then used to determine which peaks should be merged into adjacent larger peaks. This step uses two parameters which can be controlled by the user. A range of heights is found for each of the merged peaks, and these ranges define the final clusters for the histogram.

The primary goal of the cloud detection algorithm is to select those pixels which are in clouds and to delete the rest. Assuming that the histogram clusters give the height ranges of cloud layers, a simple algorithm would just delete those pixels whose height does not belong to any cluster. Our algorithm is an elaboration of this developed by trial and error. First, we find the correspondence between the clusters of adjacent sectors (the 8 pixel by 80 pixel sectors) which are at roughly the same height levels. At the boundaries of sectors there is a transition zone where the cluster ranges are interpolated between two sectors. If a cluster has no corresponding cluster in an adjacent sector, then its range is smoothly tapered to empty at the sector boundary. Once the clusters and their interpolations between sectors are derived, the pixel deletion process starts. Pixels not in clusters are deleted. Pixels in clusters at or near the ground are deleted if their normalized visible brightness is low or if the local slope of the height surface is high, with the exact mixing of these conditions controllable by user parameters. Pixels not deleted will be grouped in cloud regions in the image. All pixels within a certain distance of the border of a region, whose height is less than the low limit of the region's cluster range, are deleted. Finally, pixels whose deletion causes small holes in cloud regions are reselected for inclusion (that is, they are undeleted).

3. RENDERING CLOUDS

The input to the rendering process consists of an image of heights and an image of normalized visible intensities, with some pixels marked for deletion. The first stage of rendering is to mix the visible intensities with an artificial shade based on the cloud-top geometry. The artificial shade calculation is a simple table lookup based on the slopes of the height surface. This shade is mixed with the normalized GOES visible brightness in a proportion which the user can control.

Now the problem is to map the modified visible image onto the surface defined by the selected pixels (those in cloud regions) of the height image and to render this in perspective. An obvious approach is to render every little square of four adjacent selected pixels as a polygon, interpolating between the visible shades at the corners. However, an image remapping technique will render the image with much less computational effort. The remap technique used here is a two-stage process. In the first stage, each line of the visible image is mapped via the perspective transform into the destination image, forming a sequence of short line segments stretching roughly horizontally across it. Each short line segment is drawn between the mapped positions of two adjacent pixels from the source line. In the second stage, pixels in the destination image are shaded by interpolating verticall between the series of line segments mapped from successive source lines. This approach avoids the complexity of interpolating in two dimensions by factoring the interpolation into a horizontal component (along source lines) in the first stage, and a vertical component (along destination columns) in the second stage.

This remap algorithm suffers from a couple of problems. First, it cannot accommodate an arbitrary perspective (that is, arbitrary point of view), because of the requirement that a source line map to a sequence of line segments stretching roughly horizontally across the destination image. However, this could be easily remedied by adding an initial step to rotate the source images. Second, in areas of the image close to cloud edges, where the cloud-top surface has large slope, the series of short line segments will turn from horizontal to nearly vertical, resulting in a frayed cloud edge. This has been remedied by adding an intermediate step between the two basic remap stages which smoothes between line segments at cloud edges.

4. THE USER'S VIEW

The McIDAS system manages a large inventory of GOES images, including real-time images and non-real-time images. The real-time images usually consist of images covering North America and the globe, at different resolutions, every half hour for the last two hours. Non-real-time images are retrieved from an archive containing almost all GOES images generated during the last ten years. Given a GOES IR and visible pair of images, the user can enter a simple command which causes these to be rendered in perspective as a stereo pair over a topographical map, and loaded into a video workstation. The user may specify latitude, longitude and height bounds for the box of data to be displayed. The user also has control over some of the parameters of the cloud discrimination algorithm, although the default values usually work fairly well. These perspective stereo cloud images can be rendered from a sequence of GOES images spaced every half an hour and then displayed with animation to see cloud motion. These perspective images work well to reveal cloud-top motion, including the boiling top of a convective system and the interaction of clouds with topography.

The time to analyze the GOES IR and visible pair for cloud detection and to render the clouds in perspective varies between 20 and 30 seconds, depending on the number of clouds present and the scale of the display. The time to produce the topographical map varies between 6 and 12 seconds, depending on scale. When a time sequence of images is produced, the topographical map only has to be rendered once. In a test, a sequence of three stereo pairs of perspective images was rendered in 160 seconds. These times are all measured for the software running on an IBM 4381.

5. CONCLUSIONS

There are several ways that the detection and rendering of clouds described here could be improved. More accurate cloud-top height estimates could be made using vertical temperature profiles of the atmosphere from RAOB (balloon) or VAS (satellite) soundings. A cloud-typing algorithm (2) could be used to increase the accuracy of cloud detection and rendering. For example, the IR temperatures of thin cirrus could be corrected for their partial transparency. The sides and bottoms of clouds could be approximated based on known properties of cloud types.

A number of people who have viewed these cloud images have pointed out a curious aspect of them. Although they cover regions hundreds of kilometers across, their appearance agrees with our expectations of clouds based on our experience with regions only a few kilometers across. Clouds over large scale regions viewed from space look very flat, but in our images, in order to bring out the vertical detail of the cloud top, the vertical scale has been greatly exaggerated. This suggests that clouds obey a fractal scaling law, in which the shapes are invariant to a scale change if the vertical to horizontal aspect ratio is changed with scale.

6. ACKNOWLEDGMENTS

I wish to thank F. Mosher, who developed the basic algorithm used to derive cloud-top height from the GOES IR intensity and provided many helpful suggestions. I also wish to thank E. Suomi and L. Garrand for helpful suggestions. This work was supported by the National Aeronautics and Space Administration (NAS8-33799 and NAS8-36292).

7. REFERENCES

1. J. Coakley, and F. Bretherton, "Cloud cover from high resolution scanner data: detecting and allowing for partially filled fields of view," J. Geophys. Res., 87, 4917-4932 (1982).

2. L. Garand, "Automated classification of oceanic cloud patterns with applications to cloud type dependent retrievals of meteorological parameters," Digital Image Processing and Visual Communications Technologies in Meteorology, Cambridge, SPIE (1987).

This remap algorithm suffers from a couple of problems. First, it cannot accommodate an arbitrary perspective (that is, arbitrary point of view), because of the requirement that a source line map to a sequence of line segments stretching roughly horizontally across the destination image. However, this could be easily remedied by adding an initial step to rotate the source images. Second, in areas of the image close to cloud edges, where the cloud-top surface has large slope, the series of short line segments will turn from horizontal to nearly vertical, resulting in a frayed cloud edge. This has been remedied by adding an intermediate step between the two basic remap stages which smoothes between line segments at cloud edges.

4. THE USER'S VIEW

The McIDAS system manages a large inventory of GOES images, including real-time images and non-real-time images. The real-time images usually consist of images covering North America and the globe, at different resolutions, every half hour for the last two hours. Non-real-time images are retrieved from an archive containing almost all GOES images generated during the last ten years. Given a GOES IR and visible pair of images, the user can enter a simple command which causes these to be rendered in perspective as a stereo pair over a topographical map, and loaded into a video workstation. The user may specify latitude, longitude and height bounds for the box of data to be displayed. The user also has control over some of the parameters of the cloud discrimination algorithm, although the default values usually work fairly well. These perspective stereo cloud images can be rendered from a sequence of GOES images spaced every half an hour and then displayed with animation to see cloud motion. These perspective images work well to reveal cloud-top motion, including the boiling top of a convective system and the interaction of clouds with topography.

The time to analyze the GOES IR and visible pair for cloud detection and to render the clouds in perspective varies between 20 and 30 seconds, depending on the number of clouds present and the scale of the display. The time to produce the topographical map varies between 6 and 12 seconds, depending on scale. When a time sequence of images is produced, the topographical map only has to be rendered once. In a test, a sequence of three stereo pairs of perspective images was rendered in 160 seconds. These times are all measured for the software running on an IBM 4381.

5. CONCLUSIONS

There are several ways that the detection and rendering of clouds described here could be improved. More accurate cloud-top height estimates could be made using vertical temperature profiles of the atmosphere from RAOB (balloon) or VAS (satellite) soundings. A cloud-typing algorithm (2) could be used to increase the accuracy of cloud detection and rendering. For example, the IR temperatures of thin cirrus could be corrected for their partial transparency. The sides and bottoms of clouds could be approximated based on known properties of cloud types.

A number of people who have viewed these cloud images have pointed out a curious aspect of them. Although they cover regions hundreds of kilometers across, their appearance agrees with our expectations of clouds based on our experience with regions only a few kilometers across. Clouds over large scale regions viewed from space look very flat, but in our images, in order to bring out the vertical detail of the cloud top, the vertical scale has been greatly exaggerated. This suggests that clouds obey a fractal scaling law, in which the shapes are invariant to a scale change if the vertical to horizontal aspect ratio is changed with scale.

6. ACKNOWLEDGMENTS

I wish to thank F. Mosher, who developed the basic algorithm used to derive cloud-top height from the GOES IR intensity and provided many helpful suggestions. I also wish to thank E. Suomi and L. Garrand for helpful suggestions. This work was supported by the National Aeronautics and Space Administration (NAS8-33799 and NAS8-36292).

7. REFERENCES

1. J. Coakley, and F. Bretherton, "Cloud cover from high resolution scanner data: detecting and allowing for partially filled fields of view," J. Geophys. Res., 87, 4917-4932 (1982).

2. L. Garand, "Automated classification of oceanic cloud patterns with applications to cloud type dependent retrievals of meteorological parameters," Digital Image Processing an Visual Communications Technologies in Meteorology, Cambridge, SPIE (1987).

Proceedings, 1987 Digital Image Processing and Visual Comm. Tech. in Met., Cambridge, SPIE

4-D Techniques for Evaluation of Atmospheric Model Forecasts

David Santek, Lance Leslie, Brian Goodman, George Diak, Geary Callan

University of Wisconsin, Space Science and Engineering Center
1225 West Dayton Street, Madison, Wisconsin 53706

ABSTRACT

Evaluating model performance is difficult in light of the amount of data which is output that is to be compared to an analysis for verification. Examining 2-D plots of many different parameters at several levels is tedious. This information must then be mentally integrated in an attempt to understand where problems may exist. The use of 4-D displays can greatly aid in evaluations by presenting the output of the model as a volume instead of 2-D slices. A capability for 4-D displays of meteorological data is being developed at the Space Science and Engineering Center. The Man-computer Interactive Data Access System (McIDAS) is used for all aspects of the analysis: this includes acquiring data, running the model, storing the output and displaying the results. A version of the Australian Regional Analysis and Forecast Modules were applied to the eastern portion of the USA and adjacent Atlantic Ocean. This assimilation system is being used to analyze intensive observing periods during the GALE (Genesis of Atlantic Lows Experiment) field experiment.

Many different types of data are presented as 3-D contours (surfaces). In addition to the use of perspective, hidden surface removal, and transparency, the 3-D effect is enhanced through the use of animation and stereo viewing. Changes in the shapes and sizes of the surfaces with time can indicate how well the model is performing. By displaying both the model result and the verification of isotachs, for instance, a visual check quickly identifies differences in the jet position and strength. Or, the deviation of the model from the analysis can be generated to check for structure in the errors both spatially and temporally.

1. INTRODUCTION

Atmospheric models provide the meteorologist a look into the future by generating forecasts of various meteorological parameters. These complex models attempt to approximate the 4-dimensionality of the atmosphere around us. It is unfortunate that the output from the models consists of 2-D plots, usually horizontal slices for select vertical levels or vertical cross-sections for selected transects. The information must then be mentally integrated to picture the atmosphere as a continuum in time and space. This is a tedious task requiring examining and re-examining many contour maps. A need exists to bypass this type of analysis, not only because it is time consuming, but also it is incomplete.

A capability to display meteorological data in 4-D is being developed on the McIDAS at the Space Science and Engineering Center. McIDAS is recognized worldwide as a leader in the processing of satellite and weather data[1]. An extensive data base of weather observations, satellite imagery, and model output is available, much of it in real time, to generate data sets in x,y,z, and time. Three-dimensional images are generated using highly tuned rendering algorithms making use of hidden surface removal, transparency, and shading. The addition of stereo viewing and animation results in a depiction of the atmosphere in motion.

The entire analysis is done on the McIDAS. The large data base is drawn from for rawinsonde and satellite data, and also for model output from the NMC (National Meteorological Center) Global model to initialize the numerical model on the McIDAS. The model, a version of the Australian Regional Analysis and Forecast Modules[2], is being used on McIDAS to analyze intensive observing periods during the GALE of early 1986. The output from the model are stored as 3-D grids for use with the display software.

No photographs are included in this paper. A video tape will be shown as part of the presentation of the paper, to illustrate the techniques described. See Reference 3 for examples of pictures of stereo pairs.

2. SOFTWARE

An extensive software package is being developed to create, manage, and display 4-D data sets on McIDAS[4]. The modular design provides the necessary flexibility to assemble 3-D grids from a variety of sources and file in a common grid structure. Only one display program is needed to use these 3-D grids, regardless if the data is from a thunderstorm model or a global model. If U,V (horizontal winds) and W (vertical wind) are available as 3-D grids for many times, parcel trajectories can be generated and displayed similarly.

The display of these data sets can take on many forms, summarized as follows (with typical examples):

- Transparent objects: Wind maxima or vortex tubes
- Wire mesh: Mixing ratio surface
- Contours: 2-D graphics
- Tubes: Air parcel trajectories.

These objects or contours are drawn within a rectangular box that usually defines the latitude, longitude, and height limits. The software allows the combination of many different parameters in one display. Also, topographical information can be included along with base maps.

3. NUMERICAL MODEL

The Australian Regional Model is an analysis/forecast assimilation system. This allows for the insertion of meteorological data during the model run, to update the forecasts without an absolute re-initialization. It is applied to the Eastern half of the USA and adjacent Atlantic Ocean during the GALE period. Specifically, the model is a 65 X 65 grid on a Lambert conformal conic projection with a horizontal spacing of 60 km. There are 15 vertical levels in sigma-p coordinates, with a top at approximately 100 mb.

The case examined is a 24-hour forecast run, initialized with the NMC Global model and conventional synoptic data, with no additional assimilation of other data throughout the 24 hours. Output from the model is hourly in the form of 2-D grids (65 X 65) interpolated to the 10 mandatory pressure levels for height, temperature, dew point temperature, winds, and vertical motion. Also, 2-D grids at the surface are generated, for example mean sea-level pressure and precipitation. These model generated grids are then mapped onto a 30 X 30 X 10 grid, equally space in latitude, longitude, and height. This reduces the resolution by about a factor of 2 in each of the three dimensions from the model.

4. 4-D METEOROLOGICAL IMAGERY

Model performance is usually determined by calculating a RMS (Root Mean Square) error[2], skill score[5], or comparison of 2-D plots[6]. These are static displays that use only a small part of the total output from the model. By using the 4-D displays on McIDAS we can examine the 3-D grids in stereo and with animation. Though, it does not quantify the model's deviation from the true atmospheric conditions, it does provide a 4-D visual comparison.

The case explored is a 24-hour forecast for 27 January 1986 characterized by a strong upper level jet (75 m/s), a closed circulation through most levels, and a developing low pressure system at the surface. Two major types of 4-D displays are generated. First, time sequences of model parameters every hour are animated. This shows parcel trajectories along with transparent wind maxima volumes or vortex tubes. Surface contours of mean sea-level pressure and precipitation are also added into the displays. The use of animation and depth cues heighten the 3-D effect in the non-stereo viewing. Secondly, every 12 hours an analysis can be compared to the forecast. At these two times, 3-D surfaces from the forecast and the analysis will be in the same display. Comparisons of jet strength, extent, and location can be visually checked with one image, instead of many 2-D plots. Other parameters are also displayed, such as vorticity centers and the pressure field at the surface.

5. CONCLUSIONS

The use of 4-D display has given modelers at SSEC an added dimension in viewing model output. In one case, assimilating data into a particular forecast run resulted in a large difference in the jet strength. The 3-D view of this change was much more striking than the 2-D plots. We have also found that including more familiar 2-D contours in 3-D displays helps the meteorologist to better interpret the 3-D.

Development continues in all aspects of 4-D data sets and displays. This includes the integration of data from satellite images, conventional soundings, satellite soundings, doppler radar, weather radar, and output from various models. The eventual goal is to present meteorological data in three dimensions that conveys new information to the scientist that would not have been possible with traditional 2-D plots.

6. ACKNOWLEDGMENTS

We wish thank Bill Hibbard, the developer of the 4-D software on McIDAS. This work was supported by the National Aeronautics and Space Administration (NAS8-33799 and NAS8-36292).

7. REFERENCES

1. Suomi, V. E., R. Fox, S. S. Limaye, and W. L. Smith, "McIDAS III: A Modern Interactive Data Access and Analysis System", *J. Clim. and Appl. Meteor.*, *22*, 766-778 (1983).
2. Goodman, B. and G. Diak, "Analysis and Forecast Experiments Incorporating Satellite Soundings and Cloud and Water Vapor Drift Wind Information", Preprints, 2nd Conf. on Satellite Meteorology/Remote Sensing and Applications, Amer. Meteor. Soc., 142-144 (1986).
3. Hibbard, W., "Computer Generated Imagery for 4-D Meteorological Data", *Bull. Amer. Meteor. Soc.*, *67*, 1362-1369 (1986).
4. Hibbard, W., R. Krauss, D. Santek, and J. T. Young, "4-D Display of Weather Data on McIDAS", Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, New Orleans, Amer. Meteor. Soc., 89-93 (1987).
5. Mills, G. and C. M. Hayden, "The Use of High Horizontal Resolution Satellite Temperature and Moisture Profiles to Initialize a Mesoscale Numerical Weather Prediction Model - A Severe Weather Event Case Study", *J. Clim. and Appl. Meteor.*, *22*, 649-663 (1983).
6. McGregor, J. L., L. M. Leslie and D. J. Gauntlett, "The ANMRC Limited-Area Model: Consolidated Formulation and Operational Results", *Mon. Wea. Rev.*, *106*, 427-437 (1978).

4.3 Proceedings, 4th Int. Conf. On Interactive Info. and Processing Systems for Met., Ocean., and Hydro., Anaheim, AMS, 1988

A NEXT GENERATION MCIDAS WORKSTATION

William L. Hibbard

Space Science and Engineering Center
University of Wisconsin
Madison, Wisconsin

1. INTRODUCTION

This paper describes the goals of and a proposed design for a next generation Man-computer Interactive Data Access System (McIDAS) workstation. The basic drive behind this is a requirement for a new 4-D workstation which combines animation with depth in its displays. However, it also covers the needs for a general McIDAS workstation.

The current McIDAS workstation includes a video system which stores and displays images and graphics, and a processor which controls the video system and communicates with the user and with McIDAS applications software running in the IBM mainframe host computer. The video system stores up to 128 image frames and 64 graphics frames of 480 lines by 640 elements. Images and graphics are independently animated and colorized, with graphics and cursor overlaying images. The workstation processor is an Intel 8085 running a fixed control program, which has recently been upgraded by adding an IBM-PC/AT.

Although our workstation was at the state of the art ten years ago, now there are other science and engineering workstations with superior technology. However, McIDAS still has the advantage of a workstation highly integrated with meteorological data sources and applications software. We are interested in developing a workstation which integrates new technology into McIDAS. Capabilities for larger frames, more bits per pixel, faster frame loads, pan, zoom, and greater interactivity should be integrated with McIDAS data management and easy access by applications software.

We will describe the requirements for a next generation McIDAS workstation, a proposed system based on the AT&T Pixel Machine, and designs for integrating its video memory and processor into McIDAS.

2. BASIC REQUIREMENTS

The most basic requirement for a new workstation is to preserve our current capabilities. Thus, it must be able to store about 100 image and graphics frames of about 480 lines by 640 elements. The video output must be able to flexibly color video and graphics, to combine two or more image frames, and to overlay video with graphics and a cursor. It must also be able to generate two independent displays simultaneously for stereo. The workstation must be able to animate at rates of about 10 frames per second, with independent control of images and graphics. The cost should be competitive with the current cost.

In addition to the current capabilities, a new workstation must store image frames with at least eight bits per pixel and without any statistical encoding. More than eight bits per pixel would be desireable for GOES IR data and other new sources. The graphics frames should have at least three or four bits per pixel.

The new workstation should also include some features which enhance interactivity by reducing response time for the user's access to image data. These include large and variable sized frames with hardware pan and zoom, and faster frame loads.

3. INTERACTIVITY

Interactivity is fundamental to McIDAS. The user at the workstation enters commands for loading image frames and generating graphics which are executed within seconds or minutes. The user can manipulate the colorization of the display with immediate response. Earth coordinates and values of meteorological parameters can be retrieved at the cursor location within seconds. The cursor can also be used to direct correlation wind measurements, sounding retrievals, and cross section analyses.

Improved interactivity is the most important goal for a new workstation. McIDAS has recently been used to extract cloud detail from Voyager images of Uranus, a difficult problem requiring filtering, remapping, brightness normalization and other image processing steps. This task would be easier if the user could get immediate visual feedback to mouse or joystick control over the parameters of these image processing functions. Some modern workstations offer this capability.

The 3-D displays we are developing on McIDAS especially require a workstation with great computational power for interactivity.

PRECEDING PAGE BLANK NOT FILMED

Three-dimensional displays have special problems of visual ambiguity not shared by 2-D displays. Each pixel on a 2-D display corresponds to only one location in a 2-D data set, while each pixel of a 3-D display corresponds to a line containing a potentially infinite number of points in a 3-D data set. Also, our eyes are better at comparing 2-D locations on a screen than they are at comparing depths depicted with binocular stereo. The most effective way to resolve both of these forms of 3-D ambiguity is to give the user the ability to change the view point in 3-D with immediate response. A 90-degree rotation changes depth information into more easily perceivable screen geometry information. Rotation also separates and creates relative motion between the 3-D points which share a pixel, resolving them to the user's eye.

The computational power needed for 3-D rotation and zoom can also be applied to other forms of interactive control over the display. These include control over the density of information, the value of contour surfaces, the selection of meteorological fields to be combined in the display, and parameters of the underlying analysis. These forms of interaction should be simultaneous with time animation. As the interactivity of 3-D displays increases, particularly in the area of real time 3-D rotation, the need for binocular stereo displays with their special glasses will diminish.

There are other forms of 3-D interaction which are less demanding computationally. First, real-time rotation and zoom can be applied to simple images such as line graphics. Second, complex 3-D images can be produced at less than real-time rates and stored for multiframe animation, similar to animation on the current McIDAS workstation. Third, a 3-D cursor can be displayed in 3-D images in way consistent with perspective, hidden surface removal, and transparency. Such a cursor can be used to retrieve values of meteorological parameters. It can also be used to interactively direct the creation of 3-D image elements, such as the placement of trajectory parcels at points of interest.

4. PROPOSED SYSTEM

The video system of the proposed workstation is the ATT Pixel Machine (Runyon, 1987). It contains up to 48 million bytes of frame memory and executes up to 820 million floating point operations per second. With this computational power, the Pixel Machine can provide real-time interactive rotation of animated 3-D images similar to those we currently produce on McIDAS (Hibbard, 1986). It can also provide the other forms of interactivity described in the previous section. Of course, this would require very complex and difficult software.

The Pixel Machine has no hardware pan or zoom. However, it has sufficient computational power to implement these functions in software at reasonable animation rates, as well as false color enhancement of more than eight bits per pixel. Its video output can generate separate red, green and blue channels with eight bits each. This mode will allow very high quality images with subtle use of color.

The Pixel Machine requires a conventional host with an operating system for running development software and coordinating the operation of its many DSP-32 processors. The only currently announced host is a Sun computer running the UNIX operating system. The Pixel Machine will be available in the first quarter of 1988.

The Pixar Image Computer (Levinthal, 1984) would be a good alternate candidate for our next video system. It contains up to 120 million bytes of memory and can execute 120 million integer operations per second. The Pixar has been available for a couple of years and has a developed software base. Although it has less power for real-time rotation of complex 3-D images, it provides a variety of real-time image processing functions. The Pixar also requires a host computer. Currently announced hosts include Sun, Symbolics, Silicon Graphics and DEC micro VAX II.

A great deal of software will be required to integrate the new video system and host as a McIDAS workstation. Most of this software will run in the host for data management, communications, and applications. Software for the video processor will implement lower level but computationally intensive tasks for interactive functions.

The costs of both the Pixel Machine and the Pixar are within range for a McIDAS workstation.

In order to satisfy our basic requirements and interactivity goals a video system must combine a large and flexible frame memory with a powerful and generally programmable processor where the processor has high bandwidth access to the memory. It is likely that several other candidate systems will appear during the next few years.

5. MANAGING VIDEO MEMORY

In the current McIDAS workstation, the allocation of video memory to frames is fixed, and the frames exactly fit the screen, with no pan or zoom transformation. The frames and other video system components are managed with a disk file called the frame directory and a memory region call system common (SYSCOM). The frame directory contains entries for a fixed number of frames, documenting the source (e.g., satellite ID), date and time of the frame data, the location and resolution of the frame within a larger image, and other information. Animation and interactive functions are managed by a TV control (TVCTRL) process. Because the frames exactly fit the screen, the TVCTRL process does not need access to the frame directory to control animation. It only needs access to SYSCOM, which contains the sequence of frame numbers defining the animation loop, along with dwell times for each frame.

In order to manage variable (and user definable) sized frames with pan and zoom, these structures need to be changed. The number of frame entries needs to be variable, and some new information needs to added to the entries, including the size of the frame, the address range of video memory used to store the frame data, a flag indicating whether the data are stored in packed format, and the number of bits per pixel for packed data. Animation control should include the ability to pan while animating, and the ability to keep earth locations fixed during animation of loops whose frames possess different sizes, locations and resolutions. These abilities require that the frame directory information for a loop be accessible in memory during animation. The frame directory information also needs to be stored redundantly on disk to survive a system reboot or crash.

Currently McIDAS has a hierarchy of three coordinate systems for image data. These are the image coordinates from the raw data source, area coordinates in the image file on disk, and frame coordinates in the video frame. The relation between these coordinate systems is defined by translation offsets and resolution blow-up or blow-down factors in both line and element directions. With variable sized frames this hierarchy will be extended by adding screen coordinates in the monitor screen. As with the other coordinate systems, the user will generally specify screen locations by earth latitude and longitude and let the system convert. During animation of a loop, the user will usually select the screen window into the frames by earth coordinates. It is interesting to consider uses for animation of loops whose frames differ in size, location and resolution. A forecaster might maintain a GOES loop covering a local region every half hour, and another loop covering a larger surrounding region every hour or two. To save frame space the local loop could share the larger frames at the times when the loops coincide, resulting in a loop whose frames vary in size, location and possibly resolution. As another example, a storm might move off the edge of the region covered by a standard loop. In order to continue following the storm, the user could define another loop including the frames of the standard loop and more recent frames with a translated location to continue covering the storm. The new loop would be animated with the screen centered to cover the storm over the duration of the loop.

The allocation of video memory into a variable number of variable sized pieces is similar to the allocation of segments of main memory in a computer system. This suggests the use of an object-based system similar to those used for managing memory segments, rather than the current flat file frame directory. Each frame would be documented by a frame object. An object-based system could also be used for those parts of SYSCOM relating to the video system. Loop objects would point to a series of frames. Several loop objects could be defined and given names, so that the user could easily switch among different loops. Objects could also be defined for color tables, cursors, windows and other video system resources. Frame objects would have names, although the user would usually refer to frames by the name of a containing loop and the sequence number of the frame in the loop, or by loop name and frame time. Of course, a frame could belong to several loops.

Access to objects in an object-based system is through a set of subroutines defining the legal operations on objects. This allows object formats and processing to be changed without changing applications. It also protects system integrity from applications programs. The current SYSCOM is divided into blocks of memory, and the words of each block are allocated to fixed uses. Although the blocks can grow by adding words onto the end, this structure tends towards having a fixed number of structures of a given type (e.g., cursors, windows, loops). However, an object-based system allows an arbitrary number of each type of structure. This is more adaptable to new video systems, as well as to new uses of existing video systems. An object-based system also makes it easier to add new types of structures.

Objects can be defined for 3-D structures. A 3-D frame object could be defined to include a pair of ordinary frames, one for a 3-D image and one for a depth z-buffer. It would also include parameters for the projection mapping from 3-D space to 2-D screen coordinates. Such a 3-D frame would support operations with a 3-D cursor, and also support incremental rendering of new information in the frame in response to user interaction. This type of 3-D frame would apply to 3-D frames containing only opaque objects. However, another type of 3-D frame could be defined which would allow transparent objects. Such 3-D frame object would include an image and a z-buffer for opaque objects, an image and a z-buffer for transparent objects, and a buffer of transparency (alpha) values. This would allow 3-D cursor operations and interactive control of rendering with transparent objects. A topography frame object could be described, which would contain surface heights at a grid of points. This could be used to render a shadow of a 3-D cursor on the topographical surface in a 3-D frame. Of course, these operations with 3-D frames and cursors require processing power close to the video memory.

Most of these design ideas would apply to any video system which allows variable sized frames.

6. MANAGING THE VIDEO PROCESSOR

All functions of the video system involve executing software on the video processor. These functions include frame loading, animation, panning, zooming, drawing graphics, color table loading and cursor movement. They also include interactive functions created by users with communicating software components in both the video processor and the host processor. The video processor must appear to execute these functions simultaneously. This is a new type of resource management problem for McIDAS.

The video processor should be managed as a queue server which executes functions requested by host processes. McIDAS applications will call a system service to request that video software be invoked. If the video processor is busy when an application calls the invoke service, the request will be placed on a queue. Requests will be removed from this queue and invoked when the video processor signals that it has completed the current function. A mechanism should be provided with this service to allow data to be passed in both directions between host and video processor. Other system services will be implemented so that host programs can explicitly load and unload video processor software.

By managing the video processor with a queue server, several simultaneous and apparently continuous video processes will be implemented with multiple continuous processes in the host which repeatedly invoke routines in the video processor. A system service should also be provided to temporarily disable the queue server, in order to accommodate very complex functions with software components in both host and video processor which communicate without the queue server mechanism.

An important function for the video processor will be to provide real-time animation and interaction with 3-D images, which presents some new management problems. For multi-frame animation, the frame objects for the stored frames are used to manage the animation. However, for real-time animation there are no stored frames and frame objects. Thus SYSCOM should include information for managing real-time animation. This would include 3-D world boundaries (e.g., latitude, longitude and height bounds), the parameters of the perspective mapping, time bounds and the time step for the animation, a list of the currently displayed meteorological fields and links back to their underlying data sets (stored in the host's memory or the video memory), and current analysis and information density parameters. User interaction with real-time animation will be managed by linking this information with user input through keyboard, mouse, joystick and other devices. Of course, user interaction can be simultaneous with time animation of the display.

While video processors are designed to provide fast rendering of images, they are not necessarily suited to the data analyses associated with rendering images, particularly for large meteorological data sets. This may pose barriers to real interactivity with the data. One approach to this problem is to restrict the analysis at any instant to a subset of the data set, and to use the speed gained by this to allow the user to roam around in the data. For example, 3-D and 4-D gridded data could be preprocessed to exist at its natural resolution and at several coarser resolutions. At any given instant, the world extents of the display could be used to pick a resolution of the grids with a tractable analysis problem. To see more detail, the user would zoom in closer to a region of the display, which would be accompanied by a switch to a sector of a finer resolution grid. Thus a continuous perspective zoom would be punctuated by discrete analysis zooms, increasing the level of detail in the display. In some cases there may be a slight lag of the analysis as the user zooms into a new region of the data. For example, after an analysis zoom, wind trajectories might gradually appear over a period of a couple seconds, concurrent with time animation and rotating perspective.

In order to understand this mode of interaction, it is worth describing how it could be controlled with a mouse. With neither mouse button pushed, lateral mouse motion would control horizontal image rotation and forward and backward mouse motion would control vertical image rotation. With the left mouse button pushed, mouse motions would control horizontal and vertical translation of the image on the screen. With the right mouse button pushed, forward and backward mouse motion would control zooming towards the point of the 3-D cursor. As the user zooms in, the world box appears to expand on the screen, but it is periodically replaced with a smaller box and a more detailed analysis. During a zoom out, this would be reversed. The mouse could also be used to control the 3-D cursor, although this should be kept segregated from mouse control of the image. One method would be to use a mouse button double click to switch modes between cursor and image control.

These design ideas would apply to any video system which provides a generally programmable and powerful video processor.

7. OTHER DESIGN CONSIDERATIONS

A new video system should make it possible to achieve must faster frame loads than currently, particularly when they do not involve a resolution change or sectorization from a very large image file. This applies particularly to frame loads from the workstation host to the video system, but it should also be possible to increase the speed of data movement from the IBM mainframe to the host. For example, the "burn box" communications system on McIDAS network of the Centralized Storm Information System at the NSSFC achieves data transfer rates of 1.2 Mbps, equivalent to moving a 480 by 640 by 8-bit frame in two or three seconds.

Fast frame loads would apply in both directions, so it should be possible for users to quickly save their frames, similar to the way they now save string tables and color tables, and to quickly restore them. Faster frame transfer rates will also be desireable to keep load times for larger frames within reason.

Data retrieval times for non-image data are also important. The time to generate graphics depends on the time to get the data. Grid data are usually stored in reasonably compact arrays which can be retrieved quickly. However, irregularly spaced data are often stored in a way that their retrieval requires as many disk reads as there are data points. Commercial data management systems based on the relational model of data provide independence of the applications software from the data storage format, as well as retrieval of data in unpredictable combinations. However, only those systems which provide control over the physical grouping of data are appropriate for use with interactive graphics. On McIDAS, the MD data management system provides data independence and physical grouping of data based on geometric retrieval patterns. Designers of interactive meteorological systems should take the lead in developing data management systems based on a geometrical model of data.

8. CONCLUSION

A next generation workstation should include capabilities for variable-sized frames, pan, zoom and fast frame loading which reduce response time for access to image data. However, it is equally important to recognize that a next generation workstation should also include a powerful processor close to the video memory in order to support greater interactivity of image processing, graphics generation and image rendering. This is particularly true for 4-D displays, which will become an integral part of a general meteorological workstation. A powerful and general processor in the workstation will improve the quality of 4-D images, increase their interactivity, and gradually reduce the need for binocular stereo and stereo glasses.

9. ACKNOWLEDGMENTS

I would like to thank D. Santek, P. Meyer, D. Wylie, J. Benson, R. Krauss, W. Smith, G. Dengel, R. Dengel, T. Whittaker, and J. T. Young for helpful discussions. This work was supported by the National Aeronautics and Space Administration (NAS8-36292).

10. REFERENCES

Hibbard, W., 1986: Computer-generated imagery for 4-D meteorological data. Bull. Amer. Met. Soc., 67, 1362-1369.

Levinthal, A., and T. Porter, 1984: Chap - a SIMD graphics processor. Computer Graphics, 18, 77-82.

Runyon, S., 1987: AT&T goes to 'warp speed' with its graphics engine. Electronics, July 23, 54-56.

Workshop on Graphics in Meteorology, ECMWF, Reading, England, 1988

Visualizing Weather Data

William Hibbard and David Santek
Space Science and Engineering Center
University of Wisconsin - Madison
1225 West Dayton Street
Madison, WI 53706
(608) 263-4427

Extended Abstract

Numerical weather models are producing enormous amounts of data. A typical model output data set may consist of one billion points in a five-dimensional rectangle, composed of a 100 by 100 horizontal grid, by 30 vertical levels, by 100 time steps, by 30 different parameters. Such a data set can fill thousands of two-dimensional plots, and challenges the capacity of current systems for its management, analysis and visualization.

The University of Wisconsin-Madison Space Science and Engineering Center (SSEC) developed the Man-computer Interactive Data Access System (McIDAS) as a tool for accessing weather data as animations of two-dimensional images and graphics (Smith, 1975). This system was a revolutionary change from the traditional paper facsimile plots, because it gave the user control over the geographic extents and contents of its images and graphics, and because it provided animation. However, numerical models and modern remote sensing instruments are generating data sets whose size, dimensionality and complexity exceed the capacities of current interactive weather systems. Thus we began an effort about six years ago to develop a four-dimensional capability for McIDAS. This system generates animated displays of multivariate three-dimensional images from weather data.

Our system manages weather data as two and three-dimensional uniform grids, as trajectory paths through space and time, and as images. Grids can be grouped together for multiple parameters and multiple time steps, allowing us to manage a five-dimensional rectangle of model output data as a whole. Our system includes a wide variety of user commands for housekeeping and analysis operations on our data structures. Gridded data can be resampled in space and in time, combined by arithmetic operations, transformed to a moving frame of reference and manually editted. Gridded data provides an indicator for missing data values, and this can be exploited to visualize only a region of a grid by marking other areas as missing. Trajectories can be generated from gridded U, V and W wind components.

The four-dimensional McIDAS system can be used to generate animation sequences of three-dimensional images which combine many physical paremeters, using a variety of graphical techniques (Hibbard, 1986a; Hibbard, 1986b). All of our images are rendered in a rectangular box, including a flat or topographical map on the bottom, with optional physical and political boundaries. Our graphical primitives include:

- ** opaque and transparent trajectories, rendered as thin cylinders
- ** opaque and transparent contour surfaces
- ** grid mesh contour surfaces
- ** contour isopleth lines lying either on the topographical map or on a horizontal surface defined by some other gridded variable such as pressure or potential temperature
- ** volume rendered densities, with the opacity of each three-dimensional point proportional to some gridded variable
- ** opaque and transparent texture mapped images, lying on some plane in space
- ** visible satellite images texture mapped onto a cloud top surface defined by the corresponding satellite infrared derived heights

We often combine our animation sequences with a rocking motion of about 1 or 2 degrees, as a way of making the three-dimensional geometry of the depicted weather phenomena more apparent (Hibbard, Santek and Dengel, 1988). We also sometimes stop the time action and rotate the images through large angles.

We have applied our software to produce visualizations of data fro many different numerical weather models (Pauley, Hibbard and Santek, 1988; Meyer and Seablom, 1988; Uccellini, 1988; Santek, Leslie, Goodman Diak and Callan, 1987). We have also applied our software to a wide variety of remote sensed data from satellites, radars, lidars, observin networks and sounding instruments. These visualizations are used for teaching at the University of Wisconsin and other institutions, for conference presentations of case studies, and by modellers and instrument developers wishing to understand their data.

Our experience producing visualizations of weather data has shown that a more interactive system is very important (Hibbard, 1988). We are currenlty developing a new system based on the Stellar GS-1000 graphics supercomputer. Most of our images contain between 5000 and 20000 shaded triangles, and roughly the same number of line vectors. Our animations generally run at 7.5 frames per second. The GS-1000 should be able to produce similar animations in real-time, giving the user interactive control over the display for rotation, zooming, changing the combination of selected parameters, retrieving values at a cursor, changing the levels of contour surfaces, and varying the images in many other ways.

ACKNOWLEDGMENTS

We would like to thank V. Suomi, F. Bretherton, R. Fox, J. T. Young and R. Krauss of SSEC, and J. Dodge, G. Wilson, J. Arnold and

P. Meyer of NASA, for their help and encouragement. This work was supported by NASA (NAS8-33799 and NAS8-36292).

REFERENCES

Hibbard, W., 1986a; 4-D display of meteorological data. Proceedings, 1986 Workshop on Interactive 3D Graphics. Chapel Hill, SIGGRAPH, 23-36.

Hibbard, W., 1986b; Computer generated imagery for 4-D meteorological data. Bull. Amer. Met. Soc., 67, 1362-1369.

Hibbard, W., 1988; A next generation McIDAS workstation. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Anaheim, American Meteorology Society, 57-61.

Hibbard, W., D. Santek, and G. Dengel, 1988; Visualization of Four-dimensional Meteorological Data. SIGGRAPH Video Review #37. ACM SIGGRAPH.

Meyer, P. J., and M. S. Seablom, 1988; Application of the four-dimensional McIDAS to LAMPS model output. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Anaheim, American Meteorology Society, 33-36.

Pauley, P. M., W. Hibbard, and D. Santek, 1988; The use of 4-D graphics in teaching synoptic meteorology. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Anaheim, American Meteorology Society, 33-36.

Santek, D., L. Leslie, B. Goodman, G. Diak, and G. Callan, 1987; 4-D techniques for evaluation of atmospheric model forecasts. Proceedings, Digital Image Processing and Visual Communications Technologies in Meteorology. Cambridge, SPIE, 75-77.

Smith, E., 1975; The McIDAS system. IEEE Trans. Geosci. Electron. GE-13, 123-136.

Uccellini, L., 1988; Processes Contributing to the Rapid Development of Extratropical Cyclones. Preprints, Palmen Memorial Symposium on Extratropical Cyclones. Helsinki, Finland, Amer. Met. Soc., 110-115.

Proceedings, 5th Interactive. Info. and Processing Systems for Met., Ocean., and Hydro., Anaheim, AMS,1989

5.12

4-D DISPLAY OF GEOHYDROLOGICAL MODEL RESULTS

BY:

David A. Santek, William L. Hibbard,
Space Science and Engineering Center
1225 W. Dayton Street
Madison, Wisconsin 53706

and

Kenneth J. Quinn, Timothy E. Melka
Warzyn Engineering Inc.
1 Science Court
Madison, Wisconsin 53711

INTRODUCTION

The analysis of geohydrological data has traditionally been through the use of 2-D contour plots and vertical cross sections. This is the most natural because of the way the data are gathered and traditional display mechanisms. Although there has been an increased use of numerical models to simulate flow systems, the output presentation methods have not changed to handle the new and often voluminous amounts of data. A simple 2-D model may generate hundreds of 2-D fields that can be displayed, or contoured, on a graphics system or in a hardcopy form. Traditional methods for analysis may be appropriate, but the greater amount of data makes it more difficult. Taking it one step further to 3-D, the complexity increases by more than the additional dimension because of interactions and relationships between all three space dimensions and time. It is unfortunate that the output is often in a 2-D form, usually vertical cross-sections for selected transects or horizontal slices. It is up to the scientist or engineer to mentally integrate the information to gain a better understanding of the 4-D characteristics of water flow. Not only is this task tedious, but the results of the analysis will probably be incomplete. It behooves us to investigate ways to display the model output as a continuum in space and time to better interpret a complex flow system.

The geohydrologic model results presented in this paper are from a 3-dimensional transient groundwater flow model. Warzyn's analysis and presentation of 3-D transient groundwater flow model input and results uses typical contour maps (e.g. Figures 1 and 2). Display of the 3-D flow field as it changed through time was needed to:

- clearly illustrate the vertical position of the horizontal flow relative to clay layers separating sand and gravel aquifers, and
- present results to non-technical clients and the public.

A means to display meteorological data in 4-D has been developed at the Space Science and Engineering Center (SSEC) on the Man computer Data Access System (McIDAS) (Hibbard, et. al., 1987). The flexibility designed into the data management and display programs, allowed for a relatively easy extension for use with hydrological data. There are similar vector and scalar fields between hydrology and meteorology, but the time and space scales differ vastly. For instance, a 40-year groundwater flow simulation was stored in the 4-D database as through it took only 10 hours. Similar adjustments were used for space scales. Application of McIDAS to subsurface hydrology has been initiated on two projects where 4-D presentation of the data enables a better understanding of the interaction between the subsurface conditions and the groundwater hydrology. A still photo example of the McIDAS display is shown in Figure 3.

Through cooperation of two of the authors, Warzyn's data used by and output from a groundwater hydrology model were transferred to the McIDAS and displayed in 3-D animated sequence. The purpose of this paper is to present the results of this technology exchange and highlight the advantages of a 4-dimensional display capability. The paper will present a brief background on each software used, the hydrologic problem set used and finally, the results in both 2-D and 4-D presentations.

MCIDAS 4-D SOFTWARE

The McIDAS manages and processes satellite and weather data for displays of animated satellite images with overlayed graphics (Suomi, et. al., 1983). McIDAS runs on an IBM 4381 and uses graphic workstations built by SSEC. The 4-D software, a small part of the McIDAS, was developed to manage and display 4-D meteorological data, and has been applied to numerous data sets of varying time and space scales. The data are stored in 2-D grids, 3-D grids and trajectory files (parcel location with time). One rendering program operates on the grid and trajectory files, regardless of space or time scales, and generates 3-D images. The images may consist of any or all of the following components for depicting different types of physical parameters: transparent or opaque objects, transparent fog, wire mesh, 2-D contours, and tubes (Hibbard, 1986). For this geohydrological case, clay layers are shown as a transparent or color opaque object, within the clear matrix of the aquifer, where groundwater can easily flow. Trajectories appear as moving tubes. These objects are drawn within a rectangular box that defines the area of interest in 3 space. To heighten the 3-D effect, the use of hidden surface removal, shading, stereo viewing and animation are used. Included in some animated sequences is a slight rocking of the objects (but not the outer box) to give additional depth information.

FLOW MODEL AND PARTICLE TRACKER

The groundwater flow model(MODFLOW) used to simulate changes in head due to changes in water supply well pumping rates was written by McDonald and Harbough (1984), of the U.S. Geological Survey. The model solves the groundwater flow, partial differen- tial equation given below, using a finite difference approximation.

$$\frac{\partial(K_x\partial h/2x)}{\partial x} + \frac{\partial(K_y\partial h/\partial y)}{\partial y} + \frac{\partial(K_z\partial h/\partial z)}{\partial z} = S_s\frac{\partial h}{\partial t}+W$$

where X, Y, Z = spatial cartesian coordinates (L)
K = hydraulic conductivity of the saturated porous media (L^2/T)
S_s = specific storage (L^{-1})
t = time (T)
W = volumetric flux (L^3/T)
h = potentiometric head (L)

The model uses a block centered finite difference grid for setup of the problem domain and the strongly implicit procedure for solution. Data input includes aquifer parameter matrices (hydraulic conductivity and physical description of the geologic distributions), initial conditions (starting head) and boundary conditions (ie. pumping rates and locations, head or flow conditions on model boundaries, rainfall recharge rates, river and lake locations, head and connection with the aquifer). The finite difference grid in the horizontal, X-Y plane is uniform although not necessarily of equal spacing. Each vertical grid layer can be deformed to follow the irregular shape of the aquifer or clay layers in the subsurface. Each model layer for this well field problem is highly deformed to follow the variable thickness of the clay layers and aquifers.

Results from MODFLOW consist of a matrix of head values for each layer, for each finite time step of the model. Actual flow directions in this heterogeneous multi-layer simulation cannot be determined based on the head maps alone (e.g. Figures 1 and 2). A velocity field processor is required to determine actual 3-dimensional groundwater flow lines at any one instant or the resultant flow path through time.

A particle tracker (PATH3D), developed by Chunmiao Zheng (in publication by the Wisconsin Geological and Natural History Survey), was in the transient mode to map the flow lines through the modeled area based on MODFLOW output. PATH3D uses the head matrices from selected time steps in MODFLOW, with selected input files used to initiate the MODFLOW run. Two minor additions are required, the thickness and porosity of each layer (to compute travel time) and particle starting points in space and time. PATH3D moves each particle in response to the advection velocities interpolated between flow model cells for the duration of the time step. The next series of head matrices are then read, new velocities computed and particles moved. Output from the particle tracker model consists of the X, Y, Z and time positions of each particle. Graphical presentation of output is left up to the user.

Both the MODFLOW and PATH3D programs are run on a Micro-VAX computer at Warzyn. Run times for a flow model are approximately 186 CPU minutes for a transient (60 time step) simulation. PATH 3-D requires approximately 5.5 CPU minutes for a, 60 step transient, 10-particle analysis.

GROUNDWATER HYDROLOGY CASE

The example case for this paper is groundwater flow in the vicinity of a municipal (W series wells in Figure 1) and industrial well (F series wells) field in the midwest. Several wells within each well field were contaminated by trichloroethylene and/or tetra-chloroethylene. An extensive field investigation is in process to characterize the geology, the groundwater flow system and determine the extent of contamination within each aquifer. The groundwater modeling is being conducted concurrent with the data gathering so as to help quantify results of the field investigation, direct further investigations to identify critical areas of additional data needs and finally to evaluate remedial actions. Therefore, the model results presented here are not fully calibrated and verified to observed conditions. This paper is intended to present results of the geologic investigation and the groundwater flow modeling using the McIDAS display techniques.

The aquifer utilized by the municipal and industrial wells is a thick sequence (greater than 260 ft) of three sand and gravel outwash deposits separated by two clay till deposits. Each of the sand and gravel outwash deposits are laterally extensive, although varying in thickness from 20 to 70 ft. The supply wells are screened in, and pump from, the lowest sand and gravel deposit. The two clay till deposits separating the aquifers are not present throughout the entire modeled area. These two till deposits are displayed on the McIDAS display as transparent surfaces. The lower till is present only in the west portion of the model area (see Figure 3) as a finger extending from the northwest to the center of the model. The upper till is present across the south and west ends of the model. The relationship becomes much clearer when viewing the video tape presented at the conference.

During the field investigation contaminants have been found above the clay layers close to potential source areas. In the center of the city contaminants have been found at depth (greater than 150 ft below the water table). This raises the concern that the contaminants may have moved as a dense non aqueous phase liquid, sinking to the bottom of the aquifer.

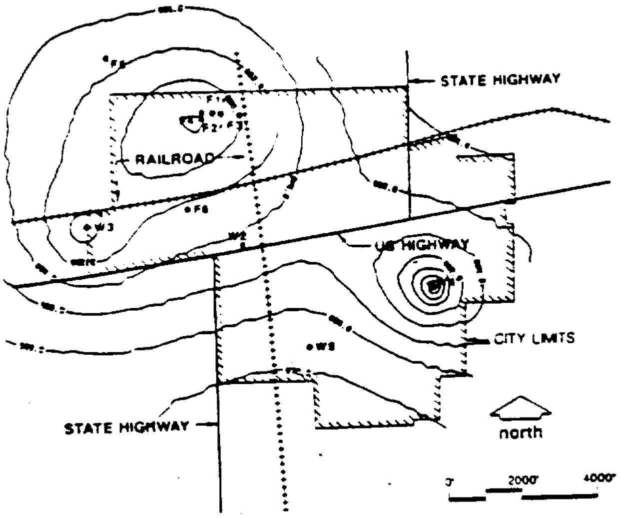


Figure 1 - Contour map of groundwater head in layer 1 - contour map of uppermost sand and gravel deposit (0.7m contour interval).

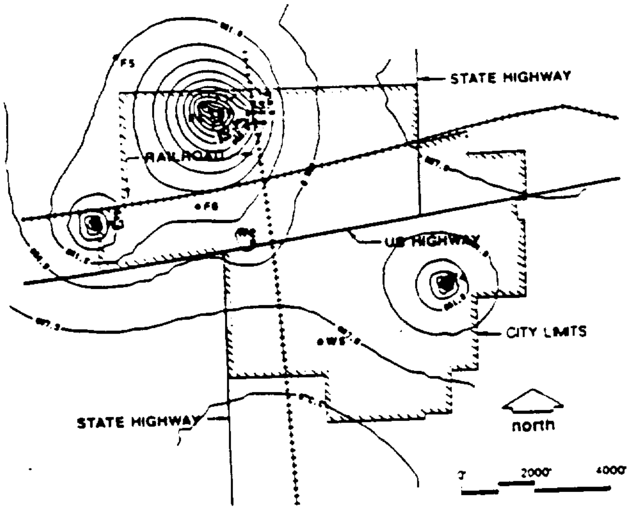


Figure 2 - Contour map of groundwater head in layer 5 - bottom sand and gravel deposit. Aquifer being pumped by the supply wells (0.9m contour interval).

The McIDAS display (Figure 3) illustrates the strong control that the clay deposits exert on the groundwater flow system that horizontal projection of the flow lines (see base of Figure 3) cannot show. Flow lines starting at the water table are relatively horizontal above a clay deposit. As the flow line reaches the end of clay deposit the small downward vertical gradient through the permeable sand and gravel results in very strong downward flow just off the edge of a clay deposit. This illustrates that contaminants can move to great depths off the edge of a clay deposit. This discounts the potential presence of a pool of highly concentrated contaminants acting under a density gradient.

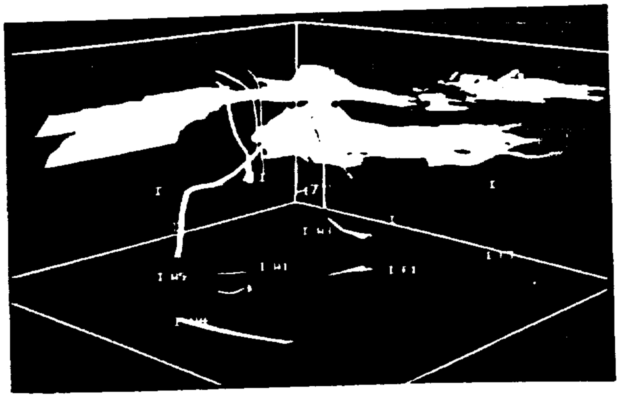


Figure 3 - McIDAS 3-D display with 2-D projection of flow lines on base. White surface are the top and bottom of clay deposits. The I symbol shows the pumping well, location. View from the east.

SUMMARY

McIDAS has definite advantages over 2-D methods for display of the complex geology to clients or management, allowing even the uninitiated an ability to grasp the conditions. Figure 3 illustrates a true 3-dimensional picture of the clay layers distribution. Rotating the display provides different views illustrating specific relationships.

McIDAS's display of groundwater flow in 4-D allows further analysis and a better understanding of the interaction between the geology and groundwater flow system. Utilization of McIDAS for the case study presented here, provided insight into the problem of how a contaminant moved 100 ft vertically down in an aquifer, when vertical downward gradients were small. This may reduce the need for additional, expensive field investigations.

ACKNOWLEDGEMENTS

Work on McIDAS for this project was supported by the National Aeronautics and Space Administration (NAS8-33799 and NAS8-36292). The groundwater modeling and cooperatio nof Warzyn with SSEC was funded by Warzyn Engineering.

REFERENCES

Hibbard, W., 1986: Computer Generated Imagery for 4-D Meteorological Data. Bull. Amer. Meteor. Soc., 67, 1362-1369.

Hibbard, W., R. Krauss, D. Santek, and J.T. Young, 1987: 4-D Display of Weather Data on McIDAS. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. New Orleans, Amer. Meteor. Soc., 89-93.

McDonald, M.G. and A.W. Harbough: A Modular Three-Dimensional Finite Difference Groundwater Flow Model. USGS. 1984.

Suomi, V. E., R. Fox, S. S. Limaye, and W. L. Smith, 1983: McIDAS III: A Modern Interactive Data Access and Analysis System. J. Clim. and Appl. Meteor., 22, 766-778.

Zheng, Chunmiao, in Press, A Three-Dimensional Particle Tracking Model for Calculating Groundwater Paths and Travel Times. Wisconsin Geological and Natural History Survey.

Proceedings, 1989 5th Interactive Info. and Processing Systems for Met., Ocean., and Hydro, Anaheim, AMS
5.20

VISUALIZING LARGE DATA SETS

William L. Hibbard and David A. Santek

Space Science and Engineering Center
University of Wisconsin - Madison
Madison, Wisconsin

1. INTRODUCTION

Remote sensing instruments and numerical models are producing large data sets which are difficult to manage, analyze and visualize. A lidar can produce a data set of 1.5 billion points, with 2000 samples per ray, a volume scan of 60 by 60 rays, and 200 volume scans over eight hours of observing. A weather model may produce 1 billion points in a five-dimensional rectangle, composed of a 100 by 100 horizontal grid, by 30 vertical levels, by 100 time steps, by 30 different parameters. An Earth Observing System (EOS) era instrument may produce 10 million observations per second.

Such huge data sets defy our traditional techniques for managing, analyzing and visualizing them. Thousands of two-dimensional plots may be required for a billion point weather model data set. And these plots may not adequately depict phenomena which weave through many horizontal planes and many vertical cross sections.

We are attempting to address this problem, with our current IBM mainframe-based system and especially with our next generation interactive system based on a graphics supercomputer workstation.

2. CURRENT SYSTEM

We are currently handling large data sets using the four-dimensional McIDAS software running on the IBM mainframe McIDAS. Using this system we have produced videotapes which are effective for visualizing large case study data sets.

Our software provides tools for managing large data sets as two- and three-dimensional uniform grids, as streamlines, as trajectories, and as images. The software has a consistent set of conventions for dealing with groupings of these structures by different parameters at a given time, and by time steps over a range of times. Thus three-dimensional grids grouped for both parameters and times comprise a five-dimensional rectangle such as a weather model might produce. Trajectories are managed in groups which indicate the motions of many air parcels throughout a four-dimensional space and time region. Our data management software includes user commands for listing and manipulating data sets, subroutine libraries to give analysis programs access to data, and tape read programs for getting data into the system.

McIDAS and its four-dimensional component provide a wide variety of data analysis functions. These operate on images, grids, trajectories and other data structures for smoothing, interpolation advection, simple arithmetic combinations, and many specialized analysis techniques. These analysis programs access the data through the data management libraries.

Our system includes a large body of software for generating three-dimensional images. These programs generate:

** shaded relief topographical maps with physical and political boundaries
** trajectories as shaded tubes
** grid mesh contour surfaces for parameters such as mixing ratio and potential temperature
** shaded contour surfaces for any scalar parameter, which may be either opaque or transparent
** contour lines, drawn either on the topographical surface or on a surface in the atmosphere (such as a pressure or theta surface)
** two-dimensional image slices in the atmosphere, such as those scanned by radar or lidar
** three-dimensional transparent fogs with opacity proportional to some scalar parameter
** cloud top surfaces generated from visible and infrared satellite images

These images can be rendered as grey shades or as color. Although the McIDAS workstation has only pseudocolor and not true color, we have been able to combine color and transparency effects. This combination requires a two-dimensional color space, which we were able to implement using the McIDAS 12-bit color table combining inputs from two image channels simultaneously. In order to produce accurate colors, we had to modify the PROMS in the workstation which implement the McIDAS encoding of 64 image intensities into 3 bits.

Although the images produced by McIDAS can be viewed directly on the workstation, we have concentrated on producing videotapes of our image animations. This is partly to reach more people with these images, but also because the workstation capacity of 128 images limits our ability to visualize a large data set. We produced a videotape from model simulations of the Presidents' Day storm with Louis Uccellini, which contains thousands of McIDAS images. It takes several days to load all these images into the workstation, so direct viewing on the workstation would not be a practical way to visualize this large data set.

Producing animation sequences of our three-dimensional images requires that the image generation programs are invoked for each image. For some cases, each image may require that a series of rendering programs be run. Thus the production of an animation sequence can be quite complex. We have developed some standard practices for managing these movie-making jobs, using the McIDAS DUO command. DUO invokes a set of other commands repeatedly, changing the command parameters for each invocation. Often these jobs involve mixtures of data management, data analysis and image generation commands. They may also require hours of CPU time, despite our best efforts to make our software efficient.

Three-dimensional images suffer from the inherent ambiguity of mapping three dimensions onto a two-dimensional plane. The right approach to resolving this ambiguity is to produce the interactive system we describe below, which will allow the user to rotate the image in real-time under joystick control. For our videotape productions, however, we have developed a compromise solution. We rock the images slightly with time animation. This is effectively illustrated in the animations we produced from Robert Schlesinger's cloud model data. The development of the storm is accompanied by a vertical rocking of about two degrees. Comparing the same sequence with and without the rocking shows a dramatic difference in depth perception. We also experimented with combining time animation with a steady rotation, but this produced ambiguity between the apparent motion of objects due to dynamics and due to rotation. The rocking is not ambiguous with the relatively steady dynamic motions of the atmosphere, and would only be ambiguous with rocking or oscillating motions of the atmosphere. For some of our animations we also stop the time dynamics to show a rotation of a stop action view of the atmosphere. This can be very effective in visualizing three-dimensional weather geometry at a single time.

A videotape intended for visualizing a large data set may require many animation sequences to see different parameter combinations, different scales, and different viewpoints. Thus a good videotape requires detailed planning reflected in a script. The script lists the different ways of viewing the data set, their sequence and the length of time each runs. Because of the amount of effort required to produce a videotape from a large data set using our current system, the script planning process is essential to conserve computer and human resources.

A videotape is produced by dubbing individual animation sequences onto tape, and then editing them together according to the script. The individual sequences are loaded into the multiframe McIDAS workstation and animated using the McIDAS animation control commands. For a very large data set, such as Greg Tripoli's RAMS cloud simulation, an individual sequence is too long to fit in the McIDAS workstation. In that case, the editing involves combining partial sequences into one long smooth animation.

We have used the current system to produce effective visualizations of large data sets on videotape. However, the time and effort required for these productions is enormous. Several hours of computer time are often needed to render an animation sequence and load it into the workstation. Producing an effective animation requires a lot of trial and error with choice of perspective point, combination of parameters, contour levels, colors, degree of transparency, map scale, etc. Thus each animation sequence may take days or weeks to produce, and a whole tape can take months.

3. INTERACTIVE SYSTEM

We are developing an interactive system based on a graphics supercomputer. This system will produce a quantum leap in our ability to manage, analyze and visualize large data sets. The basic ideas for it are described in our paper last year to this conference (Hibbard, 1988). The hardware candidates are the Stellar GS-1000, the Ardent Titan, the Alliant/Raster Technologies GX4000, the Silicon Graphics 4D/90 GTX, and the AT&T Pixel Machine. The Apollo DN10000 will become a candidate when a true rendering engine is integrated into its design, which is expected in late 1988 or early 1989. Each of these systems is capable of storing large data sets and generating real-time multivariate three-dimensional animations from them.

In order to interactively analyze and visualize data, it must be quickly accessible. Thus it is necessary to store large data sets in main memory rather than disk to get the necessary access speed. Transfer rates from modern disk drives are measured as a few million bytes per second, while access transfer rates for the main memories of graphics supercomputers are measured as hundreds of millions of bytes per second. Furthermore, the memory capacities of several of these machines are 128 million bytes, with 512 million byte machines expected within a year. These capacities are large enough to store significant subsets of current data sets for real-time interactive analysis and visualization. Of course, we will have to modify our data management software so data can be stored both in main memory and on disk, and can be moved between these media in response to the needs for access.

We will also need to extend the McIDAS data management to allow users to refer to large data groupings and apply analyses to a whole data set with a single command. Thus the user should be able to refer to a single image or to a time sequence of images with equal ease. It should also be possible to manipulate a single grid, or a

PRECEDING PAGE BLANK NOT FILMED

group of grids for different parameters at a single time, or a group of grids for different parameters and at a sequence of times, all with a single command. A principle goal of our planned system is to allow the user to interact with a large data set at the pace of his own thoughts. This will only be effective if the user is not bogged down in entering a separate command for each element of the data set.

Our proposed interactive system will produce images similar to the ones we currently produce on the IBM mainframe. However, they will be rendered in a tenth or a fifth of a second, allowing them to be rendered as they are animated. This will give the user immediate control over their rotation, zoom and combination of parameters via mouse or joystick. It will also be possible to interactively control the levels of contour surfaces, trajectory placement, information density and smoothing. Thus the user will be able to understand a complex multivariate four-dimensional data set quickly. When the user sees an interesting phenomenon in one animation sequence, he will be able to pursue its cause by displaying it in relation to other parameters, viewing it from another angle, or taking a closer look with a different sampling or smoothing of the data.

4. CONCLUSION

Our experience producing graphics animations from very large data sets has convinced us more than ever that interactivity is the key to making four-dimensional data management, analysis and visualization work. In the past we have conducted a variety of experiments with stereo displays. Although stereo produces an interesting sensation and does provide some depth information, it is certainly not the answer to the serious problem of managing and visualizing large data sets. A highly interactive system is the answer.

5. ACKNOWLEDGEMENTS

We would like to thank L. Uccellini, P. Pauley, G. Tripoli, R. Schlesinger, R. Kropfli, E. Eloranta, K. Brill, J. Nepstad, C. Ziegler, K. Quinn, D. Perkey, G. Diak, G. Callan, H. Woolf, E. Brandes, and A. Bellon for data and help designing visualizations. We would also like to thank P. Meyer, J. Arnold, G. Wilson and J. Dodge of the National Aeronautics and Space Administration for their continuing advice and support (NAS8-36292).

6. REFERENCES

Hibbard, W., 1988: A next generation McIDAS workstation. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Anaheim, American Meteorology Society, 57-61.

Visualizing Large Data Sets in the Earth Sciences

William Hibbard and David Santek
University of Wisconsin-Madison

With single data sets containing billions of points, meteorologists and other earth scientists face an avalanche of data received from their remote-sensing instruments and numerical simulation models.

The Space Science and Engineering Center (SSEC) at the University of Wisconsin-Madison led the evolution of weather visualization systems from paper to electronic displays with our Man-computer Interactive Data Access System (McIDAS). We have continued this evolution into animated three-dimensional images and recently into highly interactive displays. Our software can manage, analyze, and visualize large data sets that span many physical variables (such as temperature, pressure, humidity, and wind speed), as well as time and three spatial dimensions.[1,2]

McIDAS has produced three-dimensional animations of data sets from many diverse sources on videotape and on a special binocular stereo workstation. These animations are used in the classroom at the University of Wisconsin and other institutions, and by visiting scientists presenting case studies at conferences. Model developers are using McIDAS to produce highly interactive, real-time animations of the dynamics of their models.

McIDAS, an interactive visualization system, is revolutionizing the ability of earth scientists to manage and analyze data from remote sensing instruments and numerical simulation models.

Data management

The McIDAS system manages data from at least 100 different sources, a diversity illustrating the flexibility of our data management tools. These tools consist of data structures for storing different data types in files, libraries of routines for accessing these data structures, system commands for performing housekeeping functions on the data files, and reformatting programs for converting external data to our data structures.

McIDAS includes data structures for grids, images, paths, and nonuniform data. Two- and three-dimensional grids are spatial arrays of numbers appropriate to numerical weather-model output and analyses of remote sensed data. Our grid structures permit grouping a range of times and multiple physical variables in a data set of billions of points, allow a variety of horizontal and vertical map projections, and indicate missing data points. Images are two-dimensional arrays of pixel intensities produced by satellites, radars, lidars (laser radars), and other sources. These images may include multiple spectral bands and may be grouped into time sequences, possibly containing billions of points. Paths are sequences of points through space or space-time, usually representing trajectories of air parcels derived from wind data. Nonuniform data consist of numbers without any spatial order that might be produced by surface observations, balloons, ships, and aircraft.

The McIDAS system provides numerous analysis functions for time and space interpolation of data, for deriving grids from nonuniform data, for deriving trajectories from wind grids, for converting data to a moving frame of reference, for making measurements from image data, and for applying a large variety of operators to grid and image data. (As part of the development of our three-dimensional visualization tools, we have inventoried four-dimensional data sets in meteorology and related earth sciences. A copy of this inventory is available at cost by writing to the librarian of SSEC.)

0018-9162/89/0800-0053$01.00 ©1989 IEEE

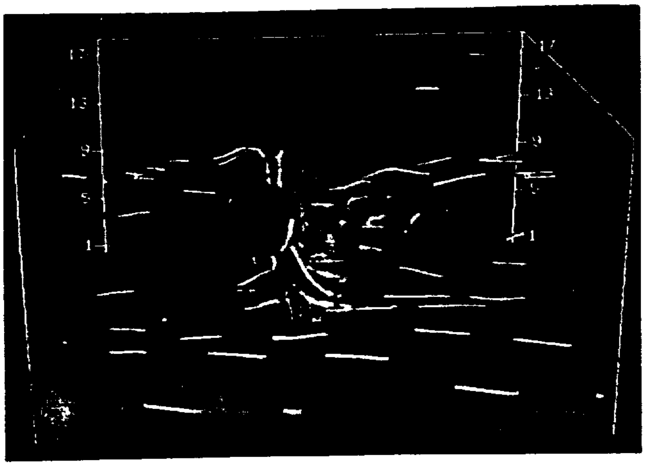


Figure 1. Regional Atmospheric Modeling System (RAMS) cloud model of a severe thunderstorm, showing a transparent 1.0-gram-per-cubic-meter condensate surface, a red 4.0-gram-per-cubic-meter hail surface, a blue 2.0-gram-per-cubic-meter rain surface, and wind trajectories. (With Gregory Tripoli, Univ. of Wisconsin Meteorology Dept.)

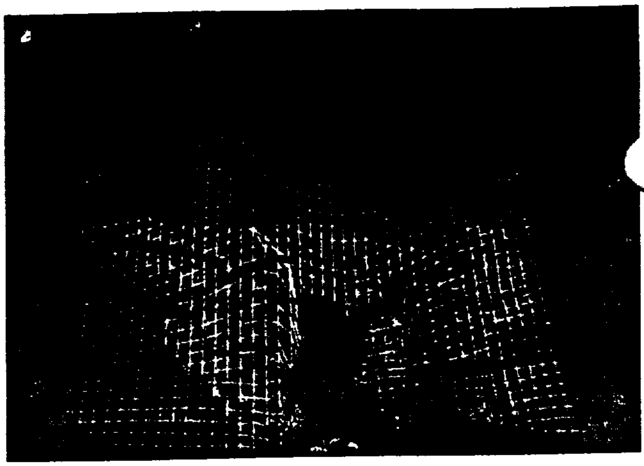


Figure 2. Limited-Area Mesoscale Prediction System (LAMPS) model output showing a grid-mesh 300-kelvin potential temperature surface, a transparent 8-gram-per-kilogram mixing-ratio moisture surface, and wind trajectories over a topographical map. (With Patricia Pauley, Univ. of Wisconsin Meteorology Dept.)

Figure 3. LAMPS model output for cloud water density over a topographical map. Volume is rendered as a transparent fog, with opacity proportional to cloud water density. (With Patricia Pauley, Univ. of Wisconsin Meteorology Dept.)

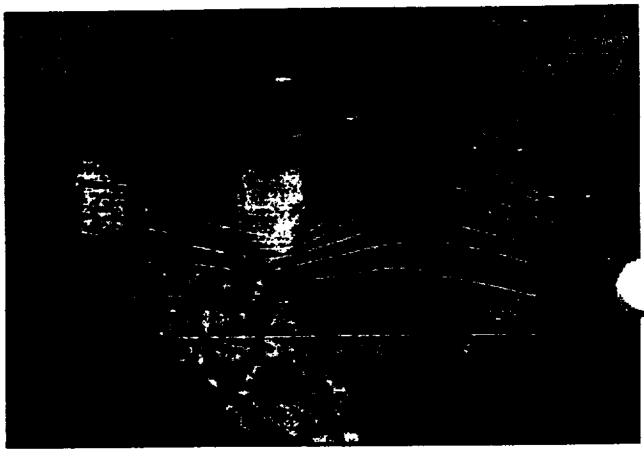


Figure 4. LAMPS model output showing a transparent 0.00016-per-second vorticity surface, trajectories, and 300-millibar height contour lines over a topographical map. (With Patricia Pauley, Univ. of Wisconsin Meteorology Dept.)

Three-dimensional visualization

The McIDAS tools for three-dimensional visualization of meteorological data run on an IBM mainframe and can load up to 128-frame animation sequences into workstations built at SSEC. The animations can be viewed directly (optionally, in binocular stereo) or recorded on videotape.

Our data are depicted in a rectangular box outline with height tick marks labeled in kilometers. A map, including topographical relief and boundary lines, can be drawn on the bottom of the box. The box visually defines the three-dimensional space, the map provides geographical context, and the topography is useful because it affects the weather.

We can depict gridded three-dimensional scalars using opaque or transparent contour surfaces (Figure 1) or a grid-mesh contour surface (Figure 2). A contour surface defines a constant value for a scalar variable. McIDAS can also depict a scalar as a transparent fog (Figure 3), where fog opacity is proportional to any scalar variable. The system generates contour lines (Figure 4) on a two-dimensional slice through a three-dimensional gridded scalar.

Wind vectors can be depicted using trajectory paths, which can be long and tapered (Figure 5), short with length proportional to speed (Figure 6), or faded to transparency (Figure 7). McIDAS also represents winds with derived scalars, such as vorticity, contoured in Figure 4, or potential vorticity, contoured in Figure 5. Figure 8 shows depth-cued trajectory paths adapted to depict a network of underground caves.

McIDAS can texture map image data to surfaces in three-dimensional perspective, as shown in the two lidar images in Figure 9. In Figure 10, we have texture-mapped a visible satellite image to a surface defined by the corresponding infrared image. P tern recognition discriminates clouds, an the visible pixels are blended with an artificial shade based on cloud-top geometry.

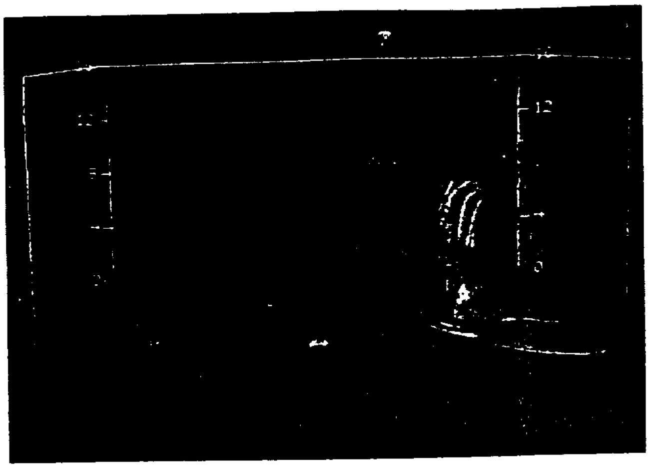


Figure 5. Mesoscale Analysis and Simulation System (MASS) model simulation of the Presidents' Day storm, showing a transparent 0.00002-kelvin per millibar per second potential vorticity surface and trajectories over a topographical map. (With Louis Uccellini, NASA Goddard Space Flight Center)

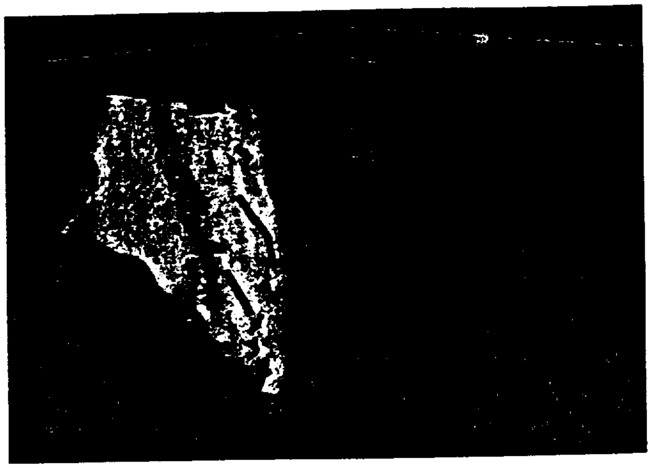


Figure 6. A microdownburst sensed by multiple doppler radars, with a transparent 30-DBZ radar reflectivity surface and wind trajectories. (With Robert Kropfli, NOAA Environmental Research Lab.)

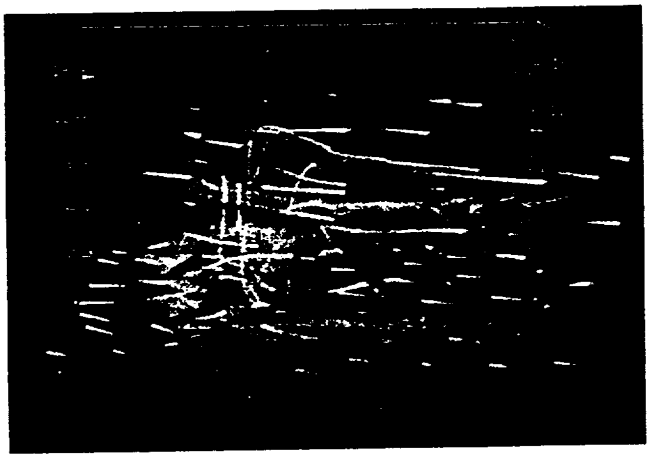


Figure 7. Thunderstorm simulation showing a transparent 0.5-gram-per-cubic-meter cloud water surface and trajectories. (With Robert Schlesinger, Univ. of Wisconsin Meteorology Dept.)

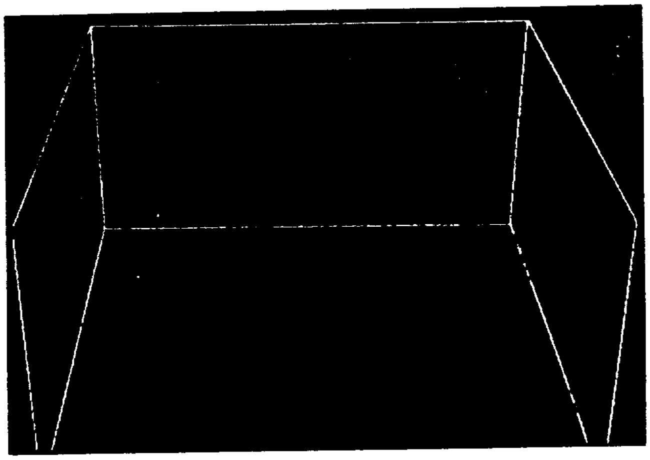


Figure 8. The Wind Cave network from data gathered by manual survey. (With Jem Nepstad, US Nat'l Park Service)

ORIGINAL PAGE IS
OF POOR QUALITY

Figure 9. Cirrus cloud section by two vertical lidar (laser radar) slices. With Ed Elorant, Univ. of Wisconsin Meteorology Dept.)

Figure 10. Perspective image of clouds over the Gulf of Mexico, generated from Geostationary Operational Environmental Satellite (GOES) infrared and visible data.

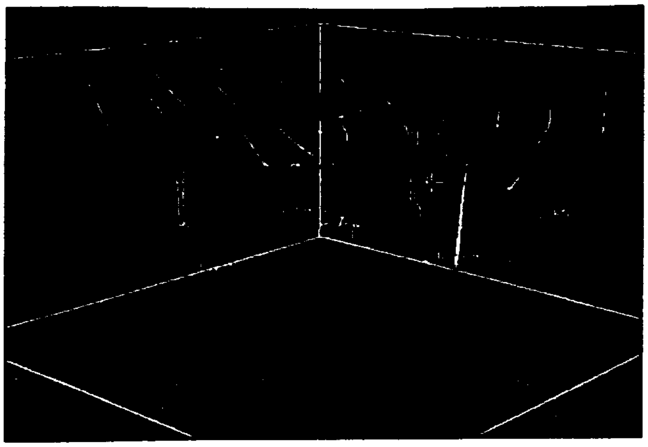


Figure 11. Hydrological model output showing transparent clay layers, ground-water trajectories, and labeled well positions. (With Kenneth Quinn, Warzyn Engineering)

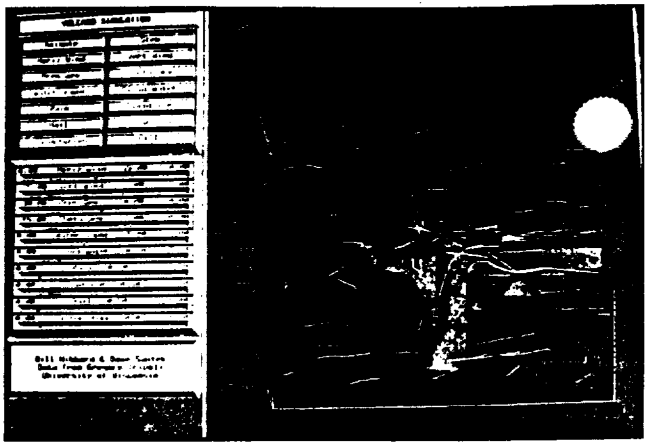


Figure 12. RAMS model of the atmospheric effects of a volcanic eruption, showing a green 0.07-gram-per-cubic-meter sulfur dioxide surface, a red 4.0-gram-per-cubic-meter water vapor surface, a blue 12.0-kelvin potential temperature deviation surface, and wind trajectories. The widgets to the left of the image control the display. (With Gregory Tripoli, Univ. of Wisconsin Meteorology Dept.)

Our images also show the interactions between physical quantities. Figure 1 shows contour surfaces for condensate (transparent), hail (red), rain (blue), and wind trajectories from a thunderstorm simulation. Figure 2, taken from a model analysis of an extratropical cyclone, shows a grid-mesh potential-temperature contour, a transparent moisture-mixing ratio surface, and trajectories over a topographical map. Figure 11 depicts underground water trajectories with transparent clay layers and text marking the locations of wells.

To enhance the depth information of our videotape animations, we usually combine time animation with a slight rocking of the three-dimensional scene.[3] The amplitude of this rocking is only one or two degrees and may be either vertical or horizontal, depending on the alignment of objects in the scene. This rocking is a very effective depth cue and is not ambiguous with most meteorological motions. We experimented with combining time animation with a steady rotation, but this created great ambiguity between the actual motion of objects and the apparent motion caused by rotation.

Transparency is an important part of our images, but it has presented some special problems. To render transparent surfaces efficiently, we developed a modified Z-buffer algorithm.[2] We analyzed and rendered each of the images in Figures 1-11 in about 30 seconds on our IBM 4381. Our workstation also poses a difficult problem for transparency: It does not separate red, green, and blue channels in its frame buffer. Rather, it was designed for false coloring of two-channel satellite data. We solved this problem by designing our images so that their pixels lie on two-dimensional planes through three-dimensional color space.

Interactive visualization

Our experience with very large data sets has shown the importance of interactivity for their visualization. Thus, we are currently developing a highly interactive version of our system—using the Stellar GS-1000 graphics supercomputer—to produce three-dimensional animations in real time (i.e., drawing at the animation rate). This system can provide an interactive window into data sets containing tens of millions of points produced by numerical models and remote sensing instruments.[4]

Figure 12 is an image produced by this system. It shows contour surfaces for sulfur dioxide (green), water vapor (red), potential temperature deviation (blue), and depth-cued wind trajectories. We can animate such images at five to 10 frames per second from a data set containing 10 different physical variables over 121 time steps in a 25 × 35 × 17-point three-dimensional grid. This data set contains 20 million grid points in a five-dimensional array.

System features. The user controls the display with a mouse, and the system gives immediate response to commands

- rotate, zoom, and pan in three dimensions;
- select any combination of the scalar variables and wind trajectories; and
- start/stop time animation and single-step time forward or backward.

The user can also select new defining levels for contour surfaces using the mouse. The computation of new surfaces occurs at a rate of about two per second and is asynchronous with the animation rendering. The new surfaces replace the old in the animation as they are computed.

Our system holds an entire data set in main memory, along with polygon and vector lists to represent the surfaces and lines generated from gridded data. We use compressed data structures to maximize the size of the data sets we can visualize. The 20-million-point data set is visualized on a 64-megabyte system; a 128-megabyte system could be used for a 50-million-point data set.

The animation involves decompressing the polygon and vector lists, transforming them to a two-dimensional projection, and shading the pixels. Because each fra[illegible] of the animation is generated as it [illegible]played, the user can instantly rotate and zoom by changing the transform function.

The user can also select the combination of viewed physical variables and control the time stepping by changing which polygon and vector lists are processed. The levels of contour surfaces are changed by computing new polygon sets from the gridded data, and this is done asynchronously with animation.

Scientific use. Interactive visualization is crucial to earth scientists, who must vary the way they look at a large data set according to its content. Controlling the combination of variables allows the user to examine specific cause-and-effect mechanisms. Coordinated hand-eye control of the view angle is a powerful way to understand three-dimensional geometries. Control of time stepping is needed to concentrate on particular events. Varying the defining levels of contour surfaces is important to quantitative understanding of data throughout a volume. We are working on an interactive mechanism allowing the user to move a plane through a three-dimensional volume, with contour lines of selected variables rendered on the plane. This mechanism will provide higher information density on the selected plane, so the user can concentrate on a specific spatial region.

A large data set often contains many interesting events and cause-and-effect links. Our system's interactivity lets scientists "play" with their data sets and pursue interesting physics at the pace of their own thoughts, without the distraction of waiting for the system to catch up.

Applications

Our visualizations are being used for teaching at the University of Wisconsin and other institutions. With Patricia Pauley of the University of Wisconsin Meteorology Department, we set up a series of animations on the binocular stereo workstation to help students in a weather laboratory better understand storm systems.[5] We are also making videotapes of weather visualizations available to other teaching institutions.

Scientists are using our visualizations to understand their data sets[6,7] and to present results at conferences. A videotape we produced with Louis Uccellini of the NASA Goddard Space Flight Center vividly depicts the development of the "Presidents' Day" storm that buried Washington, DC, in snow—when no snow was predicted.[8] This storm is being studied intensively by meteorologists trying to improve weather prediction. Our videotape clearly visualizes one simulation of this storm's dynamics and is being used for many conference and classroom presentations.

Scientists have begun using our interactive visualization system based on the Stellar GS-1000. Gregory Tripoli of the University of Wisconsin Meteorology Department used it for only a few hours, but that brief use helped him see and understand problems in his thunderstorm model that he had not seen during several years of viewing his data in two-dimensional plots. He has used this new understanding to correct problems in his model.

Clearly, interactive visualization is revolutionizing the ability of earth scientists to understand their enormous data sets. ☐

Acknowledgments

We would like to thank Verner Suomi, Frances Bretherton, Robert Fox, J.T. Young, and Robert Krauss of SSEC; James Dodge, Gregory Wilson, James Arnold and Paul Meyer of NASA; and Ian Reid, Craig Upson, David Kamins, Jeff Vroom, and Brian Kowalski of Stellar for their help and support. We would also like to thank the guest editor, Gregory M. Nielson, and the *Computer* referees for their suggestions.

This work was supported by NASA Marshall Space Flight Center (NAS8-33799 and NAS8-36292).

References

1. E. Smith, "The McIDAS System," *IEEE Trans. Geosci. Electron*, Vol. GE-13, July 1975, pp. 123-136.
2. W. Hibbard, "4D Display of Meteorological Data," *Proc. Workshop Interactive 3D Graphics*, ACM, New York, 1986, pp. 26-33.
3. W. Hibbard and D. Santek, "Presidents' Day Storm, Visualization/State of the Art: Update," *SIGGraph Video Rev.*, No. 35, 1988.
4. W. Hibbard, "A Next Generation McIDAS Workstation," *Proc. Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology*, Amer. Met. Soc., Boston, 1988, pp. 57-61.
5. P.M. Pauley, W. Hibbard, and D. Santek, "The Use of 4D Graphics in Teaching Synoptic Meteorology," *Proc. Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology*, Amer. Met. Soc., Boston, 1988, pp. 33-36.
6. D. Santek et al., "4D Techniques for Evaluation of Atmospheric Model Forecasts," *Proc. Digital Image Processing and Visual Communications Technologies in Meteorology*, SPIE, Bellingham, Wash., 1987, pp. 75-77.
7. P.J. Meyer and M.S. Seablom, "Application of the Four-Dimensional McIDAS to LAMPS Model Output," *Proc. Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology*, Amer. Met. Soc., Boston, 1988, pp. 33-36.
8. L. Uccellini, "Processes Contributing to the Rapid Development of Extratropical Cyclones," *Preprint Palmen Memorial Symp. on Extratropical Cyclones*, Amer. Met. Soc., Boston, 1988, pp. 110-115.

William L. Hibbard is a researcher at the Space Science and Engineering Center at the University of Wisconsin-Madison. His research interests are interactive computer graphics and image processing for earth sciences. He is a member of ACM, SIGGraph, SIGArt, and the IEEE Computer Society. Hibbard received a BA in mathematics and an MS in computer science from the University of Wisconsin-Madison in 1970 and 1974, respectively.

David A. Santek is a scientific applications programmer at the Space Science and Engineering Center at the University of Wisconsin-Madison. His interests include satellite data analysis, image analysis, and computer graphics. Santek received a BS in atmospheric and oceanic science from the University of Michigan in 1975 and an MS in meteorology from the University of Wisconsin in 1978.

Readers may contact the authors at the Space Science and Engineering Center, University of Wisconsin-Madison, 1225 W. Daytona St., Madison, Wis. 53706.

INTERACTIVITY IS THE KEY

William Hibbard and David Santek

Space Science and Engineering Center
University of Wisconsin
1225 West Dayton Street
Madison, WI 53706

ABSTRACT

The interactivity provided by rendering at the animation rate is the key to visualizing multivarite time varying volumetric data. We are developing a system for visualizing earth science data using a graphics supercomputer. We propose simple algorithms for real-time texture mapping and volume imaging using a graphics supercomputer.

KEYWORDS: Interactive, texture mapping, volume image, earth science.

INTRODUCTION

At the Space Science and Engineering Center we are concerned with the problem of helping earth scientists to visualize their huge data sets. A large weather model output data set may contain one billion points in a five-dimensional array, composed of a 100 by 100 horizontal grid, by 30 vertical levels, by 100 time steps, by 30 different physical variables. Remote sensing instruments such as satellites, radars and lidars produce similarly large data sets. During the last six years we have been developing software tools for managing and visualizing such data sets, as part of the Space Science and Engineering Center's Man-computer Interactive Data Access System (McIDAS). These tools run on an IBM 4381 and produce animation sequences of multivariate three-dimensional images which are viewed on our large multiframe workstations. However, each image takes 10 to 30 seconds of CPU time to analyze and render, and another 30 seconds to load into the workstation frame store, so turn-around time for producing an animation sequence can be several hours. Figure 1 is a typical image produced by this system. We have applied this system to generate animations from at least twenty different model simulation and remote sensed data sources. Although our earth scientist collaborators are usually pleased with the results, they all express a desire to change the animations with quicker response. This is by far their (and our) primary request. Therefore we have begun developing a highly interactive workstation based on the Stellar GS-1000 graphics supercomputer. This system can produce three-dimensional images from model output data sets in real-time, giving the scientist control over the image generation with immediate feedback.

This paper describes earth science data sets, the work we have done so far with the Stellar GS-1000, and some thoughts on further development.

LARGE DATA SETS

Earth scientists are concerned with the three-dimensional domains of atmosphere, ocean, and earth. They are also concerned with the two-dimensional domains which define the boundaries between atmosphere, ocean and earth, particularly with the new drive to understand our planet as a single coupled system. Data sets over these domains are generally sampled at regular grids of points in some two or three-dimensional coordinate system. The vertical dimension is often sampled at higher resolution, and visualized on an expanded scale, because of the extreme thinness of the atmosphere, oceans, and crust when viewed on a global scale. Earth systems are dynamic, so the data sets include a time dimension, usually sampled at regular intervals. The time intervals depend on the nature of the earth system being studied, and are matched to the spatial resolution to give reasonable motions. The data sets are usually multivariate reflecting the interacting physical quantities being studied. At the extreme, atmospheric chemistry data sets may include densities for hundreds of different chemical constituents.

Much remote sensed data takes the form of images. These may lie on some two-dimensional surface through three-dimensional space, or each pixel may integrate along a ray though space. Each pixel may have multiple components, for different spectra, polarizations or other physical quantities. Large satellite images are up to 16,000 by 16,000 pixels. Planned instruments will return hundreds of spectral components at each pixel.

INTERACTIVE 4-D WORKSTATION

We are actively developing software to apply the graphics power of the Stellar GS-1000 to visualize large earth science data sets. Our approach is to store a data set and derived data structures (such as polygon and vector lists) in main memory and to produce real-time animations of multivariate three-dimensional images from these data, giving the user as much interactive control over the image generation process as possible. For a five-dimensional model output data set, main memory contains:

- ** a five-dimensional (three space, time and physical variable) array containing the model output data set. For a very large data set, the memory resident array has reduced time and space resolution and coverage, and a subset of the physical variables.
- ** polygon lists representing contour surfaces through three-dimensional scalar fields.
- ** vector lists with time coordinates representing particle trajectories generated from four-dimensional vector fields.
- ** polygon lists representing a topographical map, along with vector lists for boundary lines.
- ** vector lists representing contour lines of two or three-dimensional scalar fields on a two-dimensional surface.
- ** pixel maps as the destination of rendering operations.

We give the user interactive control over:

- ** time animation, which may be enabled or disabled, single stepped forward or backward, and the animation rate changed.
- ** the perspective mapping for rotation and zoom.
- ** the combination of physical variables to be rendered, including contour surfaces and lines, trajectories and topographical map.
- ** the selection of new contour levels for contour surfaces of scalar variables.

The user gets immediate (a tenth to a fift[illegible] a second) visual feedback to all of these controls. Although the system cannot calculate the polygons for a new contour level at the animation rate, the new surfaces replace the old ones in the animation as they are calculated, so the user immediately begins to see the effect of this control. For three-dimensional grids containing 20,000 points, contour surfaces are calculated at a rate of about two per second, concurrent with real-time animation rendering. It is important that a scientific visualization workstation have the floating point performance to support this type of interaction with the analyses underlying the display.

The reaction of scientists to this interactive visualization of their data sets has been very enthusiastic. A typical pattern of use is to look at the most important variables at various contour levels, along with trajectories, noting all the expected phenomena, until something unexpected is seen. Then a cause is sought, by viewing the unexpected phenomenon in combination with other variables which offer candidate explanations. Often adjustments to contour levels are necessary to see the clearest correlation between variables. When a cause is found, it may also need an explanation, and a backward chain of cause and effect links is followed. The key element of this system is that it reacts to the scientists' controls immediately, so that their thoughts are not interrupted by waiting for the system.

We have sometimes heard that vector line segments and polygonal surfaces are not appropriate techniques for volume visualization. Our experience shows that interactivity is the most important tool for attacking multivariate, dynamic volume visualization problems. The appropriate rendering techniques for such problems are those which can be performed in real-time, giving the user interactive control with immediate response. Thus, we have been concentrating on the vector line segments and polygonal surfaces which can be rendered at real-time animation rates. In the next two sections we will discuss the prospects for real-time animation with other rendering techniques.

VOLUME IMAGING

As part of our software development on the [illegible] mainframe we have experimented with fast approximate algorithms for producing volume images, in which a three-dimensional scalar variable is depicted by a transparent fog, with the opacity at every point proportional to the value of the scalar. We have applied these algorithms to cloud water density in a three-dimensional array of about 40 by 40 by 20, as shown in figure 2. The simplest algorithms offer the hope of real-time rendering. The images they produce do have some artifacts, but they are certainly adequate for interactive visualization.

For our approximate algorithm we use the view direction to the center of the image, and choose the principal axis of the scalar grid most nearly parallel to this view direction. The grid is then treated as a series of layers perpendicular to the chosen axis, which are rendered as transparent flat surfaces, interpolating transparency over each little grid rectangle. An alpha value, which is 0.0 for totally transparent and 1.0 for opaque, is calculated at each grid point as proportional to (or some other function of) the gridded scalar value. The rectangular polygons are then rendered with variable alpha, the layers rendered from back to front. In our implementation we use an alpha buffer containing logarithm(1.0-opacity), which composites by addition rather than multiplication. Table look-ups are used to take logs and antilogs. We assume that color is constant throughout the grid, so that it is not necessary to maintain color buffers. Red, green and blue are simply proportional to the final opacity values. The calculation of opacity from grid point value includes a correction for the path length of the view ray between two layers of the grid. This correction is implemented by applying a table look-up to the opacity which raises it to the power, the secant of the angle between the view direction and the chosen principle axis. This look-up table is merged with the logarithm look-up.

This algorithm generates three noticeable artifacts. First, thin high density layers running obliquely through the grid may be rendered with oscillating bands of intensity, a form of aliasing. Second, the individual layers can be seen at the edge of the array domain. Third, parts of the image change discontinuously when the chosen principal axis changes during rotation. There is a simple fix for the third problem, which helps with the other two. Render three versions of the image, one for each principal axis, and take an average of these weighted by the cosines of the view direction with the principal axes.

The one axis method requires rendering a number of rectangles roughly equal to the number of grid points of the scalar variable, while the weighted average of axes method renders three times that number of rectangles. And, of course, each pixel must be accessed a number of times roughly equal to the number of grid layers rendered. For a 40 by 40 by 20 grid, this is 32,000 or 96,000 rectangles, and accessing each pixel 20, 40 or 100 times. These numbers are a bit high for current graphics engines to do in real-time, but they can be reduced significantly by culling out all rectangles with all four corners below some opacity threshold.

We would certainly like to see support for such rendering algorithms on commercial graphics engines.

IMAGE MAPPING

Much remote sensed environmental data is generated in the form of images on two-dimensional surfaces embedded in three dimensions. For example, a radar scan at a positive elevation sweeps out an image on a cone. Figure 3 shows two perpendicular vertical lidar scans. It is desirable to visualize such data in their natural geometry, possibly combined with other data, and rotating under user control. We believe that current graphics engines can support real-time texture mapping for such images, using a simple algorithm.

The source image to be mapped is stored in memory in a simple two-dimensional array, with X and Y pixel addresses. The surface is divided into polygons. At each vertex of the polygon mesh record the X and Y addresses of the corresponding pixel from the source image. Then render the polygon mesh in three-dimensions with a Z-buffer algorithm. However, where Gouraud shading would interpolate color intensities, interpolate instead the X and Y addresses of pixels. The rendering engine should also combine the X and Y address into a single array address, requiring a multiply. Where this is difficult, it may be possible to use shift and add, and assume that the image is stored in an array with one dimension equal to an even power of two. The result will be an image filled with addresses from the source image. These addresses would then be replaced by the values from the source image, probably using a fast scalar processor or a special vector instruction.

This algorithm requires a flexible system. The rendering processor must be able to combine pixel addresses, and the rendering output should be available to some processing element which can interpret its pixels as addresses and replace them with the addressed values. We would like to see support for this type of texture mapping algorithm on commercial systems.

CONCLUSION

The interactivity implied by real-time animation is the key to giving scientists visual access to their large, multivariate, time dynamic, volume data sets. At SSEC we have begun developing an interactive system for managing, analyzing and visualizing earth science data using the Stellar GS-1000. Our initial work with the GS-1000 demonstrates a breakthrough in providing earth scientists with access to their large data sets.

Because interactivity is crucial, the appropriate rendering techniques for volume visualization are those which can be rendered in real-time. Currently, this means rendering using vector line segments and polygonal surfaces.

However, it should be possible to adapt current graphics engines to real-time rendering of volume images, at least by approximate algorithms, and to texture mapping. The types of simple algorithms presented here, and more complex variations, argue for flexibility in commercial rendering systems. At SSEC we selected the Stellar GS-1000 because its rendering processor is micro-coded and because its pixel maps are rendered into main memory, where they are available as input for multi-stage rendering algorithms. It is even possible in such an architecture that the power of the rendering processor could be applied to data analysis. Environmental data are generated in a great variety of forms, with unpredictable processing requirements; this argues for a flexible and programmable system. Besides flexibility in the rendering processor, we also appreciate systems with powerful scalar and vector processing, and without communications bottlenecks between partitioned memories or processors.

Many meteorologists still draw contour analysis from weather observations by hand. In fact, pads of printed maps showing weather observing station locations are manufactured for this purpose. These pads give the meteorologist coordinated hand-eye interaction with their data sets, which they miss with computer-generated graphics. The graphics supercomputer workstation gives thi hand-eye feel back to meteorologists, applied t data sets containing millions rather than hundrec of points.

ACKNOWLEDGMENTS

We would like to thank P. Pauley, G. Tripoli, F Schlesinger, L. Uccellini, E. Eloranta and all of o other earth science collaborators. This work w supported by NASA (NAS8-33799 and NAS 36292).

REFERENCES

Dreben, R., Carpenter, L., and Hanrahan, [1988] Volume Rendering. *Computer Graphics* 2 4, SIGGRAPH, pp. 65-74.

Foley, J. and Van Dam, A. [1982] *Fundamenta of Interactive Computer Graphics*. Addison-Wesl Publishing Company, 664 pp.

Hibbard, W., Krauss, R.J. and Young, J.T. [198 3-D weather displays using McIDAS. Preprin *Conf. Interactive Information and Processi Systems for Meteor., Ocean., and Hydro.* (L Angeles, AMS, 7-11 Jan. 1985), pp. 153-156.

Hibbard, W. [1986a] 4-D display of meteor logical data. In *Proceedings of 1986 Workshop o Interactive 3D Graphics*. (Chapel Hill, SIGGRAP 23-24 Oct. 1986) pp. 23-36.

Hibbard, W. [1988] A next generation McID workstation. Preprints, *Conf. Interactive Inform tion and Processing Systems for Met, Ocean, a Hydro*. (Anaheim, American Meteorology Socie 1 Jan. - 5 Feb. 1988) pp. 57-61.

Hibbard, W. and Santek, D. [1988] Presiden Day Storm. Visualization/ State of the A Update. SIGGRAPH Video Review number ACM SIGGRAPH.

McCormick, B.H., DeFanti, T.A., and Brov M.D., Editors, [1987] Visualization in Scienti Visualization. *Report to the NSF by the Panel Graphics, Image Processing and Workstatio* ACM SIGGRAPH, 88 pp.

Pauley, P. M., Hibbard, W., and Santek, D. [19 The use of 4-D graphics in teaching synop meteorology. Preprints, *Conf. Interactive Inform tion and Processing Systems for Meteor., Ocea and Hydro*. (Anaheim, AMS, 1 Jan. - 5 Feb. 19 pp. 33-36.

Rogers, D. [1985] *Procedural Elements for Computer Graphics*. McGraw-Hill Book Company, 433 pp.

Santek, D., Leslie, L., Goodman, B., Diak, G., and Callan, G. [1987] 4-D techniques for evaluation of atmospheric model forecasts. In *Proceedings, Digital Image Processing and Visual Communications Technologies in Meteorology* (Cambridge, SPIE, 26-29 Oct. 1987] pp. 75-77.

Upson, C. and Keeler, M. [1988] V-BUFFER: Visible Volume Rendering. *Computer Graphics* 22, 4, SIGGRAPH, 59-64.

3-D ANIMATIONS OF A DEVELOPING EXTRATROPICAL CYCLONE FOR USE IN GENERAL METEOROLOGICAL EDUCATION

Patricia M. Pauley
Linda M. Keller

Department of Meteorology

William L. Hibbard
David A. Santek

Space Science and Engineering Center

University of Wisconsin-Madison
Madison, Wisconsin

1. INTRODUCTION

One of the objectives of any introduction to meteorology is to help the students become more intelligent "consumers" of the increasingly sophisticated presentations of weather used on television. Even TV weathercasts in smaller markets often now present two-dimensional radar loops, satellite loops, and animations of the 24 hr forecast showing highs, lows, fronts, and jet streams. The general public as well as students exploring meteorology in science classes tend to think of only surface processes and typically fail to appreciate the influence of the flow aloft on the surface "weather". A powerful tool that can be used to encourage students to view the atmosphere as three-dimensional rather than two-dimensional is computer graphics.

The work described here is an extension of Pauley et al. (1988), presenting some aspects of the large-scale flow using animations of 3-D graphics for a model simulation of an extratropical cyclone case from April 1982. This cyclone deepened rapidly over the North American rawinsonde network and is well captured in the LAMPS (Limited-Area Mesoscale Prediction System) (Perkey 1976) simulation. Model data have the advantage over observations of providing self-consistent data sets with high temporal resolution, giving smoother animations. The graphics were prepared on a Stellar GS-1000 graphics supercomputer at the University of Wisconsin-Madison using a generalized set of three-dimensional graphics developed in the Space Science and Engineering Center.

The graphics presented in the videotape include animations of such fields as three-dimensional temperature and jet stream structures; the relationship between humidity, cloudiness, and vertical motion; and the three-dimensional motion field. Some two-dimensional graphics are also presented to depict the relationships between the three-dimensional structures and their more typical two-dimensional representations.

2. THE INTERACTIVE 4-D WORKSTATION

The graphics used in this work were prepared on a Stellar GS-1000 graphics supercomputer. The software evolved from a set of 4-D graphics routines written as part of McIDAS (Man-computer Interactive Data Access System) (Hibbard 1986). The McIDAS-based 4-D graphics were invoked using interactive commands, but required considerable CPU time on the IBM 4381 and so could at times take several hours to produce a single animation sequence. The Stellar-based system can produce four-dimensional images (three spatial dimensions plus time) from model output data sets in real time, giving the scientist control over the image generation with immediate feedback (Hibbard and Santek 1989).

The Stellar graphics system can display meteorological data in several ways. The underlying topography is shown as a three-dimensional surface and can be turned on or off to allow the geographical location of features to be portrayed. Basic scalar quantities such as temperature, mixing ratio (humidity), rain water content, cloud water content, vertical motion, and horizontal windspeed are displayed as shaded surfaces. The capability also exists to display a scalar quantity as a wire mesh surface. Vector fields such as the 3-D trajectories are displayed as "ribbons" whose length is proportional to the magnitude of the vector. These graphics can be animated or viewed singly, as well as rotated or zoomed. The system is menu-driven and controlled with a mouse.

3. GRAPHICS FOR THE APRIL 1982 CASE STUDY

The case study used in this project portrays a rapidly intensifying extratropical cyclone with severe weather in an associated cold-frontal squall line, blizzard conditions in the northern Plains, and a dust storm in the southern Plains, a cyclone with something for everyone. The LAMPS simulation was initialized at 0000 UTC 2 April 1982 and ran for 36 hours. Model output is available every two hours for a total of 19 map times, allowing a smooth animation.

Samples of the graphics are shown in Figs. 1-3. It should be kept in mind that the originals are in full color and animated. Fig. 1 depicts the relationship between the jet stream, as portrayed by the 50 m/s windspeed surface, and fronts, as portrayed by the steeply sloping regions of the 0°C temperature surface. The northward progression of moist air from the Gulf of Mexico and its relationship to cloud formation is shown in Fig. 2. Fig. 3 is a frame from an animation sequence showing the 3-D flow field (trajectories) and the jet stream.

4. ACKNOWLEDGEMENTS

The authors would like to thank Prof. Don Perkey, Dr. Pete Robertson, Dr. Mike Kalb, and Mr. Mike Seablom for providing the LAMPS simulation used for this case study, and Mr. Steve Nieman for programming the isobaric post-processor.

5. REFERENCES

Hibbard, W.L., 1986: Computer-generated imagery for 4-D meteorological data. Bull. Amer. Meteor. Soc., 67, 1362-1369.

Hibbard, W.L., and D.A. Santek, 1989: Visualizing large earth science data sets. IEEE Computer, Accepted for publication.

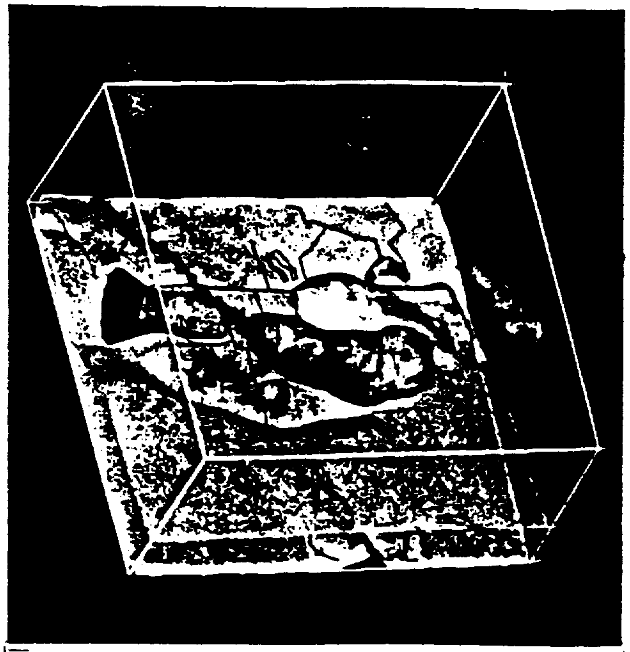


Figure 1: 0°C temperature surface (light shading) and 50 m/s windspeed surface (dark shading) with topography.

Pauley, P.M., W.L. Hibbard, and D.A. Santek, 1988: The use of 4-D graphics in teaching synoptic meteorology. Preprints of the Fourth Intl. Conf. o Interactive Information and Processing Systems for Meteor., Ocean., and Hydro., February 1-5, 1988, Anaheim, CA, 108-110.

Perkey, D.J., 1976: A description and preliminary results from a fine-mesh model for forecasting quantitative precipitation. Mon. Wea. Rev., 104, 1513-1526.

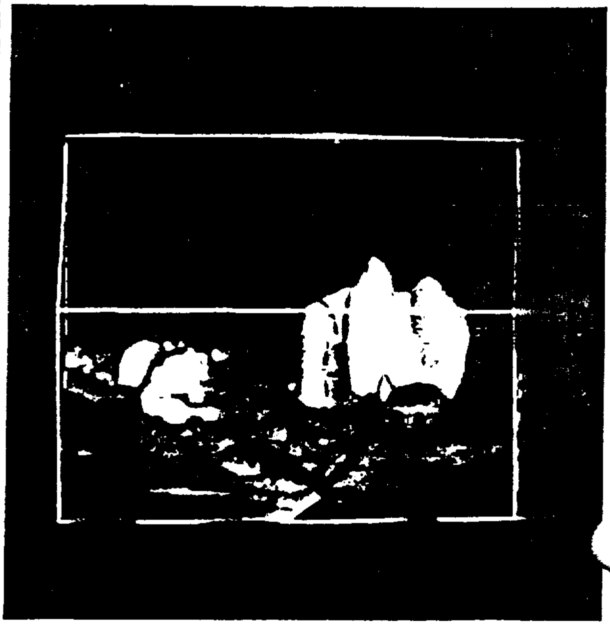


Figure 2: 5 g/kg mixing ratio surface (medium shading) and 0.1 g/kg rain water surface (white shading) with topography.

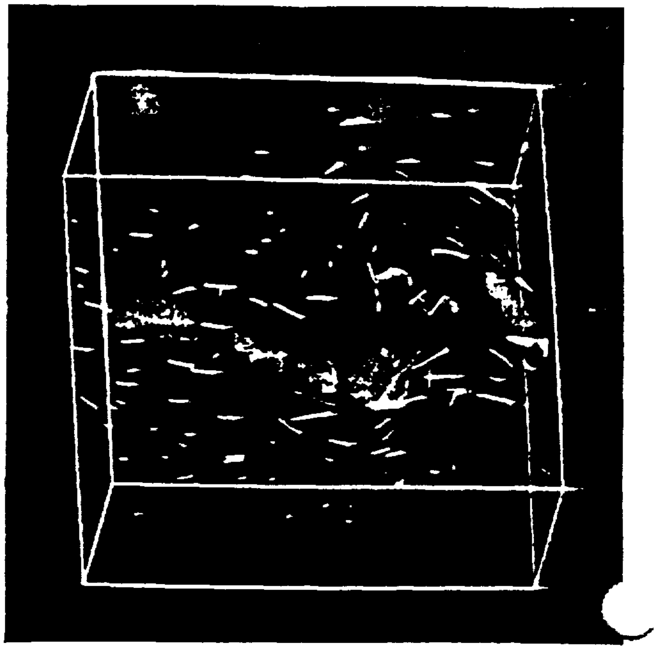


Figure 3: Trajectories and 50 m/s windspeed surface without topography.

J1.2

FOUR-DIMENSIONAL INTERACTIVE ANALYSIS: A TOOL FOR THE EFFICIENT UNDERSTANDING OF LARGE DATA SETS

Gregory J. Tripoli
William Hibbard
David Santek
University of Wisconsin, Dept. of Meteorology, Madison, WI 53706

1 Introduction

In recent years, vast improvements in computer technology have allowed us to both compile observations and model the weather with increasing temporal and spacial resolution. Consequently, the amount of information contained in both model derived and observational data sets has increased geometrically to the point where it has become difficult for one to assimilate all the information available. Recent workstation technology such as McIDAS (Man computer Interactive Data Access System), UNIDATA, AFOS and others, have given us real time access capability with may of these data sets. Nevertheless, the study of fully three-dimensional time dependent data sets remains a difficult task. Normally, it requires the examination of countless two-dimensional cross-sections in an effort to construct the mental three-dimensional image necessary for true understanding. These difficulties have further been exaggerated on the thunderstorm scale and mesoscale where atmospheric structures are even more three-dimensional in nature.

Recent advances in rendering technology have allowed us to view three-dimensional data as solid surfaces enhanced by shading. These rendering algorithms require a large amount of computer power and so have not been employed extensively in real time access systems so far. In the last year, however, workstation technology has finally advanced to the point where it is feasible to employ four-dimensional (space- time) interactive analysis of several variables simultaneously in real time. At the University of Wisconsin -Madison, we have taken advantage of these advances by developing the first interactive four-dimensional analysis workstation known as "Four-Dimensional McIDAS". Real time rendering algorithms allow us to animate a three-dimensional spacial display of any four-dimensional data set while interacting with the image through zooming, panning, rotating and even changing the particular image viewed. As a result, one can take data sets containing tens of megabytes of information and effectively view the entire data set in a matter of seconds.

In this paper, we will emphasize the utility of four-dimensional analysis of numerically modeled atmospheric fields. In the next section, we present a background of numerical model analysis and its goals. In section three, we describe how we have developed the four-dimensional McIDAS to meet these goals so far, while in section four we discuss our plans for further development. We present no figures in this reprint since we rely heavily on color and animation to make our point. Video of the analysis will be central to the oral presentation.

2 Analysis of Numerical Model Output

Atmospheric models represent not only a forecast tool, bu also a subset of a set of tools by which we can effectively build a physically consistent theory explaining observed weather behavior. The models we employ range fron the simple analytical models, such as the quasi-geostrophi omega equation, to the more complex linear semi-analyti models such as the baroclinic instability theory or CIS theory, increasing in complexity up to the full primitiv equation (PE) models. PE models incorporate a wide variety of physics and nonlinear interactions and represent th most realistic depiction of actual atmospheric phenomen we have. Such models are almost always integrated numerically and result in the generation of large space-time dat sets. To the extent that the numerical simulation matche the evolution of observed data, the model derived data se and the physics used to create that evolution explicitly reveal the physical mechanisms responsible for the observed phenomena.

In order for this powerful tool to be effective, we must: 1) be able to compare as many of the aspects of its output with observable parameters as we can and 2) be able to effectively diagnose how the model physics interacted to produce the result obtained. Because observations are normally severely limited in comparison with model simulated fields, it is quite usual to be faced with only a few sketchy observations of a very complicated process. In some cases, the observations may be difficult to find in the large model output data set and so interactivity could be advantageous for verification alone.

Perhaps the most difficult problem in model output analysis is determining exactly why the model produced the result obtained. Traditionally this has been solved either by the use of "sensitivity" tests or by detailed analysis of the magnitude and scale of modeled simulated "forcing" terms. The "sensitivity" test approach can be conclusive for simple overall "importance" tests, but is very expensive and fails to show how the importance of a process varies spatially and evolves with time. The "term analysis" approach reveals the magnitudes of local forcings, but to be an effective tool for diagnosing overall importance of a process, one must view the forcing at many different times and locations. The understanding of the interaction among several "forcings" can be especially difficult.

In the past, we have looked at the relative roles of the "forcing" terms by contouring the state variables at specific times and inferring how strong a process probably is or else picking specific locations and recalculating several of the "forcing" terms locally and comparing relative magnitudes. Numerical modelers have also occasionally presented entire contoured fields of forcing terms, where such terms are relatively simple to reproduce during analysis. More recently, even complex terms are being increasingly presented in a contoured form.

The field analysis of "forcing" terms greatly adds to the volume of variables associated with a particular model integration. For instance, a thunderstorm simulation will usually predict about ten time-dependent variables including wind, pressure, temperature and microphysical quantities. The specific microphysical interaction terms such as riming, accreation, nucleation etc., on the other hand, number as much as 75 alone. Hence when we begin to analyze the "forcing" terms, we can increase the four-dimensional analysis load by as much as one order of magnitude.

With such an enormous volume of variables and processes to analyze in a time varying three-dimensional volume, the process of building a physical picture of the time evolution of the simulated phenomena becomes inefficient and perhaps not possible if one is forced to operate in a fragmented mode, looking at one two-dimensional plot after another over the period of perhaps several months or years!

With the complexity of our models only increasing, it is necessary to find a more efficient method to study the output and the physics leading to the simulated evolution. This is where four-dimensional McIDAS is already making breakthroughs. In the next section we describe some of our current work in this area.

3 Interactive Four-Dimensional McIDAS

Recent advances in super graphics computer technology has allowed Hibbard and Santek (1989) to extend the four-dimensional display technology developed on the McIDAS mainframe (Hibbard, 1986) to run interactively in a workstation environment using the Stellar GS-1000 workstation. The workstation has sufficient power to load as many as 250 time periods of 8 predicted three- dimensional atmospheric quantities into central memory and still have sufficient room to store polygon representations of a chosen three dimensional surface for each of these fields. The high memory to screen transfer rate then allows any or several of these surfaces to be rendered onto the screen as a three dimensional color animated image moving through the 250 time periods at the rate of 10 frames per second! A sophisticated and programmable rendering processor allow the user to control the rendering by rotating, panning around or zooming into the image. In fact, one can zoom all of the way into one surface and view others imbedded inside! Moreover, views of underlying topography and flow trajectories can be rendered simultaneously. The Stellar's multiprocessing unit allows the user to recalculate the stored polygon surfaces at any chosen value while the animated image is being viewed.

Hence, the workstation has provided a powerful tool for completely viewing all aspects of simulated three-dimensional structure within a single intense analysis session. Moreover, the space-time relationship between several individual variables can be viewed clearly, giving a perspective not possible using two-dimensional or singular three-dimensional plots. For example, our 64 megabyte memory allows us to retain up to eight 3D variables at 250 time periods in memory. The animation and interactivity with the viewing perspective allow the user to move to any number of possible vantage points to view any complex combination of the three-dimensional surfaces or trajectories.

Because of the Stellar's large memory and efficient vector processing, the super graphics computer is able to perform at speeds within one order of that of supercomputer mainframes. As a result, large models can be integrated economically on these machines in a research environment.

At the University of Wisconsin - Madison, we have implemented the University of Wisconsin cloud/mesoscale model (Tripoli, 1989a,b) on the Stellar and subsequently view the output using the resident Four-Dimensional McIDAS. The numerical model can typically generate mesoscale data sets including wind, pressure, temperature, detailed microphysical fields, and soil temperature and moisture on a topography following coordinate system.

By being resident on the Stellar system, the model output can be viewed as three-dimensional surfaces either immediately after individual model integrations or displayed as the model is being integrated. Also, since the model code is resident on the graphics workstation, one can easily recover forcing terms and render them as three-dimensional surface images. This allows a perspective of the physical progression of the fields both in terms of the evolution of the state parameters and in terms of the changing physics never before possible using other analysis techniques.

At the oral presentation to this paper we will show video of some actual analysis sessions for several different simulated atmospheric phenomena.

4 Future Development

Clearly, this analysis tool already represents a breakthrough it atmospheric analysis. Our plans for the future are to integrate into this system the remaining McIDAS products. We will continue to develop the product to effectively display any physical process leading to the predicted results. Of course, there are many analysis products traditionally used to study atmospheric data sets which can also be applied to the four-dimensional analysis. Such products may include the application of band-pass filters, or averaging procedures. There is also the need to point any region within a three-dimensional image and require a trajectory to pass through that point, or perhaps ask for an analysis product in association with that point. For instance, we may ask for a time series of pressure or temperature presented in a graphical form over time.

We are also committed to making this interactive analysis compatible with other atmospheric models and data ·ets. We hope to simplify the process of creating algorithms which compute derived parameters, such as forcing terms, for rendering.

In the coming decades, as the computer tecnnology increases, we expect that we will increasingly lose the distinction between the data analysis, the model run and the model analysis. The model will likely become an essential part of any analysis system, increasing the effectiveness of both the observations and model through data assimilation. The interactive four-dimensional McIDAS running alongside and in conjunction with a sophisticated three-dimensional model is a first step in this direction .

5 Bibliography

Hibbard, W.L., 1986: Computer-generated imagery for 4-D meteorological data. *Bull. Amer. Meteor. Soc.* 67, 1362-1369.

Hibbard, W.L., and D. A. Santek, 1989: Visualizing large earth science data sets. *IEEE Computer.* Accepted for publication.

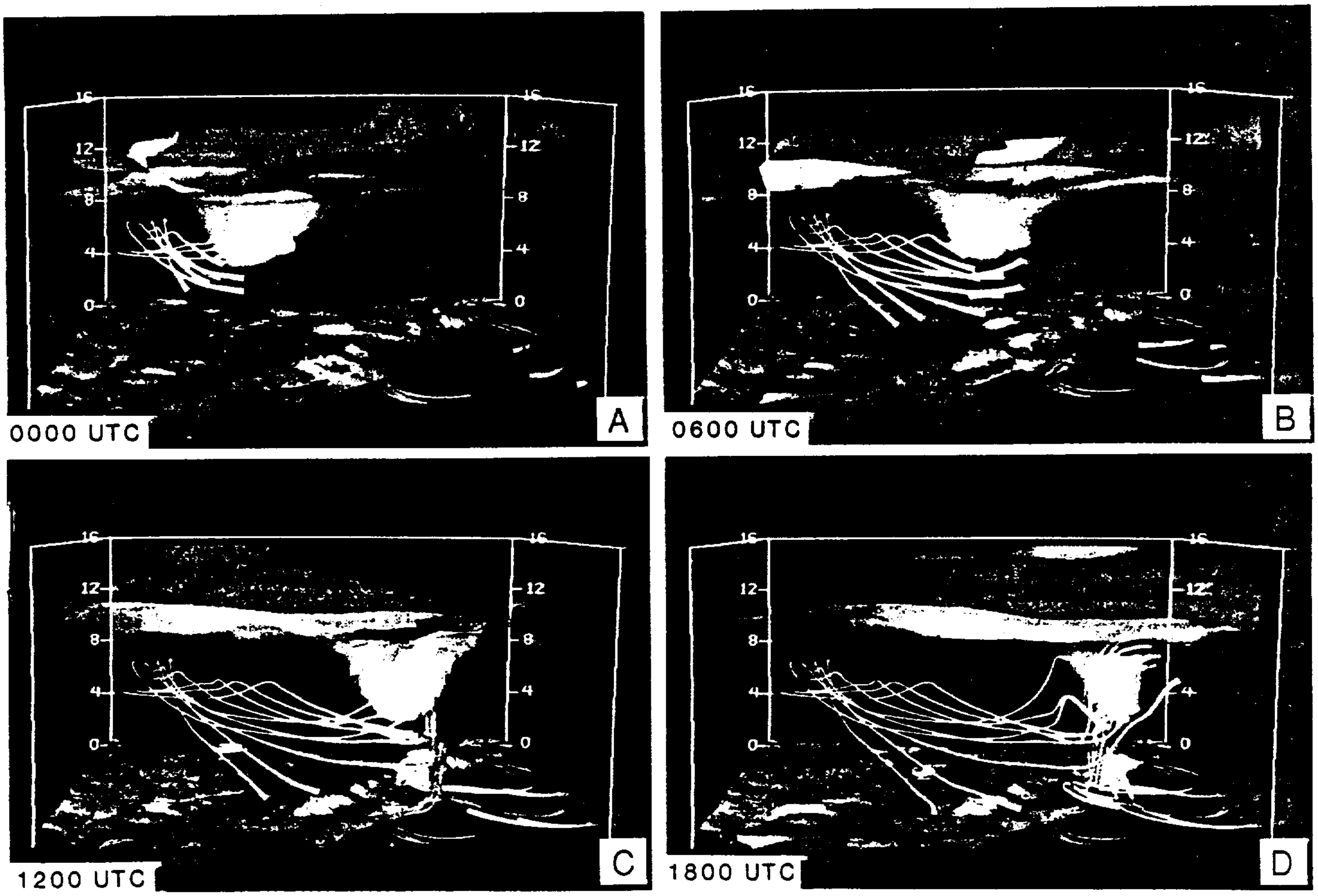
Tripoli, G.J., 1989: The University of Wisconsin Regional Atmospheric Modeling System: Dynamical Framework and Simulation of Geostrophic Adjustment *Mon. Wea. Rev.* 117, 305-328.

Tripoli, G.J. and W. R. Cotton, 1989: A Numerical Study of an Observed Orogenic Mesoscale Convective System in the Lee of the Rockies. Part 2: Analysis of Governing Dynamics. *Mon. Wea. Rev.* 117, 305-328.

Bulletin of the American Meteorological Society

Volume 70 Number 11 November 1989

3D PERSPECTIVE OF THE PRESIDENTS' DAY STORM: 19 FEBRUARY 1979



ORIGINAL PAGE IS
OF POOR QUALITY

[Reprinted from BULLETIN OF THE AMERICAN METEOROLOGICAL SOCIETY, Vol. 70, No. 11, November 1989]
Printed in U. S. A.

Application of the 4-D McIDAS to a Model Diagnostic Study of the Presidents' Day Cyclone

W. Hibbard,
L. Uccellini,
D. Santek,
and K. Brill

Cover: A three-dimensional perspective (as viewed from the south) of the 2×10^{-5} K mb^{-1} s^{-1} potential vorticity surface at 1200 UTC 19 February 1979; derived from the application of the University of Wisconsin's 4-D McIDAS to a numerical simulation of the Presidents' Day cyclone (Whitaker et al. 1988). Also included are trajectories derived for a 24-h period ending at 1200 UTC 19 February computed using 15-min model output. Blue trajectories originate within stratospheric extrusion west and north of cyclonic region along the east coast; yellow and red trajectories originate in the low levels within the ocean-influenced planetary boundary layer east and south of the cyclone. See article by Hibbard, Uccellini, Santek, and Brill for details.

Application of the 4-D McIDAS to a Model Diagnostic Study of the Presidents' Day Cyclone

W. Hibbard,[*]
L. Uccellini,[**]
D. Santek,[*]
and K. Brill[+]

Abstract

The four-dimensional (4-D) McIDAS system is applied to a numerical simulation of the rapid development phase of the Presidents' Day cyclone. Selected frames from a videotape of the model simulation are presented to illustrate the evolution of the upper- and lower-tropospheric potential vorticity maxima prior to and during rapid cyclogenesis. The 4-D structure of various airstreams converging toward the cyclone center are also displayed. Our experience with 4-D displays of model output indicates that these systems offer a tremendous opportunity to manage and dissect the information content inherent in numerical simulations. The production of the visualizations of the Presidents' Day cyclone also confirmed a need for an interactive system capable of producing various perspectives in real time, a requirement being addressed with the development of a new workstation at the University of Wisconsin Space Science and Engineering Center.

1. Introduction

The application of high-resolution numerical model output to case studies of severe weather events provides the opportunity to study atmospheric circulation systems with a confidence level not previously attained using the operational data network alone; an advancement that is having a significant impact on synoptic meteorology (Keyser and Uccellini 1987). Whitaker et al. (1988) present a synoptic analysis and diagnostic study of the 19 February 1979 Presidents' Day cyclone using a regional-scale numerical-model simulation in which the details of their analysis far exceed previous analyses based only on 12-h operational radiosonde data (Bosart and Lin 1984; Uccellini et al. 1985). Model datasets at 15-min to 1-h intervals were used to study various physical processes, including jet streak circulation patterns, a stratospheric extrusion within a tropopause fold, latent heat release, and sensible heat fluxes within the planetary boundary layer that were all related to the rapid cyclogenesis that marked that case. More importantly, the use of high-resolution model data by Whitaker et al. provides the means to resolve the interaction of the various physical processes, trace the stratospheric and lower tropospheric potential vorticity maxima, and compute detailed trajectories depicting the three-dimensional (3-D) airflow through the developing storm system.

The major problem that now confronts the meteorologist attempting to dissect model simulations of scale-interactive weather and climate processes is not a lack of useful and dynamically consistent datasets, but the real possibility of being overwhelmed by the model data. This is especially true when a short time interval in the model output is required to resolve the processes that interact to produce the rapid evolution of severe weather events. The purpose of this paper is to describe an application of the Man Computer Interactive Data Access System (McIDAS) at the University of Wisconsin's Space Science and Engineering Center (SSEC) to produce a four-dimensional (4-D) display of selected meteorological fields generated by the model simulation of the Presidents' Day cyclone.

The 4-D McIDAS system (Hibbard 1986a) is designed to manage and analyze very large datasets (either from remote sensing instruments or numerical models) and to produce complex, 3-D multivariate images from these datasets. The real power of the system involves the animation of these fields and the ability to change the perspective so that meteorologists are provided the means to "observe" the atmosphere in motion from various points of view. Sequences of the model fields and trajectories have been animated for the Presidents' Day cyclone and recorded on videotape. We can, of course, only provide snapshots of selected model-generated fields and trajectories in this paper to highlight the type of imagery and the range of perspectives available.

A brief review of the 4-D McIDAS system is provided in section 2, and the steps taken to apply this system to the Presidents' Day storm simulation are discussed in section 3. Selected examples of meteorological fields are displayed in section 4 and the results and future plans are summarized in section 5.

---

[*] Space Science and Engineering Center, University of Wisconsin—Madison, Madison, WI 53706

[**] Meteorological Operations Divisions, National Meteorological Center, Camp Springs, MD 20746

[+] General Sciences Corporation, Code 612, NASA/GSFC, Greenbelt, MD 20771

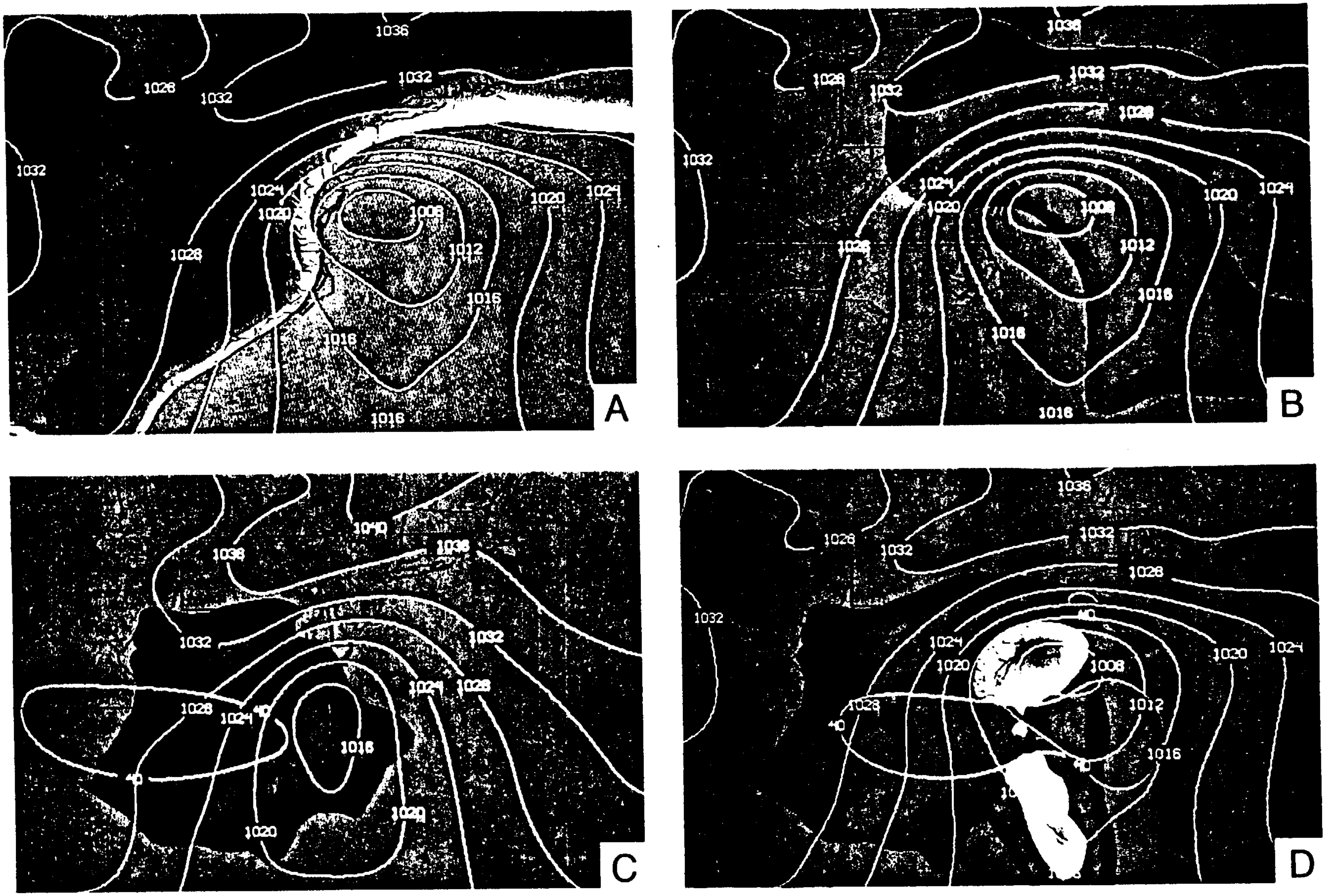
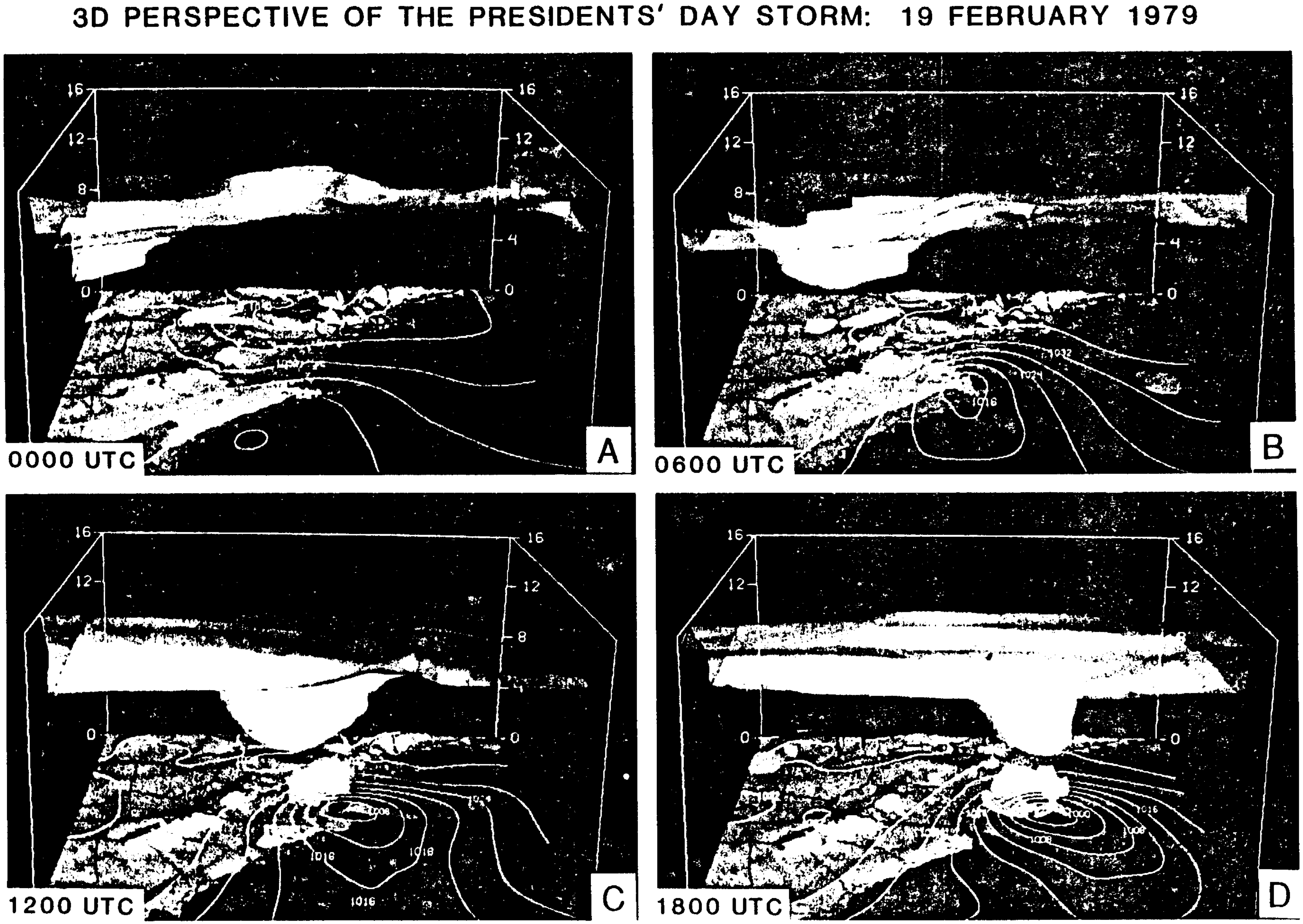


Fig. 1. Two-dimensional (2-D) displays of model-generated fields of the Presidents' Day cyclone (a) sea level pressure at 1200 UTC 19 February (mb), and temperature on lowest model sigma surface; shaded interval for isotherm analysis is 1°C with the 0°C line indicated in black; (b) sea level pressure (SLP) and relative humidity at 700-mb level at 1200 UTC 19 February; light green is 70% and dark green is 90% relative humidity; (c) SLP, 40 m s⁻¹ isotach at 500 mb level, and accumulated precipitation shown for 0600 UTC 19 February and (d) 1200 UTC 19 February; color interval for accumulated precipitation is 1 cm, 2 cm and >4 cm.

FIG. 2. Three-dimensional (3-D) perspectives (as viewed from the south) of the 2×10^{-6} K mb^{-1} s^{-1} potential vorticity surface, and SLP pattern (mb) derived from the numerical simulation of the Presidents' Day cyclone (Whitaker et al. 1988) for (a) 0000 UTC, (b) 0600 UTC, (c) 1200 UTC, and (d) 1800 UTC 19 February 1979.



3D PERSPECTIVE OF THE PRESIDENTS' DAY STORM: 19 FEBRUARY 1979

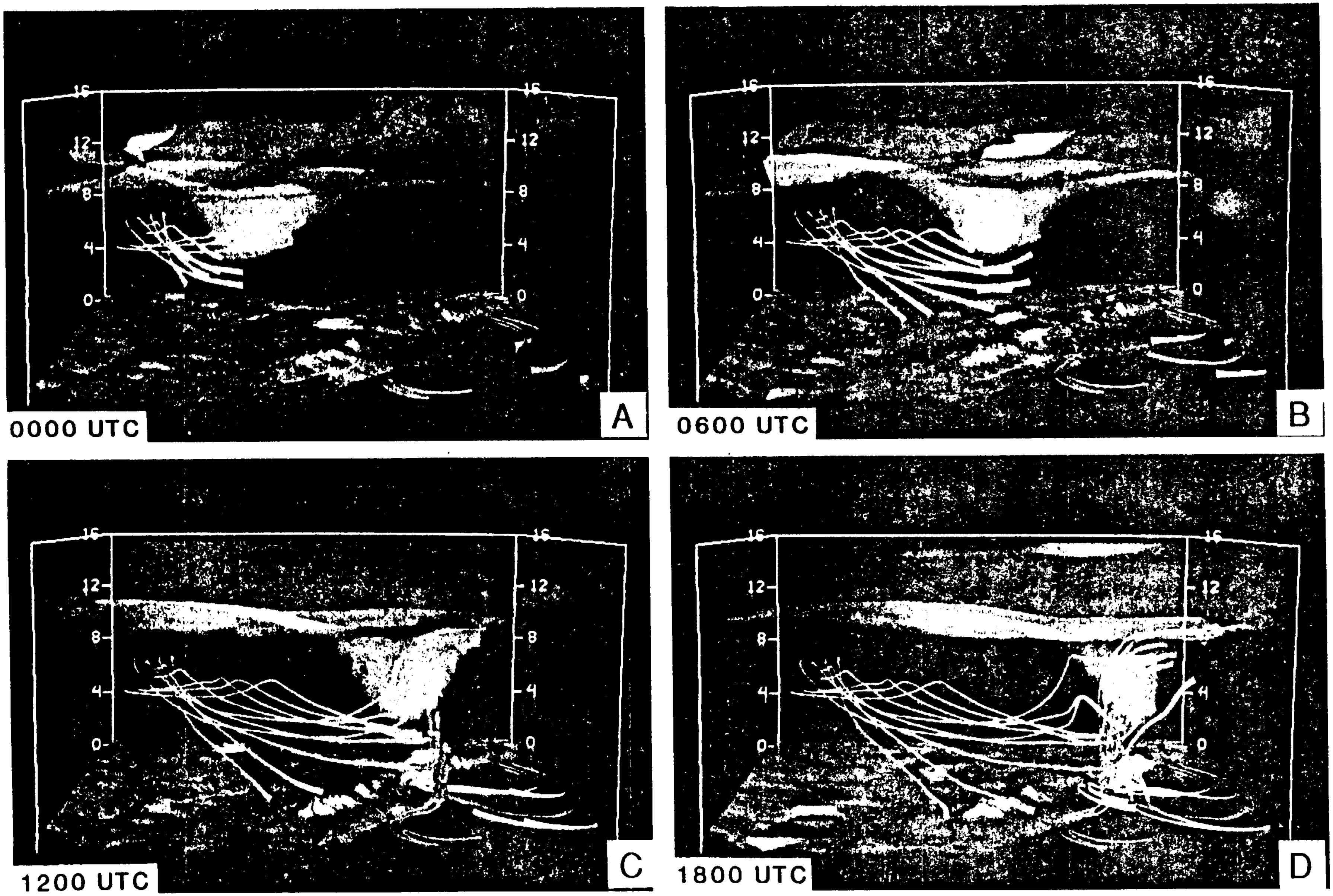
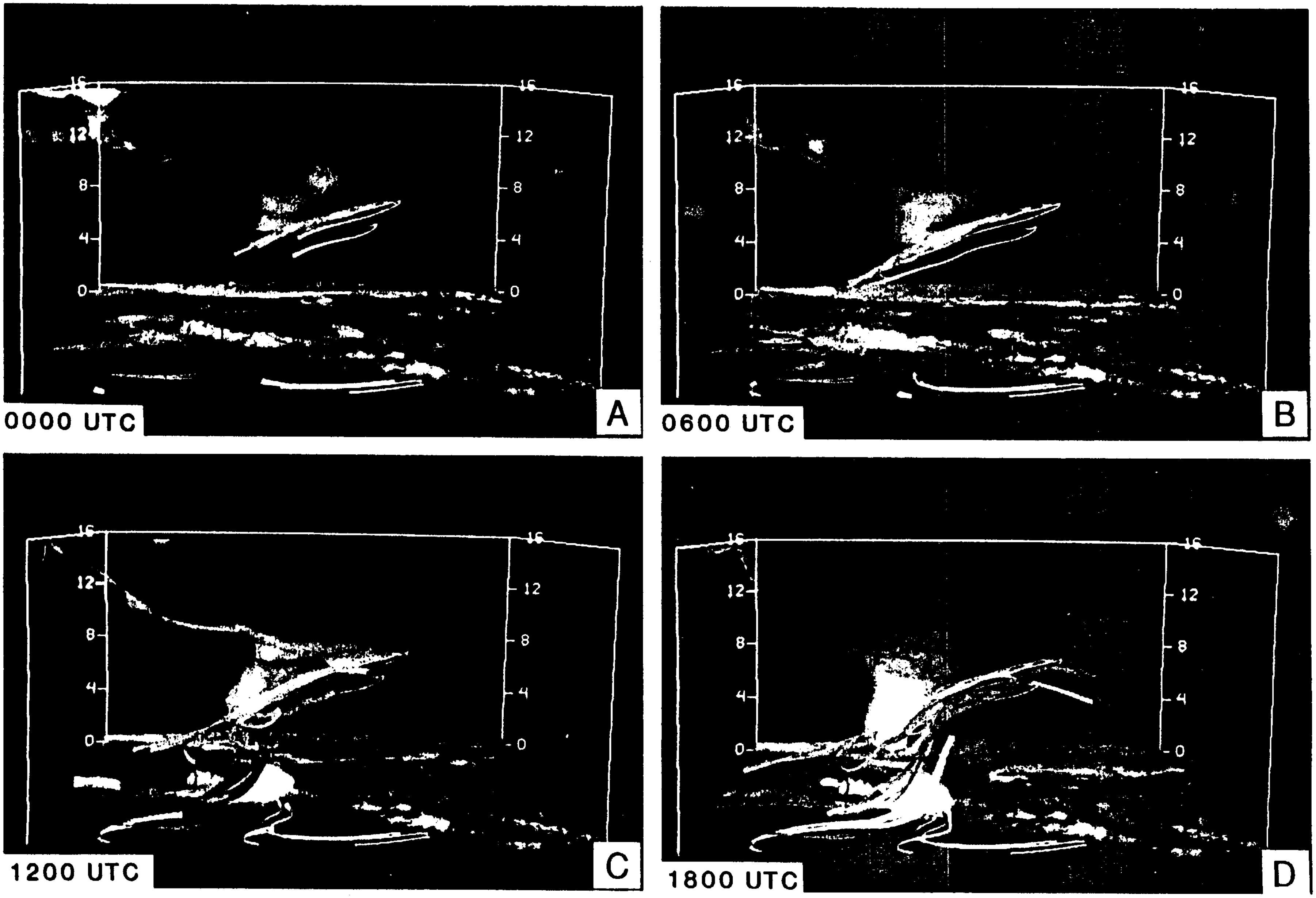


FIG. 3. As in figure 2, but with SLP analyses removed and trajectories included. Trajectories are derived from 15-min model output as described by Whitaker et al. (1988). Blue trajectories originate within stratospheric extrusion west and north of the cyclone, and yellow and red trajectories originate in the low levels within the ocean-influenced planetary boundary layer east and south of the cyclone.

FIG. 4. As in figure 3, but as viewed from an eastern perspective, illustrating the sloped nature of both the potential vorticity surface and the ascent of the trajectories approaching the storm system from the south.

3D PERSPECTIVE OF THE PRESIDENTS' DAY STORM: 19 FEBRUARY 1979



2. The four-dimensional McIDAS

The 4-D McIDAS is a set of software integrated within the McIDAS system, a development guided by its application to a variety of datasets including numerical model output (Santek et al. 1987; Pauley et al. 1988; Meyer and Seablom 1988) and remote sensing observations (Hibbard 1986a; Hibbard 1986b). The 4-D McIDAS provides tools to manage data as 2- and 3-D grids, as trajectories, as images, and as collections of data without any spatial order. The grid structures can be grouped to manage very large datasets spanning multiple physical variables and sequences of times. These 4-D McIDAS data-management tools include file structures for storing data, libraries of routines for accessing those files, user commands for housekeeping functions (listing, copying, etc.) on those files, and programs for converting external data to those file formats.

The 4-D McIDAS includes a variety of commands for analyzing data in the McIDAS file structures. These include

1) resampling grids to a different spatial resolution, map projection, or vertical coordinate
2) resampling time sequences of grids to a different temporal resolution
3) transforming grids to a moving frame of reference
4) generating general arithmetic grid operators
5) interpolating nonuniform data to a grid
6) deriving trajectories from grids of wind components
7) creating images from grids

The 4-D McIdAS is used to produce animated sequences of 3-D images from data in the McIDAS file structures. The visual elements of these images include

1) shaded relief topographical maps with physical and political boundaries
2) trajectories drawn as shaded tubes, which may be either opaque or semitransparent, and may be long and tapered or short with the length made proportional to wind speed
3) isolevel contour surfaces of 3-D scalar variables, which may appear smooth with natural shading, may be semitransparent, or may have a gridded "fishnet" appearance. (An example would be a surface of constant wind speed enclosing a jet stream.)
4) isolevel contour lines, drawn either on the topographical surface or on a surface in the atmosphere (such as an isobaric or theta surface)
5) images projected onto 2-D surfaces in the atmosphere, such as those scanned by radar or lidar
6) 3-D transluscent volumes with opacity proportional to some scalar physical variable. (This is a natural way to depict cloud water density.)
7) cloud top surfaces generated from visible and infrared satellite images

These 3-D images are assembled into animated sequences showing the time evolution of a dataset, or showing a rotating view of a single time within a dataset. The animated sequences are loaded into the McIDAS workstation where they can be viewed directly or recorded onto videotape.

The 4-D McIDAS provides a variety of means to increase viewer comprehension of the images it produces. The images are generated in color, and the user can control the color of each physical variable. The user can also control the scale of the depicted spatial region and the rotation of the user's viewpoint. Depth information is somewhat ambiguous in 3-D images, and a rotating perspective helps resolve this ambiguity. However, the apparent motions of depicted objects become ambiguous when rotation is simultaneous with time animation. Our solution to this problem is to apply a slight "rocking" to the entire 3-D domain during time animation. The rocking is not ambiguous with most meteorological motions, and it does enhance the sensation of depth.

We have used the 4-D McIDAS system to produce videotapes from many datasets. These videotapes are generally composed of many separate animation sequences, including 2- and 3-D images. The different animation sequences show the data at a variety of scales and viewpoints, and with different combinations of physical variables. Given the complexity of a large dataset, and the large amounts of human and computer resources needed to create a videotape, we have found it important to work from a script. The script provides a way to plan the presentation, to see its parts in perspective, and to avoid making costly mistakes during the repetitive parts of the production process.

3. McIDAS applied to the Presidents' Day storm

The rapid development phase of the Presidents' Day storm was simulated at the Goddard Space Flight Center (GSFC) using the Mesoscale Analysis and Simulation System (MASS) as described by Whitaker et al. (1988). The dataset consists of 2- and 3-D grids plus trajectories computed at 15-min intervals over a 36-h period from 1200 UTC 18 February to 0000 UTC 20 February 1979. Three-dimensional grids for temperature, specific humidity, u and v wind com-

ponents, and potential vorticity include data for all 32 levels of the model. There are 2-D grids for terrain topography, surface pressure and pressure tendency, surface temperature, stable and convective precipitation, and planetary boundary-layer thickness. The 2- and 3-D grids are in a polar stereographic projection of 102 × 70 grid points. The dataset also includes 21 selected trajectories computed at GSFC using an iterative technique and a 15-min model output as described by Whitaker et al. (1988).

For the purpose of visualizing the storm's development, we used the 4-D McIDAS tools to select a subset of this dataset, covering 60 time periods, at half-hour intervals between 1500 UTC 18 February and 2030 UTC 19 February 1979. The gridded data were interpolated to psuedomercator projections (constant latitude and longitude grid intervals) covering several geographical subsets of the original region and to uniform height levels in the vertical.

The goals of this effort are to visualize the basic evolution of the sea level pressure (SLP) field, the relationship of this development to upper-level jets and convergence of various airstreams during the storm development, the cold air damming along the coast, and the relation of potential vorticity to the converging airstreams and rapid cyclogenesis. Satisfying our goals required many trial and error adjustments to

1) the map scale
2) the view angle
3) the combination of physical variables
4) the contour levels of scalar variables
5) the color and transparency of various image elements
6) the means of displaying trajectories

This trial and error process was very time consuming, usually requiring several hours to see the result of even a small change. This effort illustrates the need for a more interactive system.

4. Examples of colored displays of the Presidents' Day cyclone

In this section, several examples of the 2- and 3-D colored displays are presented to illustrate the ability of the system to overlay various fields and provide insight into the structure of and airflow through the rapidly developing cyclone. These examples, of course, cannot convey the true power of the system, which can only be appreciated when viewing the videotape sequences.

a. 2-D displays

The SLP pattern and analysis of the temperature on the lowest model surface (approximately 20 mb above the ground) are shown in figure 1a and depict the position of the developing surface low-pressure system with respect to the enhanced baroclinic region along the East Coast of the United States at 1200 UTC 19 February. The model simulation captures the damming of the colder air (less than 0°C in blue) between the Appalachian Mountain range and the coastline and the warmer air (15°–20°C) immediately off the coast. The combination of the 700 mb relative humidity and SLP (figure 1b) shows that the surface low also developed along the gradient between the moist rising air to the north and east of the storm center and dry subsiding air to the south and west. The development of the surface low along this relative humidity gradient is indicative of the asymmetric cloud structure that marks extratropical cyclones—a direct result of the convergence of various and distinctly different airstreams toward the storm center.

The sequence of SLP, accumulated precipitation, and the 40 m s^{-1} isotach at the 500 mb level (figures 1c and 1d) illustrates the rapid development of the simulated cyclone between 0600 UT and 1200 UTC 19 February. The colorized sequence provides supporting evidence that rapid cyclogenesis commenced as the exit region of the upper-level polar jet streak approached the East Coast, and as the precipitation rate increased to the north of the surface low within the cyclonic side of the jet exit region. Combining these model fields within a time sequence provides supporting evidence for the importance of both the dynamic and thermodynamic processes (i.e., jet streak circulation patterns and latent heat release) and their interaction as important factors for the onset and maintenance of rapid cyclogenesis.

b. 3-D displays

Attempts have been made to link the extrusion of stratospheric air into the upper- and middle-troposphere to cyclogenesis through the principle of conservation of isentropic potential vorticity (IPV), where IPV $\cong (\zeta_\theta + f)\partial\theta/\partial p$. As stratospheric air descends into the troposphere, the air mass is stretched and the static stability $(-\partial\theta/\partial p)$ decreases significantly. Consequently, the absolute vorticity $(\zeta_\theta + f)$ increases with respect to parcel trajectories as long as the stratospheric values of IPV are preserved. Recent review articles by Hoskins et al. (1985) and Uccellini (1989) emphasize the impact of the IPV anomalies on surface cyclogenesis. Through an "invertibility principle" expressed by Kleinschmidt (1950), Hoskins et al. show that a positive IPV anomaly that extends

downward from the stratosphere into the middle troposphere provides an optimal situation for enhancing the IPV advection in the middle and upper troposphere which acts to induce a cyclonic circulation that extends throughout the entire troposphere. The downward extrusion of stratospheric air from the normal tropopause level to the middle and lower troposphere is known to occur in narrow zones along upper-level front/jet streak systems, a process called tropopause folding (Reed 1955; Reed and Danielsen 1959; Danielsen 1968).

As noted by Uccellini et al. (1985), the dominant theme has been to relate the *simultaneous development* of the tropopause fold to cyclogenesis. However, a result of the Uccellini et al. and Whitaker et al. (1988) studies of the Presidents' Day cyclone is the emphasis on the probable link between upper-level frontogenesis and tropopause folding associated with jet streak transverse circulation patterns and the *subsequent* development of surface cyclones. To show the tropopause fold prior to cyclogenesis, the subsequent translation of the stratospheric air eastward toward the East Coast, and the interaction of this air mass with a low-level IPV maximum, Whitaker et al. (1988) utilize numerous standard four-panel horizontal and cross-sectional figures in their paper. The application of the 4-D McIDAS provides a clear depiction of these features, as illustrated below.

The 3-D perspective of the simulated tropopause fold and eastward displacement of the stratospheric air mass is shown from a southern perspective in figure 2, with selected trajectories in figure 3, and from an eastern perspective in figure 4, at 6-h intervals between 0000 and 1800 UTC 19 February. The descent of the 2×10^{-5} K $\mathrm{mb}^{-1}\mathrm{s}^{-1}$ IPV surface within a tropopause fold and advection of this IPV anomaly toward the East Coast and subsequent deepening of the surface cyclone are clearly depicted in the 3-D illustrations. Figure 2 depicts the descent of the stratospheric air mass down to the 700-mb level 12 h prior to and 1500 km upstream of the developing cyclone in association with a descending airstream (blue trajectories in figures 3 and 4) originating on the cyclonic side of the polar jet streak (depicted in figures 1c and 1d). Whitaker et al. (1988) show that this stratospheric extrusion was related to subsynoptic scale processes associated with the upper-level jet/front system in the middle of the United States. The eastern perspective (figure 4) is included to show the tilt of the tropopause fold (represented by the IPV surface) toward the south as the stratospheric air descends beneath the jet core and moves toward the east, with the expected general tropopause slope (higher to the south and lower to the north) also depicted. The model-generated images of the tropopause fold are similar to the schematics produced by Danielsen (1968).

The model-simulated IPV fields in figures 2–4 also display a separate region of high IPV confined to the lower troposphere along the East Coast that extends upward during the period of rapid cyclogenesis. As discussed by Gyakum (1983), Bosart and Lin (1984), Boyle and Bosart (1986), and Whitaker et al. (1988), diabatic processes (primarily associated with the vertical and horizontal distribution of latent heat release within a low-level baroclinic zone) contribute to the development of the low-level IPV anomaly. The low-level IPV maximum will also induce a cyclonic circulation extending upward throughout the entire troposphere and add to the circulation induced by the upper-level system, as long as the low-level IPV maximum remains downwind of the upper-level maximum, maintaining a positive feedback between the two. As with the upper-tropospheric IPV surface, rotating the display from a southern perspective (figure 3) to an eastern perspective (figure 4) provides an indication of the sloped nature of the low-level inflow and IPV surface (from south to north) as the separate IPV maxima begin to merge by 1800 UTC (figure 4d).

The previous discussion of the cyclogenetic processes as viewed from an IPV perspective provides a basis for linking upper- and lower-tropospheric processes in the evolution of a rapidly developing storm. This perspective is similar to the "type B" cyclogenesis described by Petterssen and Smebye (1971), which relates the advection of absolute vorticity ahead of an upper-level trough over a thermal advection pattern associated with a low-level baroclinic zone to surface cyclogenesis. Both perspectives not only account for the upper-tropospheric processes associated with trough/ridge systems and jet streaks, but also for the influence of low-level baroclinic zones in the evolution of these storms. The depiction of model results on the 4-D McIDAS clearly show that both of these perspectives are applicable to the Presidents' Day cyclone.

Finally, the trajectories shown in figures 3 and 4 give the appearance of distinct airstreams and "conveyor belts" as described for extratropical storms by Browning and Harrold (1969) and Carlson (1980). The "dry airstream" (represented by the blue trajectories) descending from the west-southwest, a "cold conveyor belt" (represented by the red trajectories) originating off the northeast coast of the United States and approaching the cyclone from the east, and a "warm conveyor belt" approaching the storm from the south and rising rapidly in a sloped-narrow band and turning anticyclonically are all depicted by the 4-D McIDAS display in figures 3 and 4. The low-level trajectories in figures 3 and 4 that 1) approach

the storm from the east and wrap around the center, and 2) approach the storm center from the south and rise rapidly in a narrow-sloping band before turning anticyclonically pass through the low-level 2×10^{-5} K $mb^{-1}s^{-1}$ IPV surface along the East Coast. These parcel trajectories are influenced by the latent heat simulated by the model and illustrate the nonconservative aspects of the low-level flow through the IPV maximum in a region of heavy precipitation (figures 1c and d; see Whitaker et al. [1988]).

5. Summary

Examples of 2-D and 3-D illustrations of selected fields from a videotape of a model simulation of the 19 February 1979 Presidents' Day cyclone demonstrate the value of 4-D display systems for visualizing voluminous datasets produced by numerical simulations of synoptic-scale phenomena. The application of the 4-D McIDAS system shows the interaction among various processes throughout the troposphere and lower stratosphere associated with two distinct potential vorticity maxima which are difficult to visualize with standard 2-D graphics procedures. Furthermore, the system provides a clear depiction of the 3-D structure of the various airstreams or "conveyor belts" that converge toward the cyclone center and extend from the planetary boundary layer to the tropopause. As noted earlier, the power of this approach can be realized by observing the time evolution of these diagnostic fields and trajectories, which are shown on the videotape. Versions of the videotape are available upon request.[1]

The production of the videotape of the Presidents' Day cyclone was a slow and expensive process. The trial and error refinement of the animation of the Presidents' Day cyclone was inhibited by the long turnaround time of the system and the size of the dataset. What is needed is a system to provide a rapid visualization of the information inherent in numerical model output and remotely sensed observations. To meet these requirements, the McIDAS system has been recently converted to the Stellar GS-1000, a graphics supercomputer, and the 4-D McIDAS has been adapted to create 3-D animations *in real time* (the images are drawn at the animation rate). This system allows the user to explore a dataset of up to 50-million grid points, providing immediate visual feedback to the user's trial and error adjustments to the 3-D image animations (Hibbard and Santek 1989).

Acknowledgments. The authors would like to thank Dr. James Dodge of the National Aeronautics and Space Administration (NASA) for his encouragement and support of this effort. Funding for the University of Wisconsin 4-D McIDAS workstation was received from the NASA Marshall Space Flight Center (NAS8-36292).

References

Bosart, L. F., and S. C. Lin. 1984. A diagnostic analysis of the Presidents' Day storm of February, 1979. *Mon. Wea. Rev.* **114:** 2432–2465.

Boyle, J. S., and L. F. Bosart. 1986. Cyclone-anticyclone couplets over North America. Part II: Analysis of a major cyclone event over the eastern United States. *Mon. Wea. Rev.* **114:** 2432–2465.

Browning, K. A., and T. W. Harrold. 1969. Air motion and precipitation growth in a wave depression. *Quart. J. Roy. Meteor. Soc.* **95:** 288–309.

Carlson, T. N. 1980. Airflow through midlatitude cyclones and the comma cloud pattern. *Mon. Wea. Rev.* **108:** 1498–1509.

Danielsen, E. F. 1968. Stratospheric-tropospheric exchange based upon radioactivity, ozone and potential vorticity. *J. Atmos. Sci.* **25:** 502–518.

Gyakum, J. R. 1983. On the evolution of the QE II storm. II: Dynamic and thermodynamic structure. *Mon. Wea. Rev.* **111:** 1156–1173.

Hibbard, W. 1986a. 4-D display of meteorological data. Proceedings, 1986 Workshop of Interactive 3-D Graphics. Chapel Hill, NC: SIGGRAPH. 23–36.

Hibbard, W. 1986b. Computer generated imagery for 4-D meteorological data. *Bull. Amer. Meteor. Soc.* **67:** 1362–1369.

Hibbard, W., and D. Santek. 1989. Visualizing large data sets in the earth sciences. *IEEE Computer.* **22**(8): 53–57.

Hoskins, B. J., M. E. McIntyre and A. W. Robertson. 1985. On the use and significance of isentropic potential vorticity maps. *Quart. J. Roy. Meteor. Soc.* **111:** 877–946.

Keyser, D., and L. W. Uccellini. 1987. Regional Models: Emerging research tools for synoptic meteorologists. *Bull. Amer. Meteor. Soc.* **68:** 306–320.

Kleinschmidt, E., Jr. 1950. On the structure and origin of cyclones (Part 1). *Meteor. Rundsch.* **3:** 1–6.

Kleinschmidt, E., Jr. 1957. Cyclones and anticyclones. In *Dynamic meteorology. Handbuch der Physik, Vol. 48.* ed. J. Bartels. New York: Springer-Verlag.

Meyer, P. J., and M. S. Seablom. 1988. Application of the four-dimensional McIDAS to LAMPS model output. Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Anaheim. AMS Preprint, pp. 33–36. Boston: American Meteorology Society.

Pauley, P. M., W. Hibbard and D. Santek. 1988. The use of 4-D graphics in teaching synoptic meteorology. Conference Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Anaheim. AMS Preprint, pp. 37–40. Boston: American Meteorology Society.

Petterssen, S., and S. J. Smebye. 1971. On the development of extratropical cyclones. *Quart. J. Roy. Meteor. Soc.* **97:** 457–482.

Reed, R. J. 1955. A study of a characteristic type of upper-level frontogenesis. *J. Meteor.* **12:** 226–237.

Reed, R. J., and E. F. Danielsen. 1959. Fronts in the vicinity of the tropopause. *Arch. Meteor. Geofys. Bioklim.*, **A11:** 1–17.

Santek, D., L. Leslie, B. Goodman, G. Diak and G. Callan. 1987. 4-D techniques for evaluation of atmospheric model forecasts. Proceedings, Digital Image Processing and Visual Communications Technologies in Meteorology. Cambridge. SPIE. 75–77.

---

[1] Please contact William Hibbard at the Space Science and Engineering Center at the University of Wisconsin—Madison, Madison, WI 53706.

Uccellini, L. W. 1989. Process contributing to the rapid development of extratropical cyclones. Extratropical cyclones, Palmén Memorial Volume. Boston: American Meteorological Society (In Press).

Uccellini, L. W., D. Keyser, K. F. Brill and C. H. Wash. 1985. The Presidents' Day cyclone of 18–19 February 1979: Influence of upstream trough amplification and associated tropopause folding on rapid cyclogenesis. *Mon. Wea. Rev.* **113:** 962–988.

Whitaker, J. S., Uccellini, L. W. and K. F. Brill. 1988. A model-based diagnostic study of the rapid development phase of the Presidents' Day cyclone. *Mon. Wea. Rev.* **116:** 2337–2365. ●

A 4-D CASE STUDY FOR SYNOPTIC LABORATORY

Patricia M. Pauley
Department of Meteorology

William L. Hibbard and David Santek
Space Science and Engineering Center

University of Wisconsin-Madison
Madison, Wisconsin

"And let me say now that I hated the laboratory when I took the course, because I couldn't see the sense in spending so much time drawing maps.....in the end, I'm very grateful for having been exposed...so very directly to actual atmospheric phenomena.....I think a person who has made subjective analyses of weather maps has a deeper appreciation of how inadequate many objective analysis schemes are, and if you simply accepted the machine product as reality, it would be very dangerous.....But at the same time I would agree that machines...can be used very constructively to produce various kinds of analyses and so you could look at the essentials of the atmosphere."

Conversations with Jule Charney (Platzman, 1987)

1. INTRODUCTION

Instructors of synoptic meteorology have long been seeking to facilitate their students' understanding of the four-dimensional (three spatial dimensions and time) nature of the atmosphere. Traditional tools in the synoptics lab have centered on two-dimensional maps and cross-sections because of the difficulty of depicting three and four dimensions on flat sheets of paper. These same two-dimensional depictions are used in operational forecasting, reinforcing their use in the classroom. Isentropic analysis is sometimes used because of its implicit inclusion of the vertical dimension in a horizontal map format. However, students typically have difficulty understanding how to interpret features in an isentropic framework, a problem which is engendered by the lack of day-to-day operational analyses and guidance on isentropic surfaces. Three-dimensional graphics have also been used to some extent, but are generally limited in application to idealized representations of the atmosphere.

The objective of the project described in this paper is to facilitate the student's understanding of the 4-D nature of the atmosphere through a combination of traditional and computer-based methods. A series of plotted isobaric and isentropic maps for the extratropical cyclone case of 1-3 April 1982 is used to give the students experience in hand-analysis, the traditional stock-in-trade of the synoptics lab. The case study is then supplemented by a series of animated 2-D and 3-D graphics prepared on McIDAS (Man-computer Interactive Data Access System) from a model simulation of the same case.

This paper encompasses a brief description of the 4-D graphics available on the McIDAS system and a description of the case study materials. Some classroom results are also included.

2. THE MCIDAS 4-D GRAPHICS SYSTEM

The 4-D graphics system on MCIDAS consists of a set of programs which create stereo animation sequences of 3-D graphics from meteorological data. This system is described at length in Hibbard et al. (1987) and Hibbard (1986), so only a brief summary will be provided here. The 4-D programs are invoked by interactive commands which allow the user to specify the contents, scale, viewing angle, and other parameters of the display. This gives the user a ready means of interactively altering the graphics to emphasize whatever feature is of interest. The 4-D sequences are generated in five to fifteen minutes and displayed on a McIDAS workstation in stereo gray-shades or in "mono" color. The latter can be used to create video tapes such as the one presented at this workshop.

The McIDAS 4-D graphics system interfaces to a great variety of meteorological data through a 3-D grid data structure. The system depicts the data using contour surfaces of raw and derived scalar parameters, wind trajectories, topographic maps, contour lines of parameters drawn on pressure or height surfaces (including the earth's surface), and various other line graphics. The system has also been used to generate perspective displays of clouds from GOES satellite data, stacked CAPPI (Constant-Altitude Plan Position Indicator) scans from volumetric radar echoes, and planar slices generated from LIDAR data.

3. CASE STUDY DESCRIPTION

The case study used in this project is one of a rapidly intensifying extratropical cyclone with severe weather in an associated cold-frontal squall line, a storm with a bit of something for everyone. The central pressures of the cyclone are summarized in Fig. 1. During this time, the surface cyclone propagated from Colorado across Nebraska, southeastern South Dakota, southern Minnesota and central Wisconsin to a position near Green Bay at 09Z 3 April. The surface cyclone was supported by a short wave aloft which earlier had propagated around the base of a long-wave trough off the west coast. The short wave was subsequently embedded in broad southwesterly flow as it amplified in the lee of the Rockies. The initial LFM fields from 00Z 3 April are shown in Fig. 2, a time roughly in the middle of the deepening phase of the cyclone.

According to Storm Data (1982), 56 tornadoes were reported on April 2, including the first F5 tornado reported since 1977 (Fig. 3). Seventeen of these tornadoes were in the Red River region, including the F5 tornado and the F4 Paris (Texas) tornado which caused 10 fatalities and 170 injuries. Further severe weather occurred on April 3 bringing the total severe reports for the two days to 89 tornadoes, 42 reports of hail greater than or equal to 3/4", and 96 reports of severe-strength wind gusts, making this the worst outbreak of severe weather since the 3-4 April 1974 "Jumbo" outbreak. A dust storm was also associated with strong winds in the cold surge behind the system in the southern Plains in the afternoon of April 2, while significant snowfalls were reported in the northern Plains.

This case study was first used in undergraduate synoptics at the University of Wisconsin-Madison in the Fall 1987 semester. The undergraduate synoptics sequence at UW-MSN consists of two semester-long three-credit courses, typically taken by seniors after they have had one semester of thermodynamics and one of dynamics. The fall semester emphasizes extratropical cyclones, while the spring semester emphasizes severe local storms. The following plotted maps were prepared for this case; the students were given the maps indicated by (*) to analyze:

Surface charts (00Z, 06Z, 12Z, 18Z 2 April; 00Z, 09Z 3 April) (*)

Isobaric charts (12Z 1 Apr; 00Z, 12Z 2 April; 00Z, 12Z 3 April)
    850 mb (*)
    500 mb (*)
    300 mb (*)

Isentropic charts (same map times as isobaric charts)
    295K
    300K (*)
    305K
    315K

Derived fields (to be analyzed; same map times as isobaric charts)
    850 mb temperature advection (*)
    500 mb absolute vorticity advection (*)
    300 mb divergence (*)

Analyzed isentropic cross-sections
    12Z 2 Apr   stations 71119-72250

00Z 3 Apr	stations 71119-76692
12Z 3 Apr	stations 72747-72201
12Z 3 Apr	stations 71848-72201

Additional NMC products such as min/max temperature and composite moisture charts were also given to the students.

The 4-D graphics were prepared using fields from the Perkey-Kreitzberg Limited-Area Mesoscale Prediction System (LAMPS) model (Perkey, 1976) initialized at 00Z 2 April 1982 and run out to 36 hours. Frames are shown at two-hour intervals through 24 h after the initial time, except sequences 5, 6, and 7 which are through 36 hours. The sequences prepared were:

1. 2-D animation of sea-level pressure
2. 2-D animation of 500 mb heights and absolute vorticity
3. 3-D animation of sea-level pressure, 500 mb heights, and cloud water surfaces (0.1 g/kg)
4. 3-D animation of 300 mb heights, surfaces of absolute vorticity (16 x 10^{-5} sec^{-1}), and trajectories
5. 3-D animation of the 300K isentropic surface
6. 3-D animation of the 300K isentropic surface and trajectories
7. 3-D animation of the 300K isentropic surface with trajectories and mixing ratio surfaces (8 g/kg)
8. 3-D animation of isentropic surfaces from 292K to 328K, incrementing by 4K, and the 50 m/sec isotach surface for 12Z 2 April 1982
9. 3-D animation of the 310K and 340K isentropic surfaces and the 50 m/sec isotach surface for 12Z 2 April 1982, with the viewing angle changing from south to west

The terrain field is also depicted in 3-D in all of these sequences, in order to bring out any relationship between the depicted fields and the topography. Fig. 4 is a frame from sequence 7.

4. CLASSROOM RESULTS

In the Fall 1987 semester, the students first hand-analyzed their maps, then worked with the 4-D graphics while they were preparing papers describing the evolution of this particular cyclone case. While they were studying the 4-D graphics sequences, the students were asked to answer a set of questions which were designed to lead them through the graphics and direct their attention to specific aspects of each sequence. The graphics were displayed on the stereo McIDAS workstation, allowing the students to manipulate the graphics to a limited extent (change sequences, start and stop loops, single-step through loops).

A formal evaluation of the effectiveness of the 4-D graphics was attempted but proved inconclusive. A short test was given to the students after they had hand-analyzed their case-study maps but before they had worked with the computer graphics. The test included questions on quasi-geostrophic theory, vertical motion, isentropic coordinates, and map interpretation. A similar test was then given after they had used the computer graphics. An analysis of the test results showed no clear-cut effect of using the 4-D graphics---the students performed better on some questions, worse on others. The test may not have been properly designed; assessing the ability of a student to "think in 3-D" is difficult at best. A different set of maps was used for the map interpretation questions on the two exams, so the results may also reflect a difference in complexity in these cases. Furthermore, the results may also have been affected by the small sample size available (17 students).

On the other hand, a subjective evaluation showed that the 7 students who returned the evaluation felt that the 4-D graphics were valuable, especially in helping them to understand the isentropic perspective. All 7 students felt that the graphics helped them better understand the 3-D nature of the atmosphere; 5 out of the 7 felt that it helped them in writing their case study papers. The graphics also sparked a great deal of enthusiasm, which is valuable in and of itself. The students unanimously felt that the exercise was interesting; one thought it was "totally cool" and another felt it was "an experience of Biblical magnitude".

5. SUMMARY

The goal of the project described in this paper was to facilitate the synoptics student's understanding of the 4-D nature of the atmosphere through the use of a case study which combined traditional hand-analysis methods with 4-D computer graphics. A dramatic case of cyclone development over the U.S. with an associated severe weather outbreak was chosen in order to be able to explore a variety of processes. A model simulation of the case was used as the basis for the 4-D graphics; the model data have the advantage of higher time resolution to provide smoother animation sequences. Although an objective evaluation proved inconclusive, a subjective evaluation revealed that the students felt the exercise helped them better understand the 3-D nature of the atmosphere as well as stimulating their interest and enthusiasm.

6. ACKNOWLEDGEMENTS

The authors would like to thank Prof. Don Perkey, Dr. Pete Robertson, Dr.

Mike Kalb, and Mr. Mike Seablom for providing the LAMPS simulation used for this case study. Thanks is also given to Prof. Lyle Horn and teaching assistants George Phillips, Tom Zapotocny, and Brad Bramer for their work in preparing and using this case in the Fall 1987 synoptics class, and Mr. Steve Nieman for his assistance in programming the isobaric post-processor.

7. REFERENCES

Hibbard, W.L., 1986: Computer-generated imagery for 4-D meteorological data. Bull. Amer. Meteor. Soc., 67, 1362-1369.

Hibbard, W.L., R. Kraus, D. Santek, and J.T. Young, 1987: 4-D display of weather data on McIDAS. Proceedings of the Third International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, New Orleans, AMS, 89-93.

Perkey, D.J., 1976: A description and preliminary results from a fine-mesh model for forecasting quantitative precipitation. Mon. Wea. Rev., 104, 1513-1526.

Platzman, G.W., 1987: Conversations with Jule Charney, NCAR Tech. Note NCAR/TN-298+PROC.

Storm Data, 1982: Tornado Outbreak on April 2, 1982. Vol. 24, No. 4, pp. 4-8.

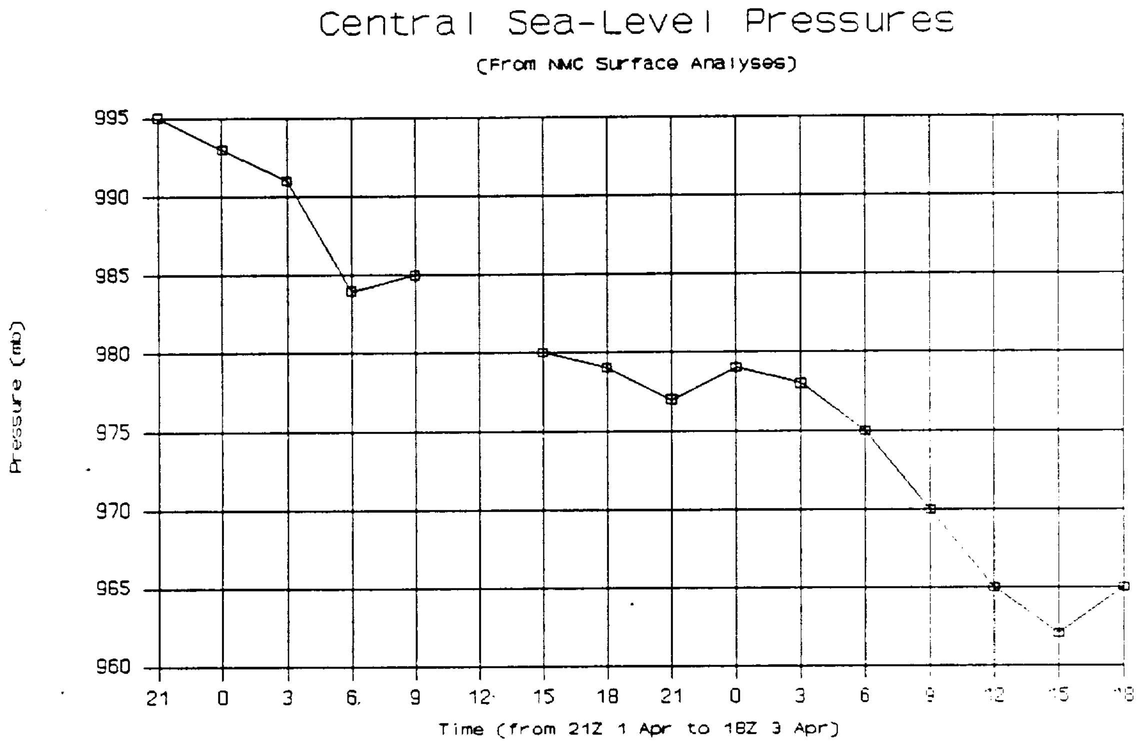


Figure 1: Central sea-level pressure trace for the 1-3 April 1982 extratropical cyclone, from the NMC 3-hourly surface analyses. (12Z 2 April is missing.)

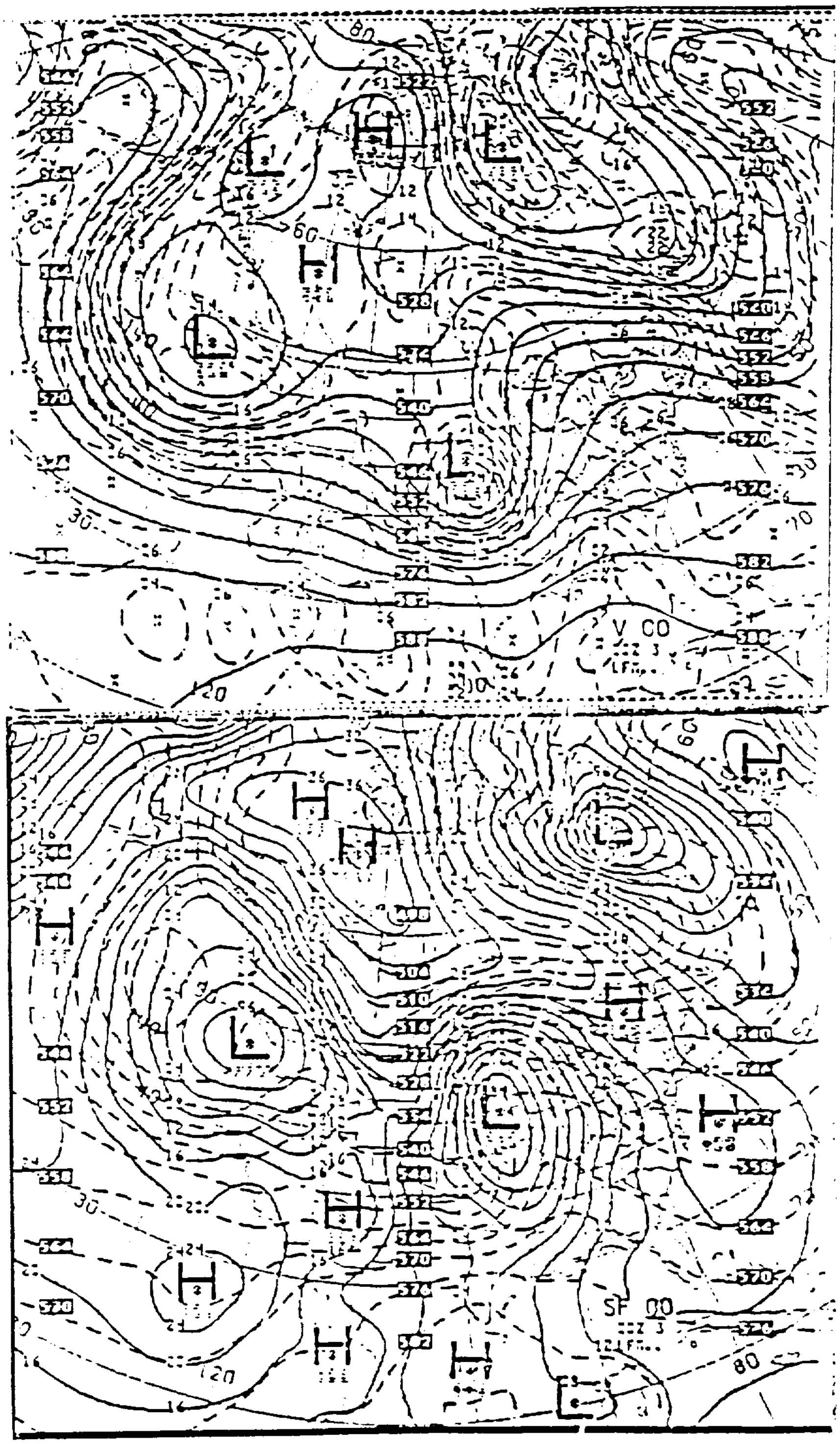


Figure 2: Initialized LFM fields for 00Z 3 April. Top: 500 mb heights and absolute vorticity. Bottom: MSL pressure and 1000-500 mb thickness.

Figure 3: Distribution of tornadoes on 2 April 1982. (from Storm Data, 1982)

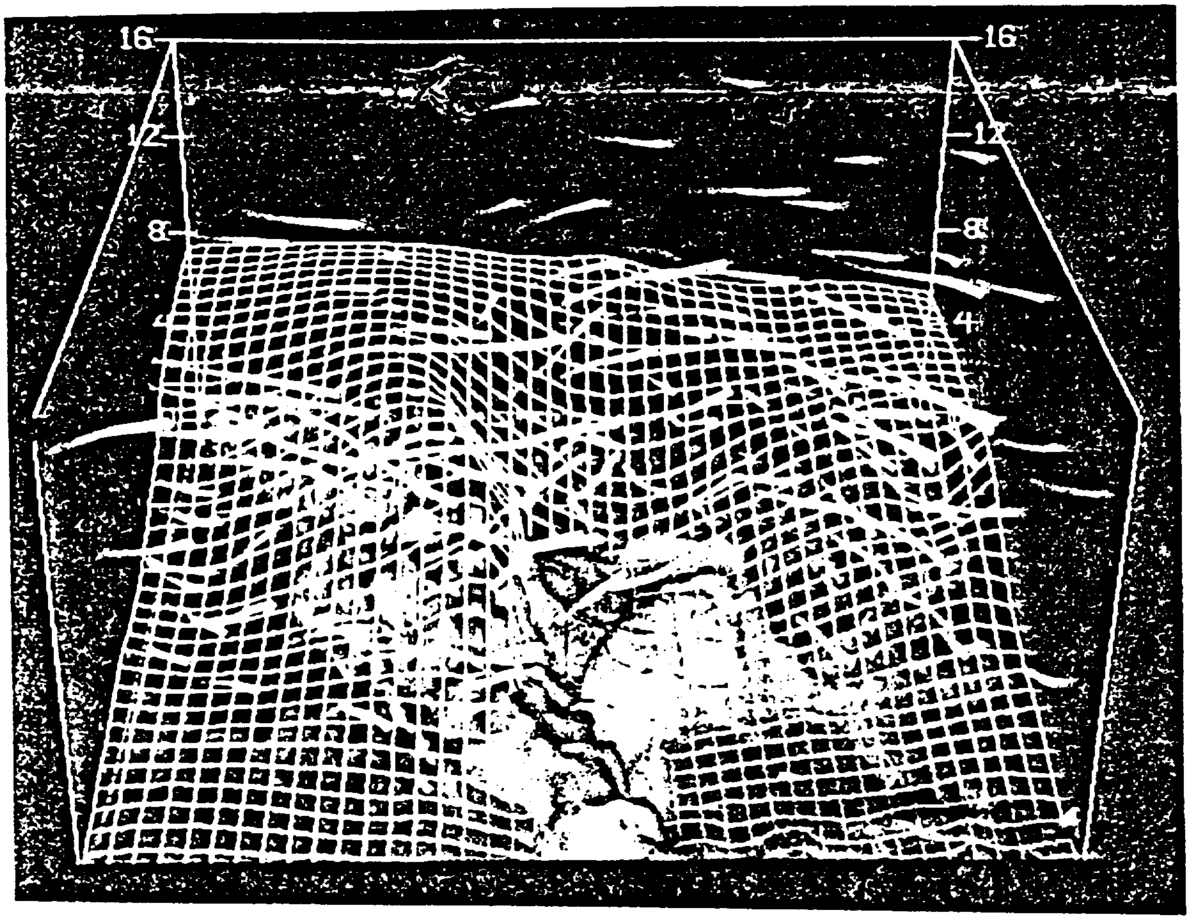


Figure 4: One frame from the 3-D animation of the 300K isentropic surface (mesh) with trajectories (ribbons) and 8 g/kg mixing ratio surfaces (semi-opaque). The vertical scale is in km.

THE USE OF 4-D GRAPHICS IN TEACHING SYNOPTIC METEOROLOGY

Patricia M. Pauley
Department of Meteorology

William L. Hibbard and David A. Santek
Space Science and Engineering Center

University of Wisconsin-Madison
Madison, Wisconsin

1. INTRODUCTION

One of the primary goals in teaching synoptic meteorology is to help the students learn how to take a variety of 2-D depictions and form a 4-D mental image of the atmosphere. The objective of the project described in this paper is to facilitate the student's understanding of the 4-D nature of the atmosphere through a combination of traditional methods (in which the student hand-analyzes isobaric and isentropic maps and cross-sections) and computer-based methods (in which the student can view and manipulate 4-D graphics as well as animations of 2-D fields).

During the current semester (Fall 1987), the synoptic laboratory students at the University of Wisconsin-Madison will be analyzing the extratropical cyclone case of 1-3 April 1982 as part of their class work. In the latter part of the semester, the case study will be concluded with an opportunity to view and manipulate a series of 4-D graphics from the same case. The familiarity with the case gained by hand-analyzing maps and cross-sections should enable the students to learn more from the computer graphics than would be possible otherwise.

The current project builds on previous work by Prof. Donald R. Johnson and his group in putting together educational videotape modules (e.g., Achtor et al., 1981a,b). These modules include a set of plotted maps to be analyzed by the students and a videotape of McIDAS (Man-computer Interactive Data Access System) satellite imagery overlaid with analyses of various fields obtained from conventional meteorological data. These modules have been very successfully used as the basis for synoptic laboratory case studies; the addition of satellite imagery and derived fields to traditional hand-analyses helps the students better understand the basic physical processes involved in the development of weather systems. It is hoped that the further addition of 4-D graphics will serve to increase the students' understanding even more, by showing the interrelationships between various traditional 2-D depictions of the atmosphere, as well as presenting the data in novel ways.

This paper encompasses a brief description of the 4-D graphics available on the McIDAS system, an outline of the progression of activities to be used in the classroom in conjunction with this project, and descriptions of sample products from this case study. As the case is being used during the current semester, pedagogical results are not yet available.

2. THE MCIDAS 4-D GRAPHICS SYSTEM

The 4-D graphics system on McIDAS consists of a set of programs which create stereo animation sequences of 3-D graphics from meteorological data. This system is described at length in Hibbard et al., 1987 and Hibbard, 1986, so only a brief summary will be provided here. The 4-D programs are invoked by interactive commands which allow the user to specify the contents, scale, viewing angle, and other parameters of the display. This gives the user a ready means of interactively altering the graphics to emphasize whatever feature is of interest. The 4-D sequences are generated in five to fifteen minutes and are displayed on a stereo McIDAS workstation.

The McIDAS 4-D graphics system interfaces to a great variety of meteorological data through a 3-D grid data structure. The system depicts the data using contour surfaces of raw and derived scalar parameters, wind trajectories, topographic maps, contour lines of parameters drawn on pressure or height surfaces (including the earth's surface), and various other line graphics. The system can also generate perspective depictions of clouds from GOES satellite data, stacked CAPPI (Constant-Altitude Plan Position Indicator) scans from volumetric radar echoes, and planar slices generated from LIDAR data.

For this project, the data were derived from a model simulation of the 1-3 April 1982 case over North America. The 24 hr simulation was run on the Limited-Area Mesoscale Prediction System (LAMPS) (Perkey, 1976) at Marshall Space Flight Center with 140 km resolution and was initialized at 0000 GMT 2 April 1982. The model output was saved at two-hour intervals and interpolated from the model's sigma-z vertical coordinate to both constant height and constant pressure surfaces. The height-level data form the basic data set for the 4-D graphics, while the pressure-level data permit the use of fields which are more familiar to the students. Model data were chosen for use with the 4-D graphics because of their better dynamical consistency and temporal resolution (Keyser and Uccellini, 1987). The behavior of the model simulation itself is also of interest in terms of discussing numerical weather prediction.

3. CASE STUDY DESCRIPTION

The case study developed by this project is being used during the current semester (Fall 1987) in the synoptic laboratory class (Meteor 452) at the University of Wisconsin-Madison. This course is typically taken by undergraduate senior meteorology majors after they have taken atmospheric thermodynamics and one semester of dynamics. During the fall semester, synoptic lab emphasizes extratropical cyclone development; during the spring semester, it emphasizes thunderstorms and severe weather. The different activities involved with this case study are spaced throughout the semester to agree in timing with material presented in lectures.

The 1-3 April 1982 case was chosen for this project because it not only represents a spectacular case of a developing extratropical cyclone over the data-rich northern Midwest in the U.S. (achieving a central pressure of 962 mb at 1500 GMT 3 April in northern Wisconsin), but also includes an extensive squall line along the trailing cold front. Thus, the case study is appropriate for use in both the fall and spring semester synoptic labs. The LAMPS model run used for the 4-D graphics was initialized at 0000 GMT 2 April, in order to encompass the period during which the cyclone occluded.

During the early part of the semester, the emphasis is placed on teaching the students subjective (hand) analysis. After performing some trial analyses, a series of computer-plotted maps for the case study, based on standard maps for the case study, based on standard rawinsonde data obtained from NCAR, is given to the students to analyze. These include surface maps, standard isobaric maps (850, 500, 300 mb), appropriate isentropic maps, and cross-sections.

As the semester progresses, the lecture material covers development theory for extratropical cyclones, including quasi-geostrophic theory. Sequences of 2-D analyses of various derived fields for the case (e.g., vorticity, divergence, advections of quantities) downloaded from the McIDAS mainframe will be displayed in the PC-McIDAS environment in the Department of Meteorology's computer classroom (Pauley and Meys, 1987).

Toward the end of the semester, the students will be assigned a paper based on the case study. They will be expected to discuss the actual development of the cyclone being studied in light of the various theories of cyclone development. At the time the paper is assigned, the 4-D imagery will be introduced to the class. After an initial presentation and orientation with the basic McIDAS commands needed to manipulate the imagery, the students will be allowed access to the stereo workstation to examine the imagery at their own pace. The satellite imagery for this case study will also be made available to the students on McIDAS.

4. EXAMPLES OF GRAPHICS

Although it is impossible to completely portray animations of stereo graphics with still photographs, three examples are included here to demonstrate the types of graphics used in this case study.

Figure 1 shows trajectories and cloud water envelopes over the surface terrain. The animated sequence shows the clouds increasing in size with time as the model "spins up". (The model has no initial cloud water.) The trajectories outline the circulation and also depict upward motion within the clouds. The orographic nature of the clouds over the Rockies is evident in this sequence. Trajectories may be combined with a variety of fields to demonstrate the relationships between the flow field itself and other atmospheric parameters.

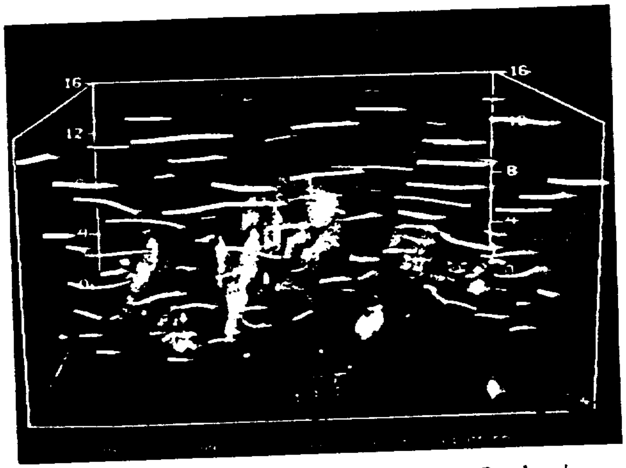


Fig. 1. Trajectories (ribbons) and 3-D cloud water surfaces for 0600 GMT 2 April 1982 above the surface topography.

The cloud water surfaces are shown with the sea-level pressure (SLP) field in Fig. 2. The SLP analysis is draped over the terrain to emphasize the role of the terrain. The relationship between the clouds and the surface flow field is brought out here.

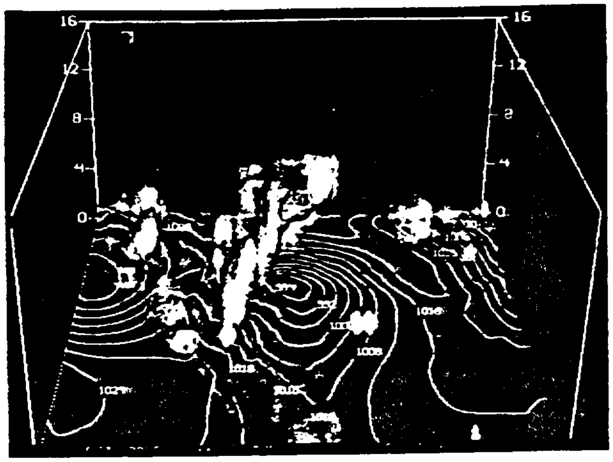


Fig. 2. Cloud water surfaces and the sea level pressure analysis for 0600 GMT 2 April 1982 above the surface topography.

The final image included in this discussion is one of the 350 mb height analysis and the 60 m/s isotach surfaces (Fig. 3). The relationship between the regions of tight height gradient and high wind speeds is apparent. The isotach surfaces further demonstrate the 3-D nature of the high-speed jet cores. Any arbitrary pressure analysis can be "floated" in the analysis volume in this fashion to portray the relationships between various conventional analyses and scalar surfaces or trajectories.

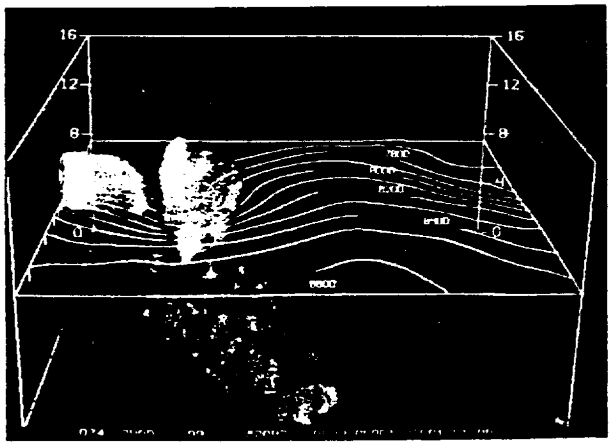


Fig. 3. 350 mb height analysis and 60 m/sec isotach surfaces for 0200 GMT 2 April 1982 above the surface topography.

5. SUMMARY

Experienced meteorologists have developed the skill of mentally visualizing the multivariate evolving 3-D geometry of weather phenomena. A major goal of the McIDAS 4-D system in general, and this project in particular, is to aid in this visualization through computer-generated images. The project described in this paper is a case study to be used in teaching synoptic meteorology, which includes traditional hand-analyses to be done by the students and computer-generated 4-D graphics sequences. This should make it an ideal tool for helping students develop their visualization skills.

6. ACKNOWLEDGEMENTS

The authors would like to thank Prof. Don Perkey, Dr. Pete Robertson, Dr. Mike Kalb, and Mr. Mike Seablom for providing the LAMPS simulation used for this case study. Thanks is also given to Prof. Lyle Horn and teaching assistants George Phillips, Tom Zapotocny, and Brad Bramer for their work in preparing and using this case in the synoptics laboratory class this fall, and Mr. Steve Nieman for his assistance in programming the isobaric post-processor and preparing this manuscript.

7. REFERENCES

Achtor, T.H., D.A. Edman, D.R. Johnson, R.S. Schneider, and C.H. Wash, 1981a: Cyclogenesis. Educational Modules for the Atmospheric Sciences, Space Science and Engineering Center, University of Wisconsin-Madison, 1225 W. Dayton, Madison, Wisconsin 53706.

Achtor, T.H., D.A. Edman, D.R. Johnson, R.S. Schneider, and C.H. Wash, 1981b: Squall line development. Educational Modules for the Atmospheric Sciences, Space Science and Engineering Center, University of Wisconsin-Madison, 1225 W. Dayton, Madison, Wisconsin 53706.

Hibbard, W.L., 1986: Computer-generated imagery for 4-D meteorological data. Bull. Amer. Meteor. Soc., 67, 1362-1369.

Hibbard, W.L., R. Kraus, D. Santek, and J.T. Young, 1987: 4-D display of weather data on McIDAS. Proceedings Third International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, New Orleans, Amer. Meteor. Soc., 89-93.

Keyser, D., and L.W. Uccellini, 1987: Regional models: Emerging tools for synoptic meteorologists. Bull. Amer. Meteor. Soc., 68, 306-320.

Pauley, P.M., and A.W. Meys, 1987: Computer-aided meteorological education at the University of Wisconsin-Madison. Proceedings Third International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, New Orleans, Amer. Meteor. Soc., 127-130.

Perkey, D.J., 1976: A description and preliminary results from a fine-mesh model for forecasting quantitative precipitation. Mon. Wea. Rev., 104, 1513-1526.

THE INTERACTIVE 4-D MCIDAS WORKSTATION

Bill Hibbard and Dave Santek
Space Science and Engineering Center
University of Wisconsin
Madison, Wisconsin U.S.A.

Marie-Francoise Voidrot-Martinez
French Meteorological Office
Paris, France

1. MCIDAS UNDER UNIX AND X WINDOWS

We have implemented a version of McIDAS (Man-computer Interactive Data Access System) running under UNIX and X Windows, on the Stardent GS-1000 workstation. This system goes a significant ways toward satisfying the goals described two years ago in "A Next Generation McIDAS Workstation" (Hibbard, 1988).

This McIDAS implementation is quite similar to previous McIDAS implementations, particularly PC-McIDAS. The file structures of this system for grids, images and other forms of meteorological data are identical with those of PC-McIDAS. All of the FORTRAN applications code is identical with PC-McIDAS, except for compiler differences. The system level code is written in C rather than assembler and consists of two C modules of 900 lines each. This McIDAS implementation supports variable sized frames dynamically created by the user, pan and zoom operations, and the flexibility to allocate video system bits to images versus graphics when McIDAS is started. Thus GOES IR data can be viewed in ten image bits overlayed by one graphic bit.

This McIDAS interfaces with the user through X Windows, providing a pseudocolor window for images and graphics and a number of Xterm windows for command input and text output. The new McIDAS function for interactive model visualization uses a true-color window for graphics and numerous widget windows for user control. McIDAS commands are entered to a standard UNIX shell, so all of the shell tools are available, including command history and shell scripts. The X Windows mechanism for cutting and pasting text is available to edit McIDAS output into text files. Numerous graphical user interface widgets are available.

An implementation of McIDAS using UNIX and X Windows, and avoiding the use of assembly language, should be easy to transport to other computer systems and to connect with other systems. The critical question for porting to another system is the performance of the X server on the target system for supporting animation.

2. INTERACTIVE 4-D VISUALIZATION

We have exploited the power of the Stardent workstation to create new McIDAS functions for interactive visualization of multivariate four-dimensional data sets, such as those produced by numerical weather models. These data sets are stored in five-dimensional arrays, composed of three space dimensions, time, and one dimension for enumerating different physical variables. For example, we have applied our system to visualize the ECMWF data set of 4 February 1988, which shows an extratropical cyclone moving across the North Atlantic during a one week simulation. Our software stores this data set in a five-dimensional grid of 24 latitudes by 46 longitudes (2.25 degree spacing) by 14 vertical levels by 168 time steps (one per hour for a week) by 8 variables (pressure, temperature, specific humidity, vorticity, divergence, potential temperature, vertical wind speed, and horizontal wind speed). This is a total of over 20 million grid points. Note that a workstation with 128MB of memory could be used to visualize a data set of 50 million grid points.

The basis of our visualizations is real-time animation. That is, the ability to generate the images as fast as they appear. This allows the user to interact with the display by controlling the contents of the images and getting immediate response. The user sees a three-dimensional box containing moving scenes generated from the weather data set. The scene may contain a topographical map, trajectory lines, and iso-level contour surfaces of scalar variables. The user can interactively select which of these visual elements to display at any moment. For example, the user can select the topographical map with a 280 Kelvin potential temperature surface and a 5 cm/sec vertical velocity surface. These will be animated to show the interactions between these variables. The user can interactively control the animation--start it, stop at an interesting time, and single step forward or backward to watch dynamics slowly. The user can interactively change the iso-levels for contour surfaces to see different values for variables. The surfaces for a selected variable are redrawn for all time steps, and this is done asynchronously with the animation, typically at a rate of about two per second. The user can also interactively rotate, pan and zoom the images in three dimensions. This is a very powerful way to understand complex geometries in three-dimensions.

3. COMMUNICATIONS

The Stardent GS-1000 supports the TCP/IP protocol over Ethernet, as well as NFS (network file system). These can be used to transfer files from another computer, or to provide a byte stream between applications on different machines. The OS/2 McIDAS includes applcations for transferring all the McIDAS file structures to and from the McIDAS mainframe. We will port these OS/2 communications programs to our UNIX implementation and link them with the raw data transfer via TCP/IP. This will be a simple task and will provide access to all the data sources of the McIDAS mainframe. We have no plans to port the McIDAS satellite data ingest software to the GS-1000.

TCP/IP is supported on almost all supercomputers, so this will provide an easy way to get model data into the GS-1000 for interactive four-dimensional visualization. The primary difficulty here is to write an application for translating the model output data format to our McIDAS file structures, a task which we have done for many different data sources.

Stardent has a commitment to support higher rate communications standards as the are defined. FDDI at 100 million bits per second is nearly settled, and there is active movement toward communications at one billion bits per second.

4. VISUALIZING DATA SETS OF BILLIONS OF POINTS

Our visualization work is based on the idea that the problem is the size of meteorological data sets and that the solution is highly interactive access to those data sets. The enormous size of the data sets gives the important information too much room to hide in. Also, the data sets are cumbersome to manage, hindering the effort to look at them in a variety of ways. However, advances in hardware and software systems are making it possible to process lots of data quickly, letting scientists make choices and get fast visual response to those choices. These tools let the user hunt quickly through a lot of data. Our software tools running on the Stardent workstation enable a scientist to interactively visualize a data set of 50 million grid points.

We have done a preliminary design for an extended version of our system which would provide interactive visualization of data sets of 5 to

10 billion grid points, 100 times larger than our current capability. This design distributes our current software onto both a Stardent workstation and a CRAY supercomputer, with the gridded data set residing on fast disks attached to the CRAY, and a very high speed link between the CRAY and the Stardent.

The distributed system could be used to visualize a data set such as the hurricane simulation which Greg Tripoli of the U. W. Meteorology Department is planning to create using his thunderstorm model (Tripoli, 1989). The hurricane data set will consist of 4 million spatial grid points by 10 physical variables by 2520 time steps (every 4 minutes for a week), and the simulation will require about 500 hours on a CRAY 2. The output data set will consist of 100 billion grid points which would need 400 billion bytes to store as 32 floating point values. This is an unrealistic amount of storage, so we would reduce the resolution of the model's output by 2 in space and time and compress the values to 8 bit integers. The resulting data set would have 1260 time steps and 500 thousand spatial grid points for a total of 6 billion bytes of storage.

The distributed system can be understood by looking at the sequence of operations involved in visualizing the hurricane data set. We would seek to produce real-time animations from this data set at about five frames per second. During each 0.2 second frame time, the system would execute the following steps:

a. The Stardent will send to the CRAY the user's controls for selecting which combination of physical variables to view, for selecting which time step to view (which 8 minute step out of a week), for selecting iso-levels for contouring variables, and for selecting the geographic extents of the region to view.

b. The CRAY will read the grids for the selected time step and physical variables from the disk. If we limit the number of simultaneous variables to three, this would be five frames per second by three variables by 500 thousand bytes per grid giving 7.5 million bytes per second of disk bandwidth. This transfer rate is achievable with supercomputer disk systems.

c. The CRAY will generate subgrids of about 50 thousand points each, by subsectoring and possible resolution blow-down, according to the user's selection of geographic extents.

d. The CRAY will generate polygonal contour surfaces from the subgrids for each selected physical variable, according to the user's selection of iso-levels. The surfaces will contain very roughly 50 thousand triangles.

This is a heavy computational load and will require polygon-finding algorithms adapted to exploit the parallel and vector facilities of the CRAY.

e. The CRAY will transmit the triangles to the Stardent. Fifty thousand triangles require 3.6 million bytes of storage, so the overall data rate would be 5 frames per second by 3.6 million bytes of triangles by 8 bits per byte giving 144 million bits per second. The current Stardent renderer requires that the triangles be compressed into polytriangle strips, with one vertex per triangle. This alters the storage to 1.2 million bytes per 50 thousand triangles and the data rate to 48 million bits per second. However, polytriangle strips present some problems for processing and image quality, so a solution without polytriangle strips is preferable.

f. The Stardent will render the triangles according to the user's selections for 3-D pan, zoom and rotation, surface color and transparency, and light source placement. This calls for the Stardent to render 5 times 50 thousand or 250 thousand triangles per second.

Note that the images generated will include a variety of visual elements such as wind trajectories and maps. They are much easier to handle than contour surfaces, so we have left them out of the above discussion. Also note that the distributed system could be applied to interactively visualize other weather and climate simulations of similar size.

5. INTERACTIVE MODEL DEVELOPMENT

We are also interested in using distributed algorithms to help scientists interactively develop thunderstorm simulations, a task requiring numerous trial and error adjustments to initial atmospheric conditions and to model parameters. This is done through an iterative cycle of short simulation runs, inspecting the model output data and comparing them to known storm behavior, and adjusting the initial conditions and model parameters.

Interacting with a running model is tricky because of man-machine coupling mismatches. Large simulations are much too slow for interaction, and we would concentrate on simulations with about 100 thousand spatial grid points. Even this size of simulation would progress too slowly to be directly visually interesting, so the visualization must be asynchronous with the model. The system will maintain an accumulating model output data set on the CRAY, and let the user move around in the time steps, rather than being in lock step with the model. To control cumulative error, the model needs to calculate many more time steps than the user wants to see. The storm model will calculate time steps for every two seconds of storm time but only store a time step in the data set for every minute of storm time. The user will visualize from the accumulating data set, using the same distributed software system that was described for visualizing the hurricane above. When the user wants to change the simulation, the system will enter a new operating mode allowing the initial conditions and model parameters to be changed and the model to be restarted at an earlier time step. This iteration will continue until the simulation behaves to the satisfaction of the scientist.

6. ACKNOWLEDGEMENTS

We wish to thank Dave Kamins and Jeff Vroom of Stardent for their help with this work. We also want to thank Verner Sumoi, Francis Bretherton, Robert Fox, James Dodge, Greg Wilson and Daniel Soderman for their support and encouragement. This work was supported by NASA/MSFC (NAS8-33799 and NAS8-36292).

7. REFERENCES

Hibbard, W., 1988: A next generation McIDAS workstation. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology, Anaheim, California, American Meteorology Society, 57-61.

Tripoli, G. J., 1989: The University of Wisconsin Regional Atmospheric Modeling System: Dynamical Framework and Simulation of Geostrophic Adjustment. In preparation.

3.21 UNIX AND X WINDOWS: THE RIGHT CHOICE FOR INTERACTIVE SYSTEMS

Bill Hibbard[1] and Dave Santek[1], Marie-Francoise Voidrot-Martinez[2]
Dave Kamins[3] and Jeff Vroom[3]

[1]Space Science and Engineering Center
University of Wisconsin
Madison, Wisconsin

[2]French Meteorological Office
Paris, France

[3]Stellar Computer, Inc.
Newton, Massachusetts

1. INTRODUCTION

We have chosen the UNIX operating system and X Windows as the basis for our Interactive 4-D McIDAS (Man-computer Interactive Data Access System) Workstation, which is being implemented on the Stellar GS-1000 Graphics Supercomputer. These choices are the overwhelming standards in the workstation market and are thus really independent of our choice of hardware. We are also planning to use TCP/IP for communications with data sources. The McIDAS running under UNIX has all of its low level system code written in the C language, in place of the assembler code used in previous McIDAS implementations. The applications are written in FORTRAN and are very compatible with other McIDAS implementations. The use of UNIX, X Windows, TCP/IP, C and FORTRAN makes this an easy McIDAS implementation, and also offers advantages for portability and for connectivity among McIDAS systems and with foreign systems.

The computer industry has built a Tower of Babel of incompatible machines, languages, operating systems and communications protocols, driven by the "not invented here" motive and by the profit motive. However, a growing group of users and vendors are trying to establish a set of standards, starting with science and engineering users and their workstations and supercomputers. Earth scientists will benefit greatly from adopting these standards for all of their interactive systems.

2. THE CHANGING NATURE OF WORKSTATIONS

In traditional interactive earth science systems, processors and workstations have been separate units connected by communications lines. The workstations consist of large image memories with video output generation circuits, and possibly a microcomputer acting as a controller. Our large software systems run on general purpose computers which communicate with the workstation at bandwidths of between 10 thousand and 10 million bits per second. In these traditional systems it is possible for the main processor, running the large software base, to be independent of the workstation, connected only by a serial line. Thus the workstation can change without changing the hardware platform for the large software investment.

New systems, such as our interactive 4-D McIDAS, are changing this. In order to give scientists complex real-time interaction with their large data sets, powerful processors must be integrated into the workstation. Current high end workstations include processors executing hundreds of millions of instructions per second. In these systems, the integrated processors will evolve with the workstation, threatening the large software investment we will make in them. Commercial standards will make it easier to port our software between workstation generations.

Traditional workstations have been capable of a limited set of real-time interactions, such as changing color enhancement, pan and zoom. These are provided with hardwired circuits and limited to their special functions. However, real-time interaction in three dimensions, or with the analyses underlying the display, require a generally programmable processor running complex software. The inclusion of a high speed general processor into a workstation is a complex task. The engineering development of the GS-1000 cost Stellar roughly 100 times as much as the development of the McIDAS Wide Word Workstation cost SSEC. This testifies both to the efficiency of SSEC's engineers and to the huge gulf separating workstations with and without integrated processors. We can no longer afford to design and support state-of-the-art workstations specifically for our earth science applications. We must use workstations from the commercial market, and their commercial software standards.

Workstations with integrated processors provide more flexible use for their large expensive memories than just image storage. The memory can store gridded data for interactive four-dimensional visualization, it can store state data when the processor is running a numerical simulation, and it can provide a very large data space for an interactive analysis.

3. DISTRIBUTED AND HETEROGENEOUS SYSTEMS

Distributed systems offer numerous advantages to earth scientists. An important lesson of our old McIDAS, based on a network of Harris minicomputers, was that a distributed system is more reliable than its parts. Despite numerous component failures, the distributed McIDAS at the National Severe Storms Forecast Center has never been down, except when it was being moved.

Distributed systems create hard partitions between machines being used for operations and for development. The operational users are protected from machine crashes due to testing new software and hardware interfaces.

Many functions of interactive systems are inherently geographically distributed, such as gathering data from diverse sources and providing data communications between users at different sites. A distributed system design is a more natural fit to such problems.

A distributed system allows dissimilar components, such as large and small workstations, large database machines, numerical modelling supercomputers, and special purpose data acquisition machines. Furthermore, a distributed system allows independent evolution of these different components.

In order to reap the benefits of distributed systems we need to adopt commercial software standards. These are the UNIX operating system, the X Windows video interface, TCP/IP and other true communications standards, as they emerge, and the C and FORTRAN languages. These standards minimize the effort of integrating our earth science application packages into the network system software. They do that by providing the same operating system (UNIX) and languages (C and FORTRAN) on all machines in the network, by providing the uniform and network transparent user interface of X Windows, and by providing a network protocol (TCP/IP) which is supported by most machines and fits naturally with UNIX and X Windows.

4. UNIX, X WINDOWS, TCP/IP, C AND FORTRAN

UNIX is a general purpose operating system available for virtually all computers, and it is the only operating system available for many workstations. UNIX was originally written to support systems programmers and suffered from performance problems. Now however, those problems have been solved, and UNIX is the standard on supercomputers and high end workstations where performance is important. UNIX is somewhat complex to learn, but so is any real operating system. As we learn the system, we are impressed by the powerful tools it provides for developing software.

X Windows is a video and user interface system which is available for a wide range of workstations. Networking is integrated into X Windows, so that a user at a workstation may be running applications on many machines simultaneously, communicating with each in a different window. X Windows is complex, primarily because it is a fairly successful attempt to provide a universal interface to cover the huge range of video system architectures. It supports operations on graphics, images, color enhancement, keyboard input, mouse input and interprocess communication.

TCP/IP is a communications protocol standard which runs on several types of network hardware, and which has been implemented on a large number of computers. It is sometimes criticized for limiting bandwidth. However, TCP/IP can deliver bandwidth one or two orders of magnitude greater than what we currently require for connections to McIDAS workstations. As new standards emerge at higher bandwidths, they should be adopted for interactive earth science systems.

C is a high level programming language which also provides some of the flexibility of assembly language. It is the "native" language of UNIX and X Windows, and has taken the place of assembly language for our UNIX and X Windows implementation of McIDAS. FORTRAN has been the programming language of science, and there is no good reason to replace it in our earth science applications. It is a standard as universal as UNIX and has been a benefit to the sciences for many years.

5. STANDARDS FOR THE BENEFIT OF EARTH SCIENCES

Most institutions in the earth sciences own a variety of computers running many different operating systems. However, the size of earth science problems requires large groups of people from several institutions to work together. The variety of systems is a real barrier to this collaboration. By adopting standards the earth science community can help remove this barrier. We need a uniform application program interface, so that applications software can be ported between systems easily. We also need a uniform user interface, so that people can use each other's systems. The standards of UNIX, X Windows, TCP/IP, C and FORTRAN provide a start at these uniform interfaces. Of course, there are areas of these interfaces which are not settled and where we must wait for further standards to emerge. The important thing is a general movement towards standards in the earth science community.

6. REFERENCES

Dueck, D., and G. Wells, 1988: An Evolutionary Approach to Workstation Development. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Anahiem, American Meteorology Society, 53-56.

Gardiner, B., W. Bergen, and M. Mathewson, 1989: New Design Concepts for the Development of Meteorological Workstation. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Anahiem, American Meteorology Society, 6-9.

Hibbard, W., 1988: A next generation McIDAS workstation. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Anahiem, American Meteorology Society, 57-61.

Kernighan, B., and R. Pike, 1984: The UNIX Programming Environment. Prentice-Hall, 357 pp.

Kernighan, B., and D. Ritchie, 1978: The C Programming Language. Prentice-Hall, 228 pp.

Xlib Programming Manual for Version 11 Release 2 of The X Window System, Volumes 1 and 2, 1987. O'Reilly & Associates.

ORIGINAL PAGE IS
OF POOR QUALITY

3.22 A UNIX AND X WINDOWS IMPLEMENTATION OF MCIDAS

Dave Santek[1] and Bill Hibbard[1] , Marie-Francoise Voidrot-Martinez[2]
Dave Kamins[3] and Jeff Vroom[3]

[1]Space Science and Engineering Center
University of Wisconsin
Madison, Wisconsin

[2]French Meteorological Office
Paris, France

[3]Stellar Computer, Inc.
Newton, Massachusetts

1. INTRODUCTION

We have implemented a version of McIDAS (Man-computer Interactive Data Access System) running under UNIX and X Windows, on the Stellar GS-1000 Graphics Supercomputer. This system goes a significant ways toward satisfying the goals described at this conference two years ago in "A Next Generation McIDAS Workstation."

The UNIX and X Windows McIDAS includes keyins, macros, a traditional McIDAS video interface, areas (image files), LW (long word) files, MD (meteorological data) files, grids, 3-D grids and on-line help. The source code for applications is very compatible with other McIDAS implementations. The UNIX version extends the usual McIDAS functions to include dynamic creation and deletion of variable sized image frames, real-time pan and zoom within images, and allocation of video bits to images versus graphics on McIDAS startup. It also includes a new set of commands for interactive visualization of large gridded model output data sets, using three-dimensional images. Running on the Stellar GS-1000 it has the performance to run large numerical weather models.

This has been a relatively easy McIDAS implementation because we exploited the existing tools of UNIX and X Windows as far as possible. We plan to use TCP/IP for communications, and expect that this will be relatively easy, by using the existing functions of TCP/IP as much as possible.

An implementation of McIDAS using these software standards, and avoiding the use of assembly language, should be easy to transport to other computer systems, and to connect with other systems. The critical question for porting to another system is the performance of the X server on the target system.

2. THE USER'S VIEW

The commands and data structures of our UNIX McIDAS are virtually identical with other McIDAS implementations. The system includes new commands, but the existing commands have not changed. Macros and string tables are the same as on other McIDAS implementations. The video system (which is implemented in a window) is similar to the traditional McIDAS workstation, with images overlayed by graphics and cursor, independently enabled and looped. The video system is controlled by the same short commands (for example LB to set loop bounds, DR to set dwell rates, EG to erase graphics, etc.) and single keystrokes (for example A to advance a frame, L to enable and disable looping, K and W to enable and disable images and graphics, T and E to report frame and earth cursor positions, etc.) Thus this McIDAS should feel familiar to users of other McIDAS implementations.

The biggest change in our user interface is the use of multiple windows and the way the user manages those windows. When the user logs on to the system it creates a McIDAS video window and one or more McIDAS command windows. The command windows accept McIDAS commands and show text output. The video window accepts single keystroke controls and shows images and graphics. The user's text input is routed to whichever window contains the cursor, according to the X Windows mechanism for keyboard focus. The user moves the cursor by moving the mouse. Thus the mouse is the mechanism for selecting between single keystroke control of the video and entering McIDAS commands, and for selecting between different McIDAS command windows. When the cursor is in the McIDAS video window, the mouse is also used to control panning and zooming of the images and graphics.

In applications where it is desireable to have separate screens for video output and McIDAS command input, the Stellar GS-2000 (which is totally compatible with the GS-1000) has an option for multiple display screens which would make a more familiar McIDAS two screen system possible.

In previous McIDAS implementations, the image and graphics frames are all of fixed size and exactly fit into the display screen. In the X Windows implementation, however, the frames can be dynamically created (and deleted) with any reasonable size, and the video window can be dynamically resized within the screen. The alignment of frames to the video window can be moved by panning with the left mouse button pushed, and the frame magnification can be dynamically adjusted between one, two and four times zoom with the right mouse button. X Windows provides simple mechanisms for dynamically moving and resizing all of the windows on the screen, for creating new windows, and for controlling which windows are visible when windows overlap. With a little practice a user becomes adept at these controls, and the way they fit with McIDAS commands and single keystroke video controls.

The user has the option, when McIDAS is started, to adjust the number of bits per pixel for images and graphics. On the GS-1000 the default is eight bits of depth for images and three bits for graphics, and any choice whose sum is no greater than 11 is legal. Thus the GS-1000 McIDAS could be used to display 10-bit IR data from GOES with a one bit graphic.

Because McIDAS commands are entered to a standard UNIX shell, all of the shell tools are available to users who want to exploit them. This includes the history mechanism for retrieving and modifiing previously entered commands, shell scripts for writing high level programs which invoke McIDAS commands, aliasing for creating short versions of commonly used commands, and many other mechanisms. In situations where the power of UNIX commands may be dangerous for users, it is possible to disable all commands except McIDAS commands. The X Windows software also provides powerful tools which users may want to exploit. The user may scroll back over hundreds or thousands of lines of output in a McIDAS command window. Sections of this output can be retrieved using the cursor, and entered into another window, for example for insertion into a document in a word processor.

3. MAJOR EXTENSIONS

The UNIX McIDAS includes new functions to exploit the power of the Stellar GS-1000. These include interactive four-dimensional visualization, image processing, numerical models, and graphical user interface.

The four-dimensional visualization includes a set of commands for managing and displaying large gridded data sets such as are produced by numerical weather models. The display shows a three-dimensional scene containing a box, a topographical map, wind trajectories and contour surfaces of scalar variables. The user can interactively rotate and zoom in three dimensions, can control the combination of displayed variables, can start and stop time animation and single step time forward and backward, and can change the iso-level of contour surfaces. On a 64 megabyte system the user can interactively visualize a data set of 20 million grid points. On a 128 megabyte system this can be up to 50 million grid points.

Image processing functions include the pan and zoom mentioned earlier. Another function is being developed for mapping images onto surfaces in three dimensions with interactive rotation, which could be useful for radar scan cones or multiple lidar slices. The power of the GS-1000 and GS-2000 will make it possible for a large variety of image processing operations to be defined as ordinary McIDAS commands and to be performed at interactive speeds.

Greg Tripoli of the University of Wisconsin Meteorology Department has ported the RAMS (Regional Atmospheric Modelling System) model to the GS-1000. The model can be run locally on the workstation and get better turnaround than is available on many shared supercomputers. Because the history output is stored locally on the workstation, it can be immediately visualized using the interactive four-dimensional tools.

We use graphical widgets to control the user interaction with the four-dimensional visualization. These widgets, which are part of Stellar's Application Visualization System (AVS), include push buttons and sliders which can be made to appear and disappear dynamically. We have also developed our own widgets, such as one showing a grey wedge and a set of red, green and blue graphs, which the user can redraw with the cursor to alter image enhancement tables. When applied thoughtfully these tools can make software much easier to use and improve interactivity. We plan to use these tools to create alternate mechanisms for controlling McIDAS, in addition to the usual McIDAS text commands.

4. IMPLEMENTATION

The overwhelming majority of this McIDAS implementation consists of FORTRAN modules which are nearly identical with the OS/2 implementation of McIDAS. The UNIX and X Windows specific parts are limited to a few short shell scripts and two C modules of about 900 lines each (this does not include the interactive four-dimensional display command, which is a major McIDAS extension). One of the C modules is the source for the McIDAS process, which creates and manages the McIDAS video window and frames and the shared memory region used for communication with McIDAS commands. The other C module is linked with all McIDAS commands and provides low level support for retrieving command parameters, file access, writing images and graphics, and communicating via shared memory. The shared memory segment is used for communicating X Windows resource ID's, frame sizes, the current enhancement tables, and the traditional McIDAS user common.

The McIDAS process uses X Windows resources to emulate the traditional McIDAS video interface. The screen is created as a pseudo-color Window and frames are created as Pixmaps. The McIDAS enhancement tables for images and graphics are combined into four different Colormaps, one for each combination of images and graphics enabled or disabled. The GS-1000 has 12 bits for pseudo-color, but because some of these are allocated beyond McIDAS, we use 11 bits for images plus graphics. We allocate a contiguous interval of pixel values between 1024 and 3071, to avoid allocations at the top and the bottom of the 12 bit range. The sections of code dealing with this color allocation may need to change if this McIDAS is ported to a different computer.

The independence of images and graphics is implemented using the XCopyArea call and manipulating the function and plane mask in the graphics context. Animation and panning are implemented by repeated calls to XCopyArea, changing the source Pixmap and the source coordinates. This works because of the performance of the X Server on the GS-1000, and X Server performance is a critical question for

ORIGINAL PAGE IS
OF POOR QUALITY

porting this McIDAS to another computer. Zoom is implemented with a GS-1000 extension to X Windows, and could only be ported by finding an equivalent extension.

The McIDAS process uses XNextEvent to retrieve mouse and keyboard events directed at the McIDAS video window. The mouse events are used to control pan and zoom, and the keyboard events are interpreted as single keystroke video controls.

McIDAS commands are entered to the C shell in xterm windows. They invoke shell scripts which invoke the command with an ampersand appended after the arguments, causing commands to run in the background. Text output from commands is written to an xterm window. Text output from the McIDAS process, for example to report the earth coordinates of the cursor, are routed to one of the xterm windows.

McIDAS LW files are implemented with the standard UNIX calls to lseek, read, write and open. All other McIDAS file structures, including image areas, are implemented using LW files. These other file structures are implemented in FORTRAN with code that is identical to the OS/2 McIDAS.

5. COMMUNICATIONS

The Stellar GS-1000 supports the TCP/IP protocol over ethernet, as well as NFS (network file system). These can be used to transfer files from another computer, or to provide a byte stream between applications on different machines. The OS/2 McIDAS includes applcations for transfering all the McIDAS file structures to and from the McIDAS mainframe. We will port these OS/2 communications programs to our UNIX implementation and link them with the raw data transfer via TCP/IP. This will be a simple task and will provide access to all the data source of the McIDAS mainframe. We have no plans to port the McIDAS satellite data ingest software to the GS-1000.

TCP/IP is supported on almost all supercomputers, so this will provide an easy way to get model data into the GS-1000 for interactive four-dimensional visualization. The primary difficulty here is to write an application for translating the model output data format to our McIDAS file structures, a task which we have done for many different data sources.

6. CONCLUSIONS

UNIX and X Windows are the overwhelming standards for the high performance workstations which are necessary for very interactive systems. Our UNIX and X Windows implementation of McIDAS runs on the Stellar GS-1000 Graphics Supercomputer giving earth scientists highly interactive access to their large data sets. This implementation combines the traditional power and flexibility of McIDAS with interactive four-dimensional visualization, interactive image processing, and supercomputer performance for running models.

By exploiting the existing functions of UNIX and X Windows, this was an easy implementation of McIDAS. It is also one that should be easy to transport to other computers and to connect to other systems.

7. ACKNOWLEDGMENTS

We wish to thank Scott Mindock, Dan Adams, John Benson and J. T. Young of SSEC and Larry Gelberg and Brian Kowalski of Stellar for contributing ideas to this work.

8. REFERENCES

Dengel, R., J. Benson, R. Lord, J. Plutchak, D. Santek, S. Shirshac, T. Whittaker and J. Young, 1989: An OS/2 implementation of a PCDbased McIDAS Workstation. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Anahiem, American Meteorology Society, 41-43.

Hibbard, W., 1988: A next generation McIDAS workstation. Preprints, Conf. Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology. Anahiem, American Meteorology Society, 57-61.

Kernighan, B., and R. Pike, 1984: The UNIX Programming Environment. Prentice-Hall, 357 pp.

Kernighan, B., and D. Ritchie, 1978: The C Programming Language. Prentice-Hall, 228 pp.

Xlib Programming Manual for Version 11 Release 2 of The X Window System, Volumes 1 and 2, 1987. O'Rielly & Associates.

The VIS-5D System for Easy Interactive Visualization

Bill Hibbard and Dave Santek

Space Science and Engineering Center
University of Wisconsin - Madison

Abstract

The VIS-5D system provides highly interactive visual access to 5-dimensional data sets containing up to 50 million data points. The user has easy and intuitive control over animated 3-dimensional depictions of multiple interacting physical variables. VIS-5D is runs on the Stardent ST-1000 and ST-2000 workstations and is available as freeware from the Space Science and Engineering Center.

The VIS-5D System

We wrote the VIS-5D software system to help earth scientists understand their large and complex data sets. VIS-5D runs on the Stardent ST-1000 and ST-2000 workstations and generates animated 3-dimensional graphics from gridded data sets in real time. It provides a widget-based user interface and fast visual response which allows scientists to interactively explore their data sets. VIS-5D generates literal and intuitive depictions of data, has user controls which are data oriented rather than graphics oriented, and provides the WYSIWYG (what-you-see-is-what-you-get) response familiar to users of word processors and spread sheets. The result is a system which is easy for scientists to use, so that they can become the producers and directors of their own animations. VIS-5D can be applied to any data set in the McIDAS grid file format and containing up to 50 million grid points. Data sets containing hundreds of millions of grid points can be resampled to this 50 million point limit for interactive visualization.

We were motivated to write VIS-5D by our experiences using our 4-D McIDAS system running on IBM mainframe computers [1]. The 4-D McIDAS system generates 3-dimensional images in about 30 seconds each, with another 30 seconds each to load the images into a workstation for animation. We used this system to produce animated visualizations for many earth scientists. They were constantly wanting to change the animations and frustrated by the turnaround time. They found the video tapes we produced useful for public presentations and for teaching, but not for their own insight into their data sets, which they continued to get from the 2-dimensional graphics systems which they could use directly. Thus we wrote the highly interactive VIS-5D system which makes 3-dimensional graphics easy for scientists to use directly [2].

The VIS-5D system is available from the University of Wisconsin Space Science and Engineering Center as freeware. In addition to the visualization software, it includes tools for managing and analyzing large gridded data sets, a skeleton program for converting external data to the McIDAS grid file formats, documentation on how to use the software, and sample data sets to practice using the software.

Five-dimensional Data Sets

VIS-5D works with data in the form of a 5-dimensional rectangular grid of points. In a FORTRAN or C program these data sets could be declared as arrays with five dimensions. Three of the dimensions are spatial, one is time, and one is used to enumerate multiple physical variables. Thus these data sets sample a spatial volume at a regular lattice of points, sample dynamics at multiple steps over a time interval, and include multiple interacting physical variables.

Although a 5-dimensional grid may seem like a specialized format, it is the usual format for output from atmospheric simulations. It is also a common output

CH2913-2/90/0000/0028/$01.00 – 1990 IEEE

format for oceanography and hydrology simulations, and for some remote sensing instruments like radars which can scan quickly enough to produce time varying volumetric images. The most general setting for the earth's physical systems is multiple variables over three spatial dimensions plus time. Our 5-dimensional rectangles are just this setting, subject to discrete and uniform sampling of space and time. Thus the data format for VIS-5D is actually widely applicable to earth science data. For meteorological data the spatial dimensions are often latitude, longitude and altitude and the variables might be temperature, pressure, moisture and three wind vector components. For oceanography data the spatial dimensions are latitude, longitude and depth and the variables might be temperature, salinity, density and three ocean current vector components. For hydrology data the variables might be proportions for different rock and soil types, and three ground water flow vector components.

The VIS-5D system provides a high degree of interaction by storing the entire data set in the main memory of the workstation. Because of its compressed formats, this can be up to 50 million grid points. For example, these 50 million points can be factored as 50 latitudes by 50 longitudes by 20 altitudes by 100 time steps by 10 different physical variables. VIS-5D includes functions for managing much larger disk based data sets, and for resampling them down to smaller extents or to lower resolution in order to fit the size limit for interactive visualization. A factor of 2 reduction in resolution in time and space yields a 16 times reduction in data volume, which would allow a simulation data set of 800 million points to be reduced to the 50 million point limit.

The VIS-5D system supports a data format for trajectory paths, which are used to represent wind, ocean currents, ground water flow, and other motion fields, and a global topographical map data set, which is useful for large scale earth based data sets. The system includes a program for calculating trajectories from gridded motion vector fields, and management functions for listing, copying, merging and deleting data sets.

What Scientists Need

We accumulated considerable experience producing visualizations with scientists using our 4-D McIDAS system. Because 4D McIDAS requires an hour or more to produce each new animation sequence, this experience gave us an understanding of which types of changes to an animation sequence earth scientists really care about. These are:

A. change the viewpoint in three dimensions.

B. change the combination of simultaneously depicted variables.

C. change the depiction of a variable. For an iso-level contour surface this is a change to the defining value. For contour lines on a surface this is a change to the position of the surface and the density of the contour lines. For trajectory lines, this is a change to the density of the lines or placing trajectories through specific points.

D. change the time dynamics. This includes the choice of whether to enable time stepping, whether to step forward or backward (useful for tracing effects back to their causes), and how fast to step.

E. change the spatial extents of the depicted region.

F. calculate new variables from existing variables, including arithmetic, differential and integral operators.

G. make objects semi-transparent.

H. avoid depicting different variables using the same color.

It is worth noting that the types of controls the scientists care about relate to data rather than graphics. The value system of scientists is very different from the value system of film and television producers. Scientists need to clearly perceive 3-dimensional geometry and time dynamics, and thus require some minimum standards of graphics quality. However, the benefits of advanced photorealistic techniques are often outweighed by the negative impact of their computational difficulty on system response time to user changes. Of course, many of the controls scientists care about are also computationally difficult, and this results in compromises in the user interface.

The VIS-5D User Interface

The goal of VIS-5D is to provide the scientist with an easy user interface for controlling the display, and fast visual response to changes. The workstation should feel like a steerable window which the scientist "flys" through a huge data set, hunting for interesting information hiding in the mass of data.

VIS-5D is able to present a simple and intuitive interface to the scientist because it deals specifically with the 5-dimensional rectangles of data produced by environmental simulations, because it generates very literal data depictions, and because it concentrates on the data oriented choices which scientists want.

The figure below shows a scene generated by VIS-5D which is part of a video depicting cold fronts moving across the North Atlantic. The data are taken from a forecast for February 4, 1988 by the European Center for Medium-range Weather Forecasts. The scene shows a topographical map, a transparent specific humidity contour surface at 6.38 grams per kilogram, and pressure contour lines at an altitude of 0.89 kilometers with a spacing of 5 millibars. The small clock hand in the upper left corner of the 3-D window shows where the scene is within the data set's time span. The highlighted widget buttons on the left show that the map, Q (specific humidity) and P (pressure) are enabled for display. The slider widgets show the level of the Q contour surface, the altitude and

Figure 1 *(Color Plate 14, page 462)*

spacing of the P contour lines, and the color (white) and transparency (0.55) of the Q contour surface.

The three spatial dimensions of the data set rectangle are depicted with a single 3-dimensional box, and all the graphical elements depicting the data set are drawn in this common frame. The 3-D box contains either a static depiction of a single time step, or an animated depiction sequencing through the data set's time steps. The graphical elements of the depiction may include a topographical map, trajectory lines, and iso-level contour surfaces and contour lines for the data set's physical variables, all drawn in the common geometry of the 3-D box. The user can rotate, zoom and pan the 3-D box, control the time stepping, and independently enable or disable the depicted graphical elements. This provides a very literal representation of the 5-dimensional rectangle; the spatial dimensions are mapped into the 3-D box, the time dimension can be animated, and the physical variables can be viewed in arbitrary combinations.

The intuitive feel of the user interface is further enhanced by the fast response of VIS-5D to user controls. The 3-D box rotates, the time steps, and the depictions of variables appear and disappear, all within a fraction of a second of the appropriate mouse movement or button click. This immediate visual feedback is a critical element of a word processor's or spread sheet's user interface, and it is even more crucial for scientific visualization systems.

The McIDAS grid file format includes information specifying the time and location of each grid point, and the names of the physical variables in the data set. VIS-5D uses this information to generate a set of graphical widgets appropriate to the data set, and to automate the management of the components of the data set. For example, VIS-5D creates widgets for each physical variable, labelled with the names taken from the grid file, which are used to independently enable and disable graphical depictions of the variables. It also creates sliders for each variable used to change the values of their iso-level contour surfaces, and to change the altitude and density of their contour lines. A change to a slider value is applied to the appropriate variable for all the data set's time steps.

Changes to defining levels of contour surfaces and contour lines require a compromise in the user interface, because of the computational difficulty in computing new polygons and vectors for their graphical depictions. Ideally, the graphics would change as the user moved the slider, and VIS-5D does achieve this for changing altitude of contour lines on some data sets. However, new contour surfaces and complex contour line sets may require a couple of seconds for computation. When time dynamics are static, the new surface replaces the old surface as soon as it is computed. When time dynamics are animating, the new surfaces appear asynchronously with the animation sequence, gradually replacing the surfaces for all the data set's time steps.

VIS-5D provides pop up slider widgets which allow the user to change the color of contour surfaces and lines and the transparency of surfaces. When a data set includes ten different variables, each depicted by both surfaces and lines, it is hard to avoid multiple graphical elements with similar colors. With the color widgets the scientist can adjust the colors as variables are viewed in different combinations. The widget buttons used to enable graphical elements for display are highlighted with the color of the corresponding surface or lines, to help the scientist identify which graphics depict which variables.

We have avoided other graphical choices in our user interface. For example, there is a single light source which is always placed pointing along the view axis (actually there is a second light source pointing the other direction on the same axis to accommodate either sign of surface normals). The surfaces are drawn according to a Gouraud shading model with fixed properties. In our experience these choices are less interesting to scientists and they tend to clutter up the user interface. Interactive rotation is a very powerful way to understand 3-D geometry, and these other controls offer only marginal improvement. Surface properties like specular highlights may actually be counterproductive by slowing the response time to interactive rotation. Surface property techniques like texture mapping can be useful when they are used to add data content to the display, although we have not yet included texture mapping into the VIS-5D system.

The Structure of VIS-5D

In order to maximize the size of the data set for visualization, VIS-5D uses a compressed format for the raw gridded data and the large polygon and vector lists used to represent contour surfaces and lines. The natural format of these data is 4 byte floating point, but they can be quickly compressed by a linear mapping into 1 or 2 byte integers, depending on the needed resolution. This compression allows a data set of 50 million grid points plus its associated polygon and vector lists to fit in 128MB of workstation memory.

The principal data structures of the VIS-5D visualization program include:

A. a 5-dimensional array of bytes, containing a compressed version of the raw gridded data. This array is organized into a series of 3-D spatial arrays indexed by time-step and variable. Each variable has a separate linear mapping for compression from its range of values to 1 byte integers.

B. a linear array which is dynamically allocated for the polygon and vector lists used to represent contour surfaces and contour lines. There is a polygon list and a vector list for each combination of time-step and variable (some lists may be empty), and an index by time-step and variable into the linear array. Vertex components are compressed by a linear mapping from the box extents to 2 byte integers, and normal components are compressed by a linear mapping from the interval (-1.0, 1.0) to 1 byte integers.

C. vector lists for trajectory lines. Each trajectory is stored as a single poly-vector with an index by time-step into the poly-vector.

D. a polygon mesh for the topographical map and vector lists for the map boundary lines.

E. a queue containing time-step and variable indices identifying 3-D grids for which contour surface polygon lists or contour line vector lists need to be computed.

F. arrays of values of iso-level contour surfaces and altitudes and densities of contour lines, indexed by variable.

G. colors and transparencies for contour surfaces and colors for contour lines, indexed by variable.

H. state information for the display, including the current time-step, whether animation is enabled, whether the map is enabled, whether the trajectories are enabled, and the transformation matrix for the 3-D to 2-D projection. This state information also includes arrays indicating whether contour surfaces and contour lines are enabled, indexed by variable.

I. ordered lists of variables recording which contour surfaces and lines have been most recently enabled for display.

J. intermediate structures used for computing contour surfaces and contour lines from 3-D grids.

VIS-5D runs under Stellix (UNIX System V with Berkeley extensions) and X Windows Version 11 Release 3. The top level pseudo-code for the visualization program is:

```
read the data set into the compressed 5-D byte array
create the linear array for polygon and vector lists
initialize the polygon and vector lists to empty
initialize the contour surface and line queue to empty
if the user specified a trajectory data set
  read the trajectory file
  build the trajectory vector lists and time-step indices
end if
if the user specified a map
  read the topography and map outline files
  resample these map data to a reasonable resolution
  build the map polygon mesh and vector lists
end if
set defaults for colors of contour surfaces and lines
set defaults for iso-levels of contour surfaces
set defaults for altitudes and densities of contour lines
create a window for the 3-D display
create widgets according to data set contents
initialize the display to the first time-step with no graphics
          enabled and nominal 3-D to 2-D projection
fork into 4 parallel threads
  thread 1
    do forever
      clear the display
      render a rectangular box
      render the map if enabled
      render the trajectories if enabled
      for each variable
        if the contour lines are enabled
          decompress the line vector list for the current time-step
          render vector list according to color for the variable
        end if
      end for
      for each variable (in order of decreasing opacity)
        if the contour surface is enabled
          decompress the surface polygon list for the current time-step
          render polygon list according to color and transparency
                  for the variable
        end if
      end for
      check for X events and widget callbacks
        adjust the projection matrix according to mouse moves
        toggle map enable/disable if requested
        toggle trajectory enable/disable if requested
        toggle contour surface and line enable/disables if requested and re-order lists of
                variables recording which have been most recently displayed
        toggle time animation enable/disable if requested
        if time animation is disabled
          increment, decrement or reset time-step if requested
        end if
        change colors and transparencies if requested
        change contour surface levels if requested
        change contour line altitudes and densities if requested
```

```
    if a contour surface or line recompute is requested
      for each time-step
        if time animation is disabled and time-step=current
          add the selected variable and time-step to the head of the queue
        else
          add the selected variable and time-step to the tail of the queue
        end if
      end for
    end if
    exit visualization program if selected
  end check for X events and widget callbacks
  if time animation is enabled
    increment the time-step
  end if
 end do forever
end thread
threads 2, 3 and 4 (they are identical)
 do forever
  if the queue contains any contour line requests
    remove the first request for contour lines
    decompress the 3-D grid for time-step and variable
    compute contour lines at altitude and density for variable
    compress vector list for lines
    deallocate previous vector list for time-step and variable
    if there is not adequate free space in the linear array
      delete the least recently used vector and polygon lists
              until there is adequate space
    end if
    allocate space in linear array and insert vector list
    add index to vector list for time-step and variable
  else if the queue contains any contour surface requests
    remove the first surface request
    decompress the 3-D grid for time-step and variable
    compute contour surface at iso-level value for variable
    compress polygon list for surface
    deallocate previous polygon list for time-step and variable
    if there is not adequate free space in the linear array
      delete the least recently used vector and polygon lists
              until there is adequate space
    end if
    allocate space in linear array and insert polygon list
    add index to surface list for time-step and variable
  end if
 end do forever
end thread
end fork
```

The ST-1000 and ST-2000 execute four instruction streams in parallel, so VIS-5D forks into four threads to take advantage of this parallelism. The X server is also a heavy computing load while VIS-5D is running and increases parallelism. Because fast response is important to VIS-5D, data should be accessed from main memory rather than disk. VIS-5D allocates a single large array from which to allocate polygon and vector lists in order to control the total use of main memory. This way it can avoid paging delays which would occur if allocated memory became significantly larger than physical memory.

The parallel threads implement critical sections where simultaneous access to common data structures could cause interference. This true for insertion and deletion in the queues, the allocation and deallocation of space in the linear array, and reading and updating the polygon and vector lists and their associated index.

VIS-5D uses Stardent's XFDI library of 3-D extensions to X for rendering, using a Z-buffer and RGB true color. We also use a modified version of Stardent's LUI widget library, which is part of their Application Visualization System (AVS).

Future Developments

We have received numerous suggestions for additional functions for VIS-5D from scientists, as well as shortcomings which we recognize. Some of these are:

A. include contour lines drawn on vertical planes which can be arbitrarily positioned. This is currently being developed.

B. dynamically calculate trajectories through space-time points specified with a 3-D cursor. This is currently being developed.

C. represent planes through 3-D grids with pseudocolored images in addition to the current contour lines. This would be useful for radar data which are less smooth than model data.

D. texture map satellite images onto surfaces in the 3-D box.

E. render 3-D grids as transparent fogs, often referred to as volume images. This may be difficult to do with fast enough response for interactive rotation.

F. provide interactive analysis operations on the 5-D grid of data, including arithmetic, differential and integral operations. This is an open-ended area of development, often dependent on the particular source of the data set.

G. increase the size of the data sets which can be interactively visualized. This applies to the total 5-D rectangle and the number of grid points in the spatial 3-D box. Assuming the current level of interactivity, this depends on faster workstations, larger memories, and disks fast enough to support interactive access.

VIS-5D is aimed at 5-D data sets similar to those produced by weather models. We are also interested in developing systems for interactively visualizing and analyzing large image data sets. The same workstation technology which makes VIS-5D possible can also be exploited for radical new ways of processing image data, although the overall structure of such an application may be quite different from VIS-5D.

Acknowledgments

We wish to thank Dave Kamins and Jeff Vroom of Stardent Computer, Inc., and Marie-Francoise Voidrot-Martinez of the French Meteorology Office for their help. We also wish to thank the many scientists we have worked with and NASA Marshall Space Flight Center for their support (NAS8-36292).

References

1. Hibbard, W., and D. Santek, 1989: Visualizing large data sets in the earth sciences. IEEE Computer 22(8), 53-57.

2. Tripoli, G., W. Hibbard, and D. Santek, 1989: Four-dimensional interactive analysis: a tool for the efficient understanding of large data sets. Preprints, 12th Conference on weather analysis and forecasting. Monterey, American Meteorological Society, J10-J12.

Appendix C

```
IGU3D -- 3-D Grid file utility

      IGU3D LIST bgridf egridf <keywords>
      IGU3D PRINT bgridf egridf <keywords>
      IGU3D SET (gridf)
      IGU3D COPY sgridf dgridf (RENUMBER)
      IGU3D MAKE gridf maxsiz ("comment)
      IGU3D DEL gridf-1 gridf-2
      IGU3D DIR gridf date project "comment
 Parameters:
      gridf | grid file number
      bgridf | beginning grid file number
      egridf | ending grid file number
      sgridf | source grid file number
      dgridf | destination grid file number
      date  | creation date of file
      project | project number
      maxsiz | maximum size for a 3-d grid (nr*nc*nl)
      "comment | comment appended to grid file directory
 Keywords:
      PROJ=  list grid files with project number(s)
      DAY=   list grid files with this date, YYDDD
```

```
IGG3D -- 3-D Grid utility

      IGG3D LISt gridbeg gridend <keyword>
      IGG3D INFo gridbeg gridend <keyword>
      IGG3D DELete gridbeg gridend <keyword>
      IGG3D GET sgridfile gridbeg gridend dgridfile gridbeg
      IGG3D MAKe grid1 oper <grid2> <keyword>
      IGG3D COPy grid1 grid2
      IGG3D INTerp grid day time SETDEL=gstep ngrids LAG=uspd vspd
      IGG3D RH Q-grid T-grid P-grid     (RH calculation)
      IGG3D THE T-grid P-grid <offset>  (Theta calculation)
      IGG3D LAG grid rday rtime uspd vspd  (Lagrangian coords)
      IGG3D ADD grid1 grid2 ... grid6 <NAME=> (combine up to 6 grids)
 Parameters:
      gridbeg | beginning grid number
      gridend | ending grid number
      sgridfile | source grid file number
      dgridfile | destination grid file number
      oper | option applied to grid, SPD, SUB, SUBL, EXT
         SPD - wind speed from u & v
         SUB - subtract grids
         SUBL- subtract logarithmically
         EXT level# - extract a level and file as 2-D grid
         FIL value lev1 lev2  - fill selected levels of 3-D
          with value. (if value=MISS, for missing value)
```

```
 Keywords:
      GR3DF=  3-D gridfile (default=from IGU3D SET)
      GRIDF=  gridfile (default=from IGU SET)
      V= input v-grid if it does not follow u-grid (def=ugrid+1)
      NAME=  parameter name to assign to result of ADD option
```

TOPO3D -- Draws 3-D topographical maps

```
      TOPO3D area lr (keywords)
 Parameters:
      area | number of area in which to draw topography
             also creates depth map in area+1
      lr | 1 for left eye view (default), -1 for right eye view
 Keywords:
      SPEC=OCEAn (flat ocean )
       =LAND (flat land)
       =ZERO (black  area)
          =BLACk (no bottom)
      LAT=slat nlat    latitude extents (default=20.0 50.0)
      LON=elon wlon    longitude extents (default=60.0 125.0)
      HGT=bhgt thgt    height extents in km (default=0.0 12.0)
      MAP= name of map file (NONE for none, default=OUTLUSAM)
      MESH=            number of mesh points (default=5000)
      VROT=            vertical rotation angle (default=0.0)
      HROT=            horizontal rotation angle (default=0.0)
      DEPTH=           view depth (default=1.0)
      DIM= dim         dimming factor (default=0.8)
      TRAJ=            TRAJECTORY TO DETERMINE LIMITS OF BOX
      GR3D=            3-D GRID TO DETERMINE LIMITS OF BOX
      GR3DF=           3-D GRIDFILE FOR GRID= (DEF= IGU3D SET)
      GRID=            2-D grid of topography values
```

DRAW3D -- 3-D Rendering program

```
      DRAW3D area lr marea (Keywords)
 Parameters:
      area | number of area in which to render
      lr | 1 for left eye view (default), -1 for right eye view
      marea | area of topography map (also uses marea+1)
 Keywords:
      GRID=            3-d grid number (def=1)
      LEV =            contour level (no default)
      TRANS=           opacity (0-1 def=.5)
      MONO= nearfact farfact taper. For depth cue (def, use =0)
      DIAM= diameter of trajectories (def=7.0)
      GR3DF= 3-D_file  Grid file for 3-D grid (def=IGU3D SET)
      DENSE= 3-d_grid_# (<0 to invert) edge_flag t (def=1 0 0.8)
      GSFC=            grid int lev hgt (def= 0 4 220 0)
      WIRE=            grid value lev skip (def=0 0 245 1)
      TEXT=            file lev size (def=X 225 10)
      TRAJ= traj# yyddd hour   (def for day & time from 3-d grid)
      LEN= length scaling factor for trajectories (def=1.0)
      COLOR= carea blue gsfc wire text   for 12-bit color
```

```
Remarks: The  - hgt - parameter in GSFC should be set to a
         positive number to use a height grid and should be set
         to negative number to use a constant height & 0 for
         surface
```

TRAJ4D -- Analyze 3-D grids for trajectories

```
 TRAJ4D itraj igrid nsets KEYWORD=...
 Parameters:
      itraj | index of trajectory analysis to make
      igrid | index of 3-D grid with first data set
      nsets | number of data sets to analyze
 Keywords:
      SETDEL= set_spacing  sets_per_file (def=3 100)
      GR3DF= 3-D Grid file number (def=IGU3D SET)
      DSP= spacing center_lat center_lon (def=4.0 -100.0 -100.0)
      NOTOPO= non-zero to turn below topo check off (def=0)
      STEP= # of steps across box (def=100.0)
```

LISTTJ -- Lists 4-D grid trajectory analyses

```
    LISTTJ low high
    LISTTJ DUMP index lowtraj hitraj npoints
    LISTTJ COMB index1 index2 outdex
    LISTTJ STAT index newval lowtraj hitraj
    LISTTJ BASE index zbase
    LISTTJ CEIL index zceil
    LISTTJ COPY index outdex
    LISTTJ APND index traj x y z tdel
```

RGB3D -- Load color table for 3-D images

```
 RGB3D BLUE GSFC WIRE TEXT        (DEF=20 55 61 56)
 RGB3D -1 iarea
 Keywords:
      SIX=  1 (low frame) or -1 (high frame)
```

GG3D -- 3-D Grid resample

```
      GG3D AVE grid
      GG3D SAM grid
      GG3D MAX grid
 Parameters:
      grid | source grid number
 Keywords:
      LAT= lats latn     Latitude extents (def from source grid)
      LON= lone lonw     Longitude extents (def from source grid)
      SIZE= nr nc nl     Grid size for Lat, Lon & Vert
      HGT= HGTB HGTT     Height extents (def from source grid)
      GR3DF=  3-D gridfile (default=from IGU3D SET)
      GLEV= grid level for adjustment
```

```
RG3D -  Generate 3D grid from gridded data (DAS)

    RG3D yyddd hour parm  <Keywords>

      yyddd hour - Day/time of data (def= 0Z today)
      parm - parameter to regrid

   Keywords:
      LAT= lats latn    Latitude extents (def=20.0 50.0)
      LON= lone lonw    Longitude extents (def=60.0 125.0)
      SIZE= nr nc nl    Grid size (def=30 30 10)
      HGT= hgtb hgtt    Height extents (def=0.0 15.0 km)
      LEV= levt levb    Pressure level extents (def=100.0 1000.0)
      GR3DF= 3-D gridfile  Grid file (def=IGU3D SET)
      GRIDF= gridfile   Grid file (def=IGU SET)
      GRID= num         Number to file 3-D grid (def=next)
      RANGE= beg end    Range of grids to search (def=1 159)
      VT= valid-time    Valid time for grid (def= not used)
```

```
 GRID3D - Produces a 3-D grid from RAOB data

  GRID3D yyddd hour parm1 <parm2...>  <Keywords>

   yyddd hour - Day/time of RAOB data to grid (def= 0Z today)
   parm1 <parm2..> - P, T, TD, SPD, THA, THAE, MIX, Z, U, V, W def=P
  Keywords:
      LAT= lats latn    Latitude extents (def=20.0 50.0)
      LON= lone lonw    Longitude extents (def=60.0 125.0)
      SIZE= nr nc nl    Grid size (def=20 20 10)
      HGT= HGTB HGTT    Height extents (def=0.0 12.0)
      LEV= levt levb    Pressure level extents (def=100.0 1000.0)
      MD= mdmand mdsig  MD file numbers (def=current)
      GRID= num         Number to file 3-D grid
      GR3DF= 3-D file   Grid file for 3-D grid (def=IGU3D SET)
      CUT= SD cutoff    SD to throw out data (def=4)
```

```
 CLD3D -- Make 3-D cloud images

      CLD3D irarea varea darea marea lr (keywords)
 Parameters:
      irarea | IR source area number
      varea | visible source area
      darea | destination area (also darea+1 for debug)
      marea | area of topography map (also uses marea+1)
      lr | choose eye for view, +1 for left, -1 for right (def=1)
 Keywords:
      LAT= lats latn    latitude extents (def=30.0 40.0)
      LON= lone lonw    longitude extents (def=80.0 90.0)
      HGT= hgtb hgtt    height extents in meters (def=0.0 24.0)
      VROT= vertical rotation angle (def=0.0)
      HROT= horizontal rotation angle (def=0.0)
      DEPTH= view depth (def=1.0)
      MDF= MD file for surface temp (SVCA) (def=current)
```

```
Keywords for pattern recognition and rendering:
     CUT= rat ivcut zcut (def=10.0 100 3.0) ratio, vis pix, height
     SHADE= vamb (def=0.25) visible fraction
     STAT=wvar wtot ismth iwide (d=3.0 1.5 15 15) statistical params


Draws 3-D Lidar images

     LID3D sarea darea lr trans (keywords)
Parameters:
     sarea | number of first area containing lidar image
     darea | number of area in which to draw 3-d radar image
     lr | 1 for left eye view, -1 for right eye view (def=1)
     trans | 1-opaque, 2-constant trans, 3-variable trans def=1)
Keywords:
     VROT=            vertical rotation angle (default=0.0)
     HROT=            horizontal rotation angle (default=0.0)
     DEPTH=           view depth (default=1.0)
     CUT=             invisible pixel cutoff (default=20)
```