Visualizing Gridded Datasets with Large Number of Missing Values

Suzana Djurcilov and Alex Pang
Computer Science Department, UCSC
suzana@cse.ucsc.edu, pang@cse.ucsc.edu

Abstract

Much of the research in scientific visualization has focused on complete sets of gridded data. This paper presents our experience dealing with gridded data sets with large number of missing or invalid data, and some of our experiments in addressing the shortcomings of standard off-the-shelf visualization algorithms. In particular, we discuss the options in modifying known algorithms to adjust to the specifics of sparse datasets, and provide a new technique to smooth out the side-effects of the operations. We apply our findings to data acquired from NEXRAD (NEXt generation RADars) weather radars, which usually have no more than 3 to 4 percent of all possible cell points filled.

Key Words and Phrases: curvilinear grid, sparse data visualization, Delauney triangulation, isosurface, NEXRAD.

1 INTRODUCTION

In atmospheric sciences, researches are often facing the problem of visualizing gridded datasets that are incomplete: either a grid of data is extrapolated from a set of scattered data observations and there is insufficient support to validate a data point, or there is simply no observation for that specific region. Observed meteorological data are mostly point (0 dimension) measurements made at sparsely distributed weather stations. They may also be 1 dimensional measurements such as those obtained a weather balloon or a vertical wind profiler which measures wind velocities at various heights. They may also be 2 dimensional such as those obtained from satellite imagery or derived from radial velocities from a pair of radars (e.g. CODAR). More recently, volumetric weather measurements are available through the NEXRAD radar which makes a series of conical scans. This paper is a case study in visualizing 3D volumetric NEXRAD data which is characterized by a large proportion of missing or invalid data points.

2 BACKGROUND

NEXRAD, also known under its official name WSR-88D is a meteorological instrument generating dozens of data types, including reflectivity, storm total precipitation and wind radial velocity. It does so by processing the radar signal bounced off the particles in the atmosphere, be it rain drops, hail, or ice crystals. Occasionally, NEXRAD may report non-weather related phenomena, such as smoke plumes from fires and large flocks of birds. It is a doppler radar and uses the "doppler effect" to determine the direction and velocity of the particles along a radial direction. On a clear day, NEXRAD can retrieve information as far out as 143 miles. As opposed to the conventional weather radar which makes one circular sweep, NEXRAD makes up to 14 conical sweeps with varying elevation angles from the horizon, often ranging from 0.5 to 19.5 degrees above the horizon. The sweep pattern, including grid points where available, is illustrated in Figure 1. The most interesting feature of NEXRAD is that it measures the weather within its local volume of space, producing a curvilinear volume scan from the conical sweeps. NEXRAD is most noted for its usefulness in early storm detection and rainfall estimates.

Currently, the National Weather Service has 160 WSR-88D radars employed nationwide. For this case study, we use data from the San Francisco Bay Area NEXRAD, stationed atop Mt. Umunhum.

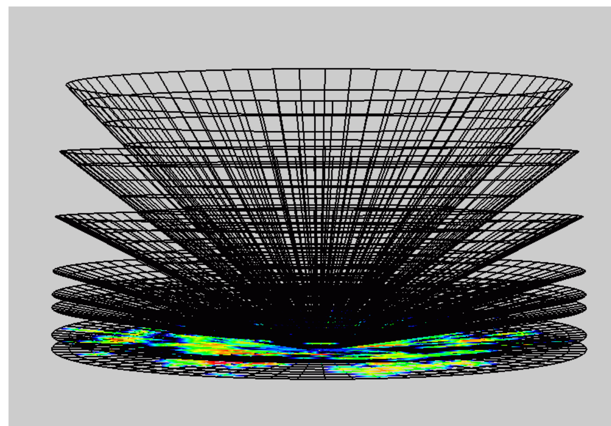


Figure 1: NEXRAD grid pattern

Our primary interest was developing 3D images, since the current visualization techniques for NEXRAD simply discard the 3D nature of the dataset and provide 2D plots of the lowest layer of the grid. The main difficulty arose from the fact that most visualization techniques that are designed for gridded data assume that the underlying dataset is complete. That is, it contains a valid data value at each grid point. When incomplete, gridded datasets are often filled with pre-defined values outside the valid data range [5, 2, 1] for example by NaN, the IEEE representation for Not-a-Number. How incomplete, invalid, or missing values are treated varies from implementation to implementation, and from application to application. Often times, the visualization program will leave holes in the visualization where the data is missing. Other times, they may explicitly be flagged and shown together with the rest of the data. Depending on the type of data, this may or may not be desirable. According to [9]: "Empty spaces [must] signify absence of phenomena and not missing data." Because of the nature of radar returns from NEXRAD, this is often the case – there is not sufficient phenomena (e.g. precipitation, air particles) at places where there are no measurements available. As such, NEXRAD data requires special handling particularly when it comes to interpolating neighboring grid points with missing data as is often the case during a resampling operation. The treatment is similar to that of interpolating valid measurements from a sparse number of point measurements from scattered weather stations. For example, Pang et al. [7] explored the inclusion of data quality or uncertainty parameters to the different measurements when attempting to resample to a full grid using a variety of global interpolation and approximation methods.

3 VISUALIZATION ISSUES

One of the primary use of NEXRAD data is identifying storm clouds and tracking them over time. Figure 2 shows a sample NEXRAD data set for a particular instance in time. Each colored point represent a location in the vicinity of the radar with a valid reading. The value at each location is pseudo-colored such that low values are bluish, and high values are reddish. It is pretty apparent that there is a significant amount of dark regions with no data. Most standard visualization packages either fails or give incorrect visualizations of this dataset.

In this section, we explore several ways of extracting and presenting relevant NEXRAD information. In this exercise, we assume that the value of 23.5 represents some physically significant phenomenon such as water particles in the clouds. Hence, the feature of interest is to locate these rain cell clusters.

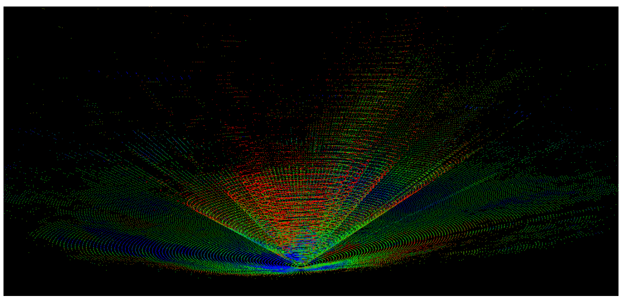


Figure 2: Pseudo-colored point cloud of NEXRAD data

3.1 Point Cloud

A natural start with a scattered dataset is to simply draw a colored point for each data point, also known as a point cloud. The image in Figure 2 is exaggerated in the vertical direction and quickly points out the conical nature of the grid. It is also easy to see that most of the data are found in the lower layers of the atmosphere, with density quickly receding in the higher levels. To find the regions where there is rain (i.e. values above 23.5), we generate another point cloud representation for those values. This is shown in Figure 3 below, along with the underlying terrain: the greater San Francisco Bay Area. The white points are at 23.5 moving towards red as the values increase. We can observe that the largest cloud cluster is formed above the entrance to the bay, in the upper-left part of the image.

3.2 Delaunay Triangulation

While the point cloud is a cheap and simple way of identifying important values in the dataset, the presentation is less than ideal and tend to get confused because of poor depth cues, specially in a static view. Hence, the next step in the process might be to try and extract a surface from these points of interest. The problem reduces to one of creating a set of triangles connecting points that are natural neighbors. This task is accomplished by using Delaunay triangulation on the desired subset of points – with values ranging from 23.4 to 23.6. We also modify the algorithm slightly to eliminate large triangles that may crop up that connect different clusters together.

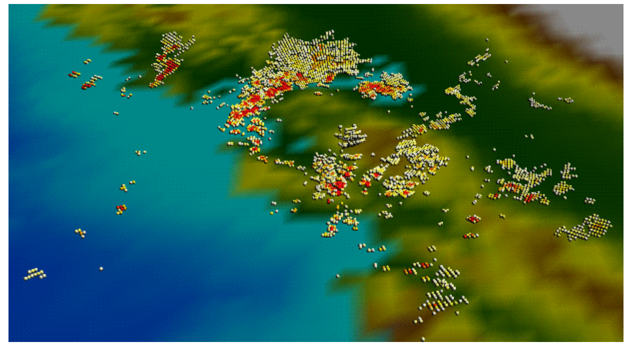


Figure 3: Pseudo-colored point cloud of NEXRAD data representing "rain" (value above 23.5)

The result of this operation can be seen in Figure 4. Here, we start to see more coherent structures of what might be cloud formations. This Delaunay triangulation and thresholding process may be further improved by adding a physically meaningful set of constraints that specify boundary conditions and surface tension coefficients.

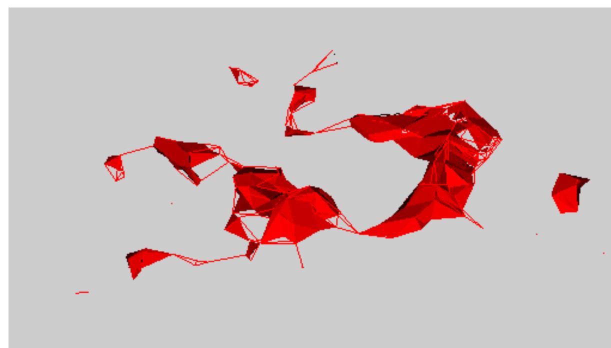


Figure 4: Delaunay triangulation of data points between 23.4 and 23.6

3.3 Interpolation

Interpolation is a popular option employed to obtain unknown or missing values from known ones. A wide variety of algorithms exist, for example, Shepard's [10] inverse distance weighting functions, Hardy's radial basis multi-quadrics [4], and thin-plate splines [3] to mention a few. One must be careful when selecting the interpolation method, as most of them do not provide any means of estimating the interpolation error. Those that do, such as kriging [6], are computationally expensive and do not scale well with number of data points. Many researchers, e.g. [13, 7], have had to grapple with the issues arising from inadequacy and uncertainty in implementing interpolation methods, caused by the lack of constraints on the output data. An example of such side-effects can be seen in Figure 5. Here, the same dataset as seen previously in Figure 2 was resampled to a 40x40x20 grid using Shepard's method. We then generated isosurfaces using a threshold of 23.5. The resulting isosurfaces have a layered look as if the volume of the clouds increase at different layers. This problems is not a feature of the data but rather an artifact from the interpolation algorithm. Using Hardy's multi-quadrics also has a major disadvantage because it can produce interpolated values outside the original data range.

Finding a reliable interpolation scheme, especially for unevenly distributed data is still a great challenge. Furthermore, for many

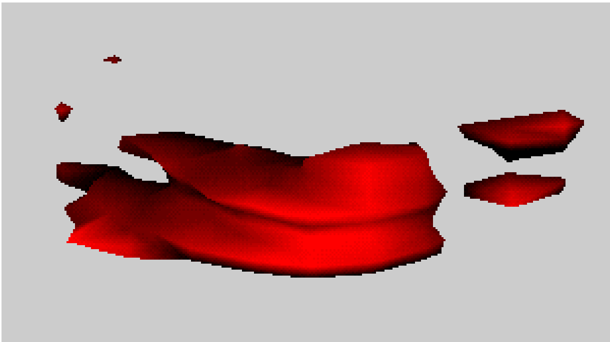


Figure 5: Undesirable artifacts when using Shepard's interpolation

environmental data it means going from a small set of trustworthy information to a large set of untrustworthy information. In the case of NEXRAD, where missing data signifies a lack of atmospheric phenomena, it would be meaningless to attempt and produce a value at all cost.

3.4 Isosurface

So then, why not simply replace the missing data with a value that is outside the valid range (e.g. NaN) and proceed with standard algorithms? Contour lines in 2D or isosurfaces in 3D can be erroneously formed where none are meant to exist. In general, false contours will show around the clusters of valid points, thus giving the dataset an artificial "layer" which influences and will be seen on most thresholding levels.

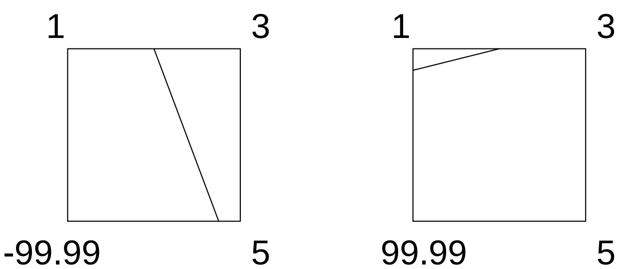


Figure 6: Incorrect contours when arbitrary numbers are used

Figure 6 shows the problems encountered when users choose to arbitrarily fill the dataset with data that is outside the valid range. In this illustration, we have chosen 99.99 and -99.99 to represent "invalid" numbers, and gave an example of iso-lines at threshold value 2. As can be seen from this example, due to linear interpolation along the cell edges, incorrect contour edges are produced in cells containing invalid points. Volume visualizations, such as isosurfacing and direct volume rendering will suffer as well, unless the algorithms are transformed to explicitly ignore invalid data points during interpolation procedures. In addition, algorithms using gradient information to calculate the surface normals must be adjusted to use central, forward or backward differences, depending on what configuration of data points are available for the cell. Perhaps, a better approach is to ignore cells with missing data points. This approach is presented below.

3.5 Modified Normals

Fortunately, nowadays many visualization programs are able to ignore the "missing" grid points and produce an image for only the part of the grid that yield complete cells. An example produced by VTK [8] can be seen in Figure 7. Unfortunately, when there are large gaps, and the data is of high frequency, such as in the case with NEXRAD, the isosurfaces appear grainy and spiky.

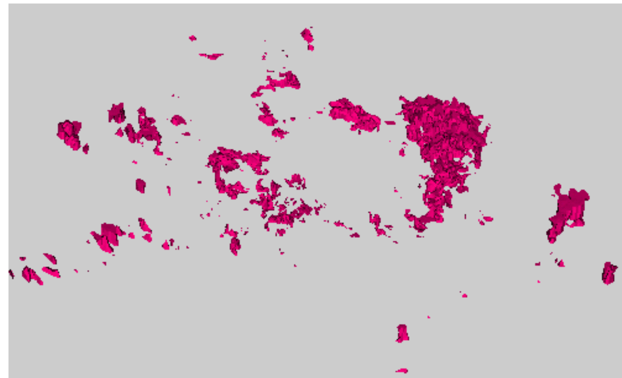
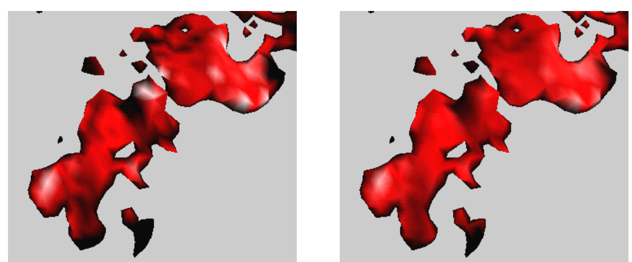


Figure 7: Isosurface at 23.5 threshold

We have developed a method to smooth out some of the roughness seen in an isosurface created from sparse gridded data. The first step is implemented using a first-order neighborhood smoothing algorithm similar to the one seen in [11]. We use the gradient information of the neighbor points to adjust the surface normals. Out of the 6 neighbor points, each valid one adds 10 percent of its value to the neighbor. The idea is to exploit the nature of atmospheric data and propagate the neighboring information among the cells. In essence we are compensating for a lack of information by passing information around. Another way of thinking about it is that we are using a low-pass filter to smooth the signal. Figure 8 shows the visual effects of smoothing the surface normals.



Without smoothing   With smoothing

Figure 8: Effects of modifying surface normals

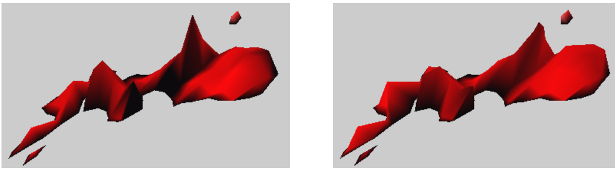
3.6 Modified Isosurface

In addition to modifying the surface normals, we take the previous method a step further and smooth both the point location and the surface normals of the isosurface triangles.

This method works as follows:

- Calculate the gradient at each valid grid point, using either central, forward or backward differences
- Update the gradient at each valid grid point by adding influences from the valid neighbors
- Reposition each valid grid point in the direction of the gradient vector
- Create isosurface using standard techniques

Images of an isosurface detail before and after the modification are seen in Figure 9 below. Our results show that although remarkable effects are not likely, this method does smooth out the sharp edges to bring out the cloud-like nature of the dataset.



Original geometry   Modified geometry

Figure 9: Effects of modifying geometry and surface normals

3.7 Smoothed Isosurface

Finally, the most successful visualization was produced when we used a Taubin's [12] smoothing algorithm that functions on the isosurface geometry by balancing out the bumps and valleys. This is a non-shrinking variant of the Gaussian smoothing algorithm that moves vertices of the isosurface geometry inwards and outwards, in alternating steps. The image in Figure 10 was created after 20 iterations of the algorithm.

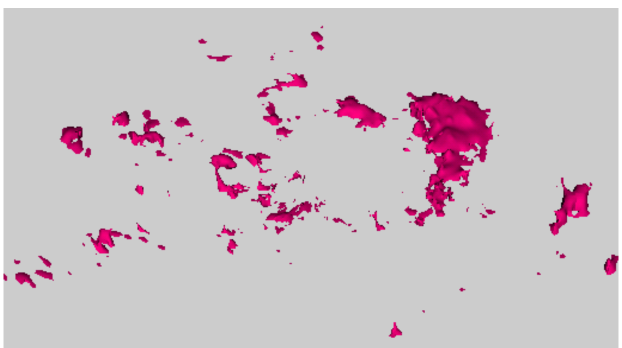


Figure 10: Geometrically smoothed isosurface

4 CONCLUSIONS

Unevenly sampled data pose a common problem for scientists needing efficient and appropriate visualizations. We have presented some of the challenges inherent in handling sparse gridded data and shown some common methods in visualizing such datasets. Furthermore, we have presented a measure for improving the isosurfaces produced from gridded datasets with a large number of missing values. In the future, we would like expand the domain of our research into other physical sciences and experiment with a variety of gradient filters.

5 ACKNOWLEDGEMENTS

We would like to thank the Santa Cruz Laboratory for Visualization and Graphics (SLVG) for the wonderful research environment. This project is supported by ONR grant N00014-96-1-0949, NASA grant ncc2-5281, LLNL Agreement No. B347879 under DOE Contract No. W-7405-ENG-48, DARPA grant N66001-97-8900, and NSF grant IRI-9423881.

References

[1] Advanced Visual Systems. www.avs.com.

[2] IBM Visualization Data Explorer. www.almaden.ibm.com/dx.

[3] F.L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 567–585, June 1989.

[4] R. Hardy. Multiquadratic equations of topography and other irregular surfaces. *Journal of Geophysical Research*, 76:1905–1915, 1971.

[5] William Hibbard and David Santek. The Vis-5D system for easy interactive visualization. In *Visualization '90*, pages 28–35, 1990.

[6] G. Matheron. Principles of geostatistics. *Economic Geology*, 58:1246–1266, 1963.

[7] Alex Pang, Jeff Furman, and Wendell Nuss. Data quality issues in visualization. In *SPIE Vol. 2178 Visual Data Exploration and Analysis*, pages 12–23. SPIE, February 1994.

[8] W.J. Schroeder, K.M. Martin, and W.E. Lorensen. The design and implementation of an object-oriented toolkit for 3d graphics and visualization. In *Proceedings of Visualization 96*, pages 93–100, 472. ACM, 1996.

[9] Hikmet Senay and Eve Ignatius. Rules and principles in scientific data visualization. Technical report, Department of Computer Science, George Washington University, Washington, D.C, 1990.

[10] D. Shepard. A two-dimensional interpolation function for irregularly spaced data. In *Proceedings of the 23rd National Conference*, pages 517–524. ACM, August 1968.

[11] C. Siege, P. F. Hemler, and D. J. Vining. Surface smoothing for enhanced 3D medical visualization. In *Proceedings of the SPIE*, pages 582–592. IEEE, February 1997.

[12] G. Taubin. A signal processing approach to fair surface design. In *Computer Graphics Proceedings*, pages 351–358, 1995.

[13] Y. Xiao, J. Ziebarth, C. Woodbury, E.Bayer, B.Rundell, and J. van der Zijp. The challenges of visualizing and modeling environmental data. In *Proceedings of Visualization 96*, pages 413–416, 1996.