Visualization databases for analysis of large and complex data

Ryan P. Hafen, Saptarshi Guha, William S. Cleveland
Purdue University Regional Visualization and Analytics Center

What is a visualization database?

A visualization database (VDB) is a repository of a vast number of images that can be queried and viewed on an as-needed basis. This is nothing new - in fact, many people use one example of a visualization database almost every day:

Example: Google Earth is a visualization database

The Google Earth application uses a 70 terabyte database to serve processed images of the globe to the user. Each row of the database corresponds to a single geographic segment which corresponds to a few raw images. The user, interested in specific geographic regions, queries Google Earth to retrieve the corresponding images. Google Earth is a viewer into a virtual display consisting of thousands of images.

This is an example of a large visualization database consisting of many images. Here, the context is simple: retrieving world satellite imagery.

If Google can do it for spatial data, why can't we do it generally?

Our context for visualization databases: Large complex data sets

We apply the visualization database concept to the analysis of *large and complex data*, which is the standard for homeland security data. Large and complex data have the following properties:

- a very large number of observations (records)
- many variables
- complex data structures not readily put into a tabular form
- complex patterns and dependencies in the data that require complex methods of analysis and models

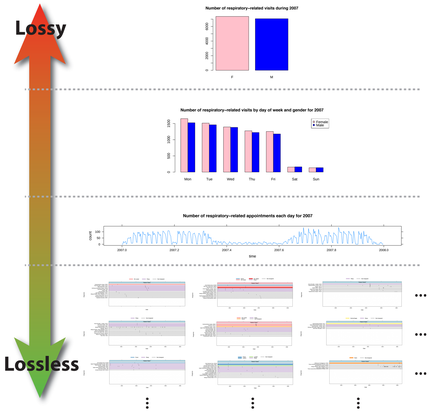
To comprehensively analyze the data, the VDB creates thousands of images. Unlike Google Earth, the images are *not given* but need to be intelligently generated. This makes the problem more challenging.

The challenge: lossless analysis

Nothing serves analysis better than visualization. For large, complex data sets, we need detailed, comprehensive analysis that does not lose important information contained in the data. Summary analyses will not be enough to help the analyst find the needle in the haystack. With large and complex data, the goal should be lossless analysis: analyze the data at a level of detail that does not sacrifice information.

***Outbreak of infectious disease:* an example**

In an ongoing disease surveillance project, we are tracking patient visits to the Purdue University Student Health clinic. Variables include time of visit, diagnosis, and gender. Below is a continuum of examples of visualizations of this data ranging from lossy to lossless.



At the lossy end of the spectrum, the data is aggregated into counts of respiratory illnesses broken down by gender. All other information is lost. Each plot successively loses less information.

At the lossless end, we see detailed plots of visit times and diagnoses for each patient. All of the information in the data is visually coded for the variables mentioned, enabling patient-level visual analysis. There are over 60,000 patients in the database and hence over 60,000 plots.

It is clear that to achieve lossless visualization, we need to be able to quickly create, manage, and view many images. We need a visualization database system.

How a visualization database works

There are two main components of a visualization database system:

- **image generation:** includes graphical methods, computational algorithms, statistics, and tools for visual perception.
- **management:** store, query, and view images in the database.

Creating a visualization database

For lossless analysis, the data is broken down into subsets. Plots of different objectives are made for each subset. Once a plot is made for one subset, it can easily be replicated across other subsets. For example, in the disease data example to the left, each subset was a patient.

Critical factors in VDB display design involve:

- maintain visual identification and assembly of graphical elements that must be seen as a **single gestalt**
- visually decoded gestalts must convey quantitative information with requisite **accuracy**
- **maximize density** subject to constraints of visual pattern perception
- cannot fuss with very large displays, interacting with the micro -elements to get them right - need algorithms that **get the display right the first time**
- display the same information many times with **different designs**

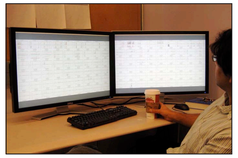
Viewing the images in a VDB

The atomic element of a VDB is a single plot. Studying the VDB involves viewing different collections of the atomic elements.

How am I going to look at all those plots?

There are different modes of viewing a collection of plots. In some cases many aspects of the visual patterns are studied intensively for each plot. In other cases, just a single aspect is studied, and the analyst can race at very high speed through the plots.

More screen space is better than less. We have found that using two 30" monitors works very well. This allows many plots to be seen at once, all within the natural human visual field.



The viewer component of the VDB in action.

Why use VDBs for analysis?

No Surrender
VDBs help achieve lossless analysis, not surrendering to the size and complexity of the data.

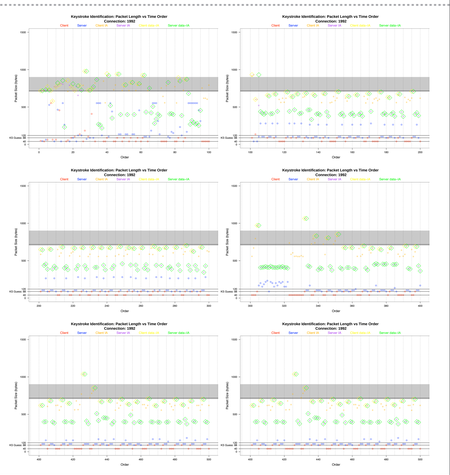
Cheap and easy
The hard part is building the visualization tool in the first place. Once it is built, it is easy to apply to many subsets of the data. And disk is cheap.

Think of it as a new database
We query the database on an as-needed basis. Some displays are studied very intensively, others less so, and some hardly at all.

Field testing has shown it works
Visualization databases have played a vital role in a number of situations involving large and complex data. Examples are:

- Internet packet header data for network security, historical and streaming
- disease surveillance for homeland security based on an integrated database for humans, companion animals, livestock, and wildlife
- indoor RFID tracking of nurses from 36 hospital units across the U.S.

At the right are a few plots from two different displays used in the visualization database for the network security project.



Acknowledgements

This research is funded through the National Science Foundation and through Regional Visualization and Analytics Center of Excellence by a grant from the Department of Homeland Security, Science and Technology Directorate, Office of University Programs. We would also like to thank our collaborators: The Indiana State Department of Health (ISDH), the Indiana Network for Patient Care (INPC), the Purdue University Student Health Clinic (PUSH), and the Stanford RVAC.

For further information

Please contact Ryan Hafen: rhafen@purdue.edu
More information on this and related projects can be obtained at http://www.purvac.org.

PURVAC
Purdue University Regional Visualization and Analytics Center

RVAC
Regional Visualization and Analytics Centers