2015/7/21

林涛

Related Work 参考文献的摘要和简化

1. large graph

An Experimental Comparison of Fast Algorithms for Drawing General Large Graphs

In the last decade several algorithms that generate straight-line drawings of general large graphs have been invented. In this paper we investigate some of these methods that are based on force-directed or algebraic approaches in terms of running time and drawing quality on a big variety of artificial and real-world graphs. Our experiments indicate that there exist significant differences in drawing qualities and running times depending on the classes of tested graphs and algorithms.

Hachul and Jünger's compared different large graph drawing algorithms by the differences in drawing qualities and running times.

/*

A survey of two-dimensional graph layout techniques for information Visualisation

(same topic, but listed more algorithms)

*/

A Fast Multi-Dimensional Algorithm for Drawing Large Graphs

GRIP is a hierarchical force-directed method for drawing large graphs.

OpenOrd: An Open-Source Toolbox for Large Graph Layout

OpenOrd has a parallel implementation of large graph layout.

A Scalable Parallel Force-Directed Graph Layout Algorithm

Tikhonova and Ma presented a scalable parallel graph layout algorithm based on the force-directed model.

---

2. Info Parallel

imMens: Real-time Visual Querying of Big Data

Data analysts must make sense of increasingly large data sets, sometimes with billions or more records. We present methods for interactive visualization of big data, following the principle that perceptual and interactive scalability should be limited by the chosen resolution of the visualized data, not the number of records. We first describe a design space of scalable visual summaries that use data reduction methods (such as binned aggregation or sampling) to visualize a variety of data types. We then contribute methods for interactive querying (e.g., brushing & linking) among binned plots through a combination of multivariate data tiles and parallel query processing. We implement our techniques in imMens, a browser-based visual analysis system that uses WebGL for data processing and rendering on the GPU. In benchmarks imMens sustains 50 frames-per-second brushing & linking among dozens of visualizations, with invariant performance on data sizes ranging from thousands to billions of records.

Liu et al. put up a combination of multivariate data tiles and parallel query processing for real-time interacive querying large data sets.

Visual analytics of large-scale climate model data

This application paper presents a visual analytics tool designed to explore large-scale scientific data modeled after a natural climate phenomenon. The data are modeled on a high-performance computer and exported to a personal computer for interactive visualization. The system is co-designed by visual analytics researchers and domain scientists after a year of rapid prototyping and evaluation of multiple information and scientific visualization techniques using a model dataset that includes both scalar fields and flow fields. Five information-visualization and one scientific-visualization techniques are included in the visual analytics system to balance analytical effectiveness and computation time for large-scale interactive exploration. The paper discusses the system design, explains the design rationale, and shares computation performance and results of different visualization techniques. The primary contribution of this application paper is to show that we can interactively and effectively visualize a large amount of scientific model data on a modest desktop computer. The computation performance results of the individual visualization techniques and the overall system also provide benchmark references for other large-scale visualization development efforts.

Shen et al. introduced a two-tier visual analytics system, which incorporates a Hadoop-based parallel data processing platform to handle queries of massive web session data.

/*

Visual analytics of large-scale climate model data

*/

3. B/S

A functional framework for Web-based information visualization systems

The accelerating evolution of information visualization research in the last few years has led to several specific system implementations. The obvious drawbacks of this development are highly dependent software systems, which are only available for a restricted number of users. Today, due to the remarkable advances in hardware and software technologies, not only very expensive graphics workstations, but also low-cost PCs are capable of running computational demanding visualization systems. Furthermore, the rapid development of the medium World Wide Web along with state-of-the-art Internet programming techniques has led to a trend toward more generally usable visualization systems. In this paper, we propose a functional developer's framework for general Web-based visualization systems which makes intelligent use of application specific software and hardware components on the server side, as well as Java's benefits on

the client side. To demonstrate the framework's abilities, we have applied it to two practical visualization tasks and report on our experience concerning practicability and pitfalls

Bender et al. presented a framework which allows the easy and efficient implementation of general Web-based visualization systems which can either follow the fat server, the fat client, or a hybrid approach.

A scalability study of web-native information visualization

Several web-native information visualization methods (SVG, HTML5's Canvas, native HTML) are studied to contrast their performances at different data scales. Using Java implementations of parallel coordinates and squarified treemaps for comparison, we explore the design space of these web-based technologies in order to determine what design trade-offs are required.

Johnson et al. contrasted the performances of several web-native information visualization methods (SVG, HTML5's Canvas, native HTML) at different data scales.

Thin Client Visualization

We have developed a Web 2.0 thin client visualization framework called GeoBoosttrade. Our framework focuses on geospatial visualization and using scalable vector graphics (SVG), AJAX, RSS and GeoRSS we have built a complete thin client component set. Our component set provides a rich user experience that is completely browser based. It includes maps, standard business charts, graphs, and time-oriented components. The components are live, interactive, linked, and support real time collaboration.

Eick et al. developed a thin client visualization framework that provides a rich user experience that is completely browser based.

/*

The Scalable Reasoning System: Lightweight Visualization for Distributed Analytics

A central challenge in visual analytics is the creation of accessible, widely distributable analysis applications that bring the benefits of visual discovery to as broad a user base as possible. Moreover, to support the role of visualization in the knowledge creation process, it is advantageous to allow users to describe the

reasoning strategies they employ while interacting with analytic environments. We introduce an application suite called the scalable reasoning system (SRS), which provides web-based and mobile interfaces for visual analysis. The service-oriented analytic framework that underlies SRS provides a platform for deploying pervasive visual analytic environments across an enterprise. SRS represents a 'lightweight' approach to visual analytics whereby thin client analytic applications can be rapidly deployed in a platform-agnostic fashion. Client applications support multiple coordinated views while giving analysts the ability to record evidence, assumptions, hypotheses and other reasoning artifacts. We describe the capabilities of SRS in the context of a real-world deployment at a regional law enforcement organization.

Pike et al. introduced the scalable reasoning system (SRS), which provides web-based and mobile interfaces for visual analysis.

*/

---

4. adaptive LOD

ASK-GraphView: A Large Scale Graph Visualization System

We describe ASK-GraphView, a node-link-based graph visualization system that allows clustering and interactive navigation of large graphs, ranging in size up to 16 million edges. The system uses a scalable architecture and a series of increasingly sophisticated clustering algorithms to construct a hierarchy on an arbitrary, weighted undirected input graph. By lowering the interactivity requirements we can scale to substantially bigger graphs. The user is allowed to navigate this hierarchy in a top down manner by interactively expanding individual clusters. ASK-GraphView also provides facilities for filtering and coloring, annotation and cluster labeling.

Abello et al. realized a scalable architecture and sophisticated clustering algorithms to visualize large graphs with interactive rates.

HiMap: Adaptive Visualization of Large-Scale Online Social Networks

Visualizing large-scale online social network is a challenging yet essential task. This paper presents HiMap, a system that visualizes it by clustered graph via hierarchical grouping and summarization. HiMap employs a novel adaptive data loading technique to accurately control the visual density of each graph view, and along with the optimized layout algorithm and the two kinds of edge bundling methods, to effectively avoid the visual clutter commonly found in previous social network

visualization tools. HiMap also provides an integrated suite of interactions to allow the users to easily navigate the social map with smooth and coherent view transitions to keep their momentum. Finally, we confirm the effectiveness of HiMap algorithms through graph-travesal based evaluations.

Shi et al. implemented the methods of adaptively visualizing the online social networks with clustered graph.

Level-of-Detail Visualization of Clustered Graph Layouts

The level-of-detail techniques presented in this paper enable a comprehensible interactive visualization of large and complex clustered graph layouts either in 2D or 3D. Implicit surfaces are used for the visually simplified representation of vertex clusters, and so-called edge bundles are formed for the simplification of edges. Additionally, dedicated transition techniques are provided for continuously adaptive and adjustable views of graphs that range from very abstract to very detailed representations.

Balzer et al. presented level-of-detail techniques for visualizing large and complex clustered graph layouts with implicit surfaces.

Node-attribute graph layout for small-world networks

Small-world networks are a very commonly occurring type of graph in the real-world, which exhibit a clustered structure that is not well represented by current graph layout algorithms. In many cases we also have information about the nodes in such graphs, which are typically depicted on the graph as node colour, shape or size. Here we demonstrate that these attributes can instead be used to layout the graph in high-dimensional data space. Then using a dimension reduction technique, targeted projection pursuit, the graph layout can be optimised for displaying clustering. The technique out-performs force-directed layout methods in cluster separation when applied to a sample, artificially generated, small-world network.

Gibson et al. showed an optimization for small-world network layout by using attributes to layout the graph in high-dimensional data space.