A Temporal and Multi-Resolution Visualization System for Large-Scale Data

LIEU-HEN CHEN1,+, YU-SHENG CHEN1, WEI-FAN CHEN1, HAO-MING HUNG1
AND YASUFUMI TAKAMA2

1Department of Computer Science and Information Engineering
National Chi Nan University
Nantou, 545 Taiwan
2Faculty of System Design
Tokyo Metropolitan University
Tokyo, 192-0397 Japan
E-mail: lhchen@csie.ncnu.edu.tw

In this paper we propose a multi-resolution visualization approach based on a force-directed algorithm to graphically and hierarchically represent large-scale data sets. Considering the limitations of the human visual cognitive system, it is reasonable to dynamically and interactively extract the most important or user-specified information for representation. To smoothly animate the merging or splitting of nodes and edges in a graph, we adopt a jelly-like metaball technique. Our prototype system clusters and graphically represents thousands of nodes and links in a novel way that allows a user to recognize visually. By adopting our system, the hierarchy of information and the evolving trends are dynamically visualized through an interactive 3D graphical interface, in addition to simultaneous display of data detail and summarization. Furthermore, user perception can be enhanced by adopting human 3D stereopsis skills such as Perspective, Chiaroscuro, and Focusing method.

Keywords: information visualization, human-machine interface, graph animation, metaball, hierarchical structure, human perception

1. INTRODUCTION

As the Internet continues to evolve, massive amounts of data are being produced in various formats and accumulated in different ways. Unfortunately, neither the human visual cognitive system nor display technology has kept pace with this dramatic evolution.

Research has provided methodologies which are helpful for searching, analyzing, data-mining and categorizing information, allowing data with high degrees of connectivity and relativity to be grouped hierarchically. From a graph theory perspective, nodes and edges can be used to represent data and relationships, respectively. However, when the content and relationships of a large-scale data set are transformed and represented in this way, the sheer number of nodes and edges generated on a graph become far too great for users to manage effectively. Therefore, without an effective visualization mechanism, infographics of large-scale data sets can only help users shift from an overwhelming 1-D text/digit riddle into a chaotic multi-dimensional information labyrinth. It could be said that in many cases, the more data system provides, the less information users receive. An
example is shown in Fig. 1. This graph contains numerous nodes and edges, making it too complex to understand. If there is no clustering, filtering or other strategies to organize the data, the graph would not be suitable to represent large-scale data.

In this paper, we propose a multi-resolution, hierarchical and interactive visualization system for representing large amounts of data in the form of a 3D graph. The reason of using 3D structures is that users can change viewpoint of the infograph in six degree-of-freedom (rotation and translation). In layout problem, 3D graph has more space and flexibility than 2D [1]. To start, we hierarchically group highly-related data into clusters, and adopt a recursive force-directed algorithm to allocate nodes and groups in a 3D space. We then integrate metaballs and concepts of LOD (level of detail) to represent data that can be controlled by the user. Our approach not only aims to improve the efficiency of infographing large-scale data sets, but also provides a plausible and recognizable visualization result.

The remainder of this paper is organized as follows. First, in Section 2, a brief review of related works is presented. Our method and system configuration are then introduced in Section 3. Experiment result images are shown and discussed in Section 4. Finally, we conclude in Section 5.

2. RELATED WORK

To meet the different needs of users, M. L. Huang suggested graphs of three levels of representation [2]. The highest level of abstract graph representation displays the ab-
extract structure of a graph. The middle level of geometric representation displays the relationship between nodes. The lowest level of graphical representation focuses on representing node attributes and their relationship with other nodes.

To determine the positions of nodes in a graph according to their relationship between each other, a force-directed method can be adopted. As one of the pioneers, in 1991, Thomas M. J. Fruchterman et al. proposed a specific and intuitive force-directed algorithm [3]. After then, Y. Koren et al. proposed another approach with a binary stress model to deploy a graph in a 3D space [4]. Koren’s binary stress model merges the advantages of both stress and electric spring models. By minimizing these two models, the relative position and distance between nodes are calculated according to their path lengths. H. Omote et al. proposed arranging intersecting clustered graphs by using an alternate electric model based on a spring model and Coulomb’s law [5].

A. Quigley et al. tried to accelerate force calculations by using a Quad-tree/Oct-tree for spatial partitioning [6]. When the distance from the current node to all nodes in a cluster is larger than the threshold, a pseudo-node is used instead to make a rough estimate of the approximate force from all nodes in the cluster to the target node.

C. Walshaw described a multilevel technique to enhance the performance of force-directed algorithm [7]. C. Muelder et al. proposed a method to apply a treemap to a clustering hierarchy for the graph. This deploying strategy can greatly reduce the iteration times [8].

M. Balzer et al. were the first to adopt the level of detail (LOD) concept to organize infographics of large-scale data sets [9]. They also proposed using implicit surfaces to simultaneously represent the boundaries of different levels.

Y.-H. Chan et al. were able to improve user comprehension of information by adopting different representation styles. They abstract data either by merging multiple paths or clustering nodes of the same types [10]. Moreover, metadata of each entry is used to enhance its relationship with others.

J. Abello et al. proposed a node-link-based graph visualization system, which clustered the nodes into groups to form a hierarchical graph. Hierarchical structures are highly effective for large-scale data set application. For interactive navigation, a structural system must represent context when users explore it [11].

C. Friedrich et al. noted that a graph animation must preserve a mental map. In graph animation, minimal changes help smooth transitions and also define the criteria for good animation [12, 13]. C. Erten et al. proposed evolving graphs to represent 2D/3D graph animations of evolution. In any timeslice, changes to the layouts of the graphs would be force-directed [14].

When animation display is too fast or complex, viewers are unable to accurately perceive it. Integration of interactive systems to control the animation is necessary [15].

These force-directed approaches provide a reasonable approach to deploy data on a 2D plane or a 3D space. However, because they need to iteratively perform force-calculations until the positions of all nodes are stabilized, when the size of a graph grows, the time costs for calculations increase disastrously. Moreover, these pioneering efforts failed to consider the upper bound of the human visual recognition system in the representation stage. The above problems become very critical issues for any interactive system, which has to dynamically represent large-scale data in a user-recognizable way. Therefore, this paper proposes a novel approach to improve both of the efficiency and
the visual recognizability of infographing large-scale data sets.

Our approach represents large amounts of information through animation is to display a hierarchical and dynamic graph organized by multiple levels of resolution. This approach allows users to change their level of observation from very abstract to very detailed, and observe the whole evolvement of situation interactively.

3. SYSTEM CONFIGURATION

Our system can be divided into two parts. For research convenience, we do not delve too deeply into the field of data mining. Instead, we use a simple and straightforward approach to extract data relationships. We first use a text parser to extract possible candidates for keywords from the provided data, typically a short article or essay for a certain subject. We then accumulate the frequency of occurrence for every keyword candidate across that data. Finally, the keywords with the highest frequency of occurrence are obtained.

3.1 Graph Construction

In the first pass, we begin to translate the abstract data into a recognizable graph. Because the distribution of nodes and edges can be very complicated in large-scale data sets, conventional force-directed algorithms commence by randomly deploying all nodes. The positions of each node are progressively refined by iteratively calculating the attractive and repulsive forces among them. The computational time required for this approach grows in proportion to the square of the number of nodes. In addition, unpredictable iteration times cause the force-directed approach to be impractical for a visualization system, which has to represent dynamic information in real-time. Moreover, as shown in Fig. 2, the result images are generally too complicated to be easily understood by users.

Fig. 2. Metaball graph. (60 nodes)

To keep a graph recognizable and comprehensible, we believe that it is important to control the total number of vertices simultaneously shown on a screen. At the same time, it is also important to reasonably deploy the 3D positions of vertices, so that the relationships between data can be geometrically reflected.

To achieve these goals, we first hierarchically cluster data nodes with high relativity. We then adopt a force-directed algorithm to progressively deploy nodes and groups into a 3D space. This divide-and-conquer approach prevents applying expensive force calculations. It also provides a hierarchical spatial structure with vertices being appropriately distributed in a flexible way. To be more specific, we list the part of pseudo-code as follows:
Fig. 3. Algorithm of clustering nodes into a hierarchical structure.

Algorithm 1 Cluster Nodes into hierarchical structure

Require: All nodes
Ensure: hierarchical structure
1: while the number of nodes in a group > thresholdV do
2: Assign intern-related k nodes (i.e. contain same features) into a new subgroup
3: end while
4: while the number of subgroups > thresholdV do
5: Merge the smallest subgroup to the second smallest one
6: end while
7: Cluster the structure of subgroups

Fig. 4. Algorithm of allocating edges of internal nodes.

where thresholdV is the threshold for visualization. The predefined constant, thresholdV, determines the maximum number of vertices which can be simultaneously shown on screen. In this pseudo-code, it is reasonable to cluster nodes and subgroups into a parent group according to their similarities. In addition, for each iteration of force-directed calculation during the deployment stage, the number of nodes and groups is controlled by the value of thresholdV.

Our grouping algorithm is divided into three passes. The first pass groups N nodes top-down into a hierarchical structure. The weighting values of edges are then accumulated bottom-up. Finally, all nodes are top-down deployed hierarchically using a force-directed algorithm. Because the number of nodes and groups on the same level are less than or equal to thresholdV, the computational time of each iterating step of force-directed calculation in the deployment stage is $O(thresholdV^2)$. Most important of all, the total iteration times to reach a stable situation of deployment are dramatically reduced in comparison with the conventional approach which uses all nodes simultaneously.

3.2 Visualization Based on Metaballs

In the second pass, we need to hierarchically render and animate our multi-resolution group while interacting with users in a way which intuitively conforms to the user’s own experience. To do this, metaballs are used to create smooth-surfaced images of organic-looking objects [16, 17]. Metaballs are often used to simulate the visual merging and splitting effects of water drops and molecules.
Our approach adopts metaballs for the following purposes:

1. Animating the merging and splitting effects of connected groups during the changes between different resolutions, as shown in Figs. 5 (a)-(c).
2. Emphasizing the relationships between groups by graphically representing the set intersection and set union operations, as shown in Figs. 5 (d)-(e).
3. Representing the propagation paths of information between nodes and groups. Conventional metaball approaches use special functions to calculate the color of each pixel by comparing its density with the thresholds. In a 3D environment, the density of each pixel is calculated using the following equation.

$$F(x, y, z) = \frac{r_i^2}{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2}$$

where $r_i$ is the radius of a metaball and $x_i, y_i, z_i$ is the center.

If the value of $F(x, y, z)$ is smaller than or equal to the threshold, the projected pixel $P(x', y')$ is considered to be located on or included within the range of a metaball’s surface.

With the exception of the highest view level, our prototype system always aligns the screen center to the last user-selected metaball. In the case of highest level, a dummy group is used as a virtual root and is not rendered. With the metaball fixed to the center of the screen, users can zoom in, zoom out, and rotate their viewing angle. Users may switch between different resolutions by double left-clicking on a group shown on the screen.

A filter can be used to distinguish and highlight objects. For instance, the discussions for one topic can be classified into several groups. Users can emphasize the existence of interested groups and links by adopting the light source. The balls of user-selected group and connected groups are consequently highlighted, and non-directly related balls are masked. As shown in Fig. 6, different groups are briefly assigned with different colors. Then, users select the red group “Olympic” to highlight the related groups “world” and “olympics”, which are connected to the selected group. This filter can help the user to focus on groups of interest without replacing the entire graph.

In addition, we add a function for user to input a keyword for highlighting groups which contain the keyword. As shown in Fig. 7, if a user inputs the keyword “Brazil”, the groups “brazil”, “world”, and “cup” are highlighted. This means these groups contain the keyword “Brazil”. This function provides a clue for users to easily focus on potentially related groups.

Moreover, in our visualization system, the Chiaroscuro method can be used to en-
hance users’ stereo perception. As shown in Fig. 8, shadows are useful since they not only add realism, but also provide important visual cues which help users understanding relative object placement in a 3D scene.

By utilizing the Perspective method and hidden surface removal process, information with lower priority can be easily filtered out. This technique is useful for concentrating users’ attention on more important or interesting topics. At the same time, our system provides an auxiliary visualization mode, which rotates the viewpoint automatically, to remind users about the global structure and prevent some detail from being ignored in Fig. 9.
4. IMPLEMENTATION

In this section, we discuss the implementation of our system with experiment results. To keep the screen simple and clean for user comprehension, when the number of visible groups or nodes exceeds the threshold $V$, some of them will be wrapped into a subgroup and hidden from the screen. When there exists any group containing nodes more than threshold $V$, the system will analyze keywords in subgroups, and use the keyword with highest frequency for further classification. This strategy gives different meaning for each hierarchy, and enables users to select and observe the subgroups more precisely. Using this strategy, data is consequently organized in a multi-resolution format.

In the first experiment, our system collected more than 10,000 comments about term “Chrome” and displays them in a hierarchical structure. The highest level provides users with an abstract of the whole infograph, and the lowest level shows precise detail of a section of an information network, as shown in Figs. 10 and 11 respectively. Furthermore, by adjusting the size of threshold $V$, users can control the degree of data abstraction as shown in Fig. 12.

Fig. 9. The viewpoint moves around the groups.

Fig. 10. An abstract of the whole infograph. The threshold $V$ is 16.
Because data are graphically arranged and represented in a multi-resolution 3D manner, users can intuitively traverse these hierarchical networks by entering a group for more detail, or exiting a node for a larger global presentation while preserving context. By rotating view angle and zooming in and out, users can readily observe and understand the graphs. When a new group is inserted into the graph, any groups which have a positive correlation to this node will be attracted toward the new node’s group. Therefore,
the size of a group basically reflects the number of similar data/comments, and the distance between them reflects the strength of correlation based on degree.

The second experiment deals with temporal graphs. To keep the screen simple and clean, we set the default value of \( \text{thresholdV} \) at 8 in this experiment. This graph represents specific time-ordered comments that were displayed according to update time. The system shows how we constructed the keywords table, which serves as the basis for grouping data. After users explicitly defined a certain keyword in which they are interested, or implicitly provide a starting data entry for analysis, we collected comments or discussions from certain original thread by using widely available search applications, e.g., Google Plus API. In addition, we classified these comments into different groups based on keywords. The keywords were extracted and counted by adopting a stemming algorithm, Snowball [18]. In our example, the term “Google Apple” was defined by users and used as the searching scope. By utilizing Snowball, keyword candidates were selected from existing online articles and then prompted to users. During visualization, not only the web-mined information changed, but the grouping basis was also dynamically updated by accumulating and sorting the weighting value of each keyword in real time.

Nodes are sequentially inserted and rendered in accordance with the submitted time of each article to enhance user perception to the trend of information. Moreover, users can replay changes to the infograph forward or backwards by utilizing a time scroll bar located at the bottom of the screen in Fig. 13. In this example, the term “Google Apple” is used as the web-mining target for visualization. As shown in Fig. 13 (a), of the first 90 comments, half of the comments are related to Google. After 72 days later, more than 400 comments were shown. However, the number of comments about Apple is increased and exceeds the number of comments about Google. (In fact, the Apple CEO has announced the new iOS version and other products at the Apple Worldwide Developers Conference on June 10 to 14, 2013), as shown in Fig. 13 (b). Three weeks later, 1720 comments were collected, grouped and hierarchically visualized as shown in Fig. 13 (c).

Fig. 13. System displays of the relativity between different threads, along with their comments and groupings. Each group displays the number of nodes contained within.
Fig. 13. (Cont'd) System displays of the relativity between different threads, along with their comments and groupings. Each group displays the number of nodes contained within.
In the case of the traditional text-based interface, all of this information was sequentially represented for browsing. As time went by, it became very difficult for users to follow the trend because of the large number of comments. In contrast, according to user feedback, the time scroll bar approach used in our system greatly improves user perception as to how trends evolve over time. Furthermore, feedback results indicate that the 3D graphical structure also helps users to find comments of interest in an intuitive manner.

5. CONCLUSION

In this paper, we proposed a temporal and multi-resolution visualization system for representing large-scale data sets in the 3D graph form. In our system, not only is the final static 3D infograph visualized, but also dynamic animation of the evolution of the information as it changes and evolves over time, providing users with a unique and highly useful approach to understanding and interactively assessing large amounts of data.

By adopting a divide-and-conquer strategy, our system efficiently deploys data in a 3D space based on a force-directed approach. Each data item is represented as a node, and the relationship between two items is represented as an edge. Our approach determines the importance of a node depending on its degree, i.e., the number of edges which lead to it. A node of higher importance has greater mass, and thus a greater influence on other nodes.

Our approach reflects human experience in an intuitive system that can handle tremendous numbers of elements. This similarity is intentional, and helpful for users to quickly acquire a global image of a very complex large-scale data set. Furthermore, by integrating the concept of LOD with the human visual recognition system, our system not only improves the efficiency and preserves the spatial flexibility of graph deployment, but also provides a user-friendly interface for visualizing graphs.

The degree of importance assigned to data may vary from case to case, and one of the advantages of our system is that it can be easily adapted to different situations because the data mining part is clearly separated from the infographing and visualization modules. However, there is no guarantee that what has been determined as the most important data will always be of the most interest to individual users. Therefore, as part of our future work, we plan to provide support for users to manually select several data entries, along with a simplified network structure to set an initial state as the starting constraint for the force-directed algorithm.

In our prototype system, we construct the initial table of keywords for data grouping by extracting those possible candidates from previously related documents and comments on the Internet. It is necessary to improve the current keyword extraction approach used in our prototype. More consideration of both the semantics and pragmatics of keywords and sentences are also necessary for grouping data in a more appropriate manner.

Moreover, in the current stage, node colors are assigned simply depending on their degrees of connectivity. A node with higher degree tends to be assigned a warmer and brighter color. More investigation into an adaptive and flexible coloring strategy which takes the theory of color psychology and the semantics of data into consideration is necessary. In addition, a data-mining mechanism for extracting and abstracting text content,
such as common keywords between groups, may be helpful.

A demonstration video of our system is available at http://cgda.csie.nctu.edu.tw/Info-
Vis/TV.mp4.

REFERENCES


**Lieu-Hen Chen** (陳履恆) is currently an Associate Professor of the Department of Computer Science and Information Engineering at the National Chi Nan University, Taiwan. His research interests include computer graphics and digital arts. He received a B. S. in Computer Science and Information Engineering from the National Taiwan University in 1989, and his M.S. and Ph.D. degrees in Electrical and Electronic Engineering from the University of Tokyo, in 1994 and 1998, respectively.

**Yu-Sheng Chen** (陳昱升) is currently a Ph.D. candidate in the Department of Computer Science and Information Engineering at National Chi Nan University. His research interests include information visualization, level of detail, computer animation, and virtual reality. Contact him at s98321902@ncnu.edu.tw.

**Wei-Fan Chen** (陳偉瑋) is currently a graduate student in the Department of Computer Science and Information Engineering at National Chi Nan University. His research interests include level of detail, computer animation, and virtual reality. Contact him at s100321522@ncnu.edu.tw.
Hao-Ming Hung (洪皓銘) is currently a university student in the Department of Computer Science and Information Engineering at National Chi Nan University. His research interests include information visualization, computer animation. Contact him at s99321024@ncnu.edu.tw.

Yasufumi Takama (高間康史) is currently an Associate Professor of the Faculty of System Design at the Tokyo Metropolitan University, Japan. His research interests include visual data mining, web intelligence, information visualization, recommendation, intelligent interactive systems, Artificial Intelligence and Soft Computing. He received the B.S. degree, M.S. degree, and Dr. Eng. degree from University of Tokyo in 1994, 1996, and 1999 respectively. Contact him at ytakama@sd.tmu.ac.jp.