

# Information Visualization Techniques for Building Better Visualization Models



Rachael Fernandez and Noora Fetais

## 1 Introduction

Recent advances in Information Technology have led to the generation of vast amounts of data that overwhelm the user. This data deluge makes it hard to extract useful information and the user is often swamped with ginormous amounts of data. Much of this data is in textual format, and this makes it harder to make sense of the data at hand.

Card et al. defined information visualization as the “use of computer supported, interactive, visual representations of data to amplify cognition” [1]. Visualization technologies help the users to identify patterns or abnormalities in the data so that the user can make informed decisions. At the high level, Information Visualization (IV) systems are made of two components, namely, Visual Representation and Interaction [2]. Visual Representation refers to the mapping of data to the objects in the scene. However, without any interactivity, the information rendered by the visualization systems are just static and do not fulfill the purpose of communicating the entire picture to the users. For example, a static image graph is much less effective than an interactive graph in which the user can filter the data that is displayed.

This leads to the need for developing interactivity techniques, which help the user in exploring the data. The interaction with the visualized scene starts with the need to explore the data, which helps the user to better understand the data and the visualized scene. Once the user has completed the interaction, the system processes the input and provides feedback to the user. The users often interact with the visualization system to adjust the following options [3]:

---

R. Fernandez · N. Fetais (✉)  
KINDI Center for Computing Research, Doha, Qatar  
e-mail: [rf1405233@qu.edu.qa](mailto:rf1405233@qu.edu.qa); [n.almarri@qu.edu.qa](mailto:n.almarri@qu.edu.qa)

1. **Modify the data to be visualized:** The user can explore the various facets of the data by selecting and filtering information based on the needs. This helps in viewing an overview of the data or exploring some subset of the data in detail.
2. **Customize the display:** The users often interact with the system to customize the visualization screen. For example, the screen brightness, background color, or visualization shapes can be changed as per the user's requirements. These modifications help in exploring of the data smoother for the user.

Interactive Visualization systems help to combine representation and interaction of data, where an overview of the data is provided, followed by the details of the selected object as and when they are demanded by the user. The use of interactive options on the visualized object helps us to interact and manipulate the data object as opposed to viewing a static image [3, 4].

A complete Visualization System is constructed by going through a series of stages, which are often referred to as the Visualization Pipeline. According to [8], the pipeline transforms the data (value) into the visualization (view). The Prefuse Toolkit describes a simple visualization pipeline, in which the abstract data that is to be visualized is first filtered and transformed into a visual form. This transformation is done by applying the visual attributes on the objects to be visualized. These objects are finally rendered on the scene and can be viewed by the user [9].

This chapter is structured as follows: Sect. 2 discusses a simple visualization pipeline. A list of attributes to enhance representative and interactive visualization techniques for two-dimensional (2D) scenes is described in Sect. 3. Section 4 introduces visualization techniques for enhancing cognition levels in three-dimensional (3D) scenes. This is followed by a short discussion of the popular visualization tools and the functions that are available in each of these tools in Sect. 5. The chapter concludes with a summary in Sect. 6.

## 2 Visualization Pipeline

Existing visualization pipelines focus on the representative visualization module of the model and ignore the integration of the interactivity attributes into the model. In this section, we will outline the general steps that are required for creating a visualization model that couple both representative and interactive techniques.

The pipeline goes through the following five steps:

1. **Map to a Geometrical Shape:** The geometrical shape to which the data can be mapped to must first be identified. For example, a polygon with multiple vertices can be used for representing multidimensional data in a fixed-range, whereas concentric-circles can be used for representing different levels of access.
2. **Layout:** If the object is made up of multiple objects, then the layout and the relationship between the different objects have to be defined. For example, in a simple 2D chart, plotting of values along the axes indicate an increase of the value.

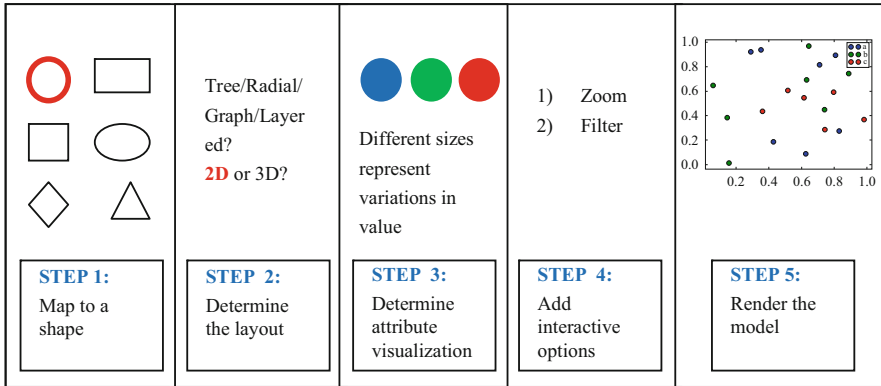


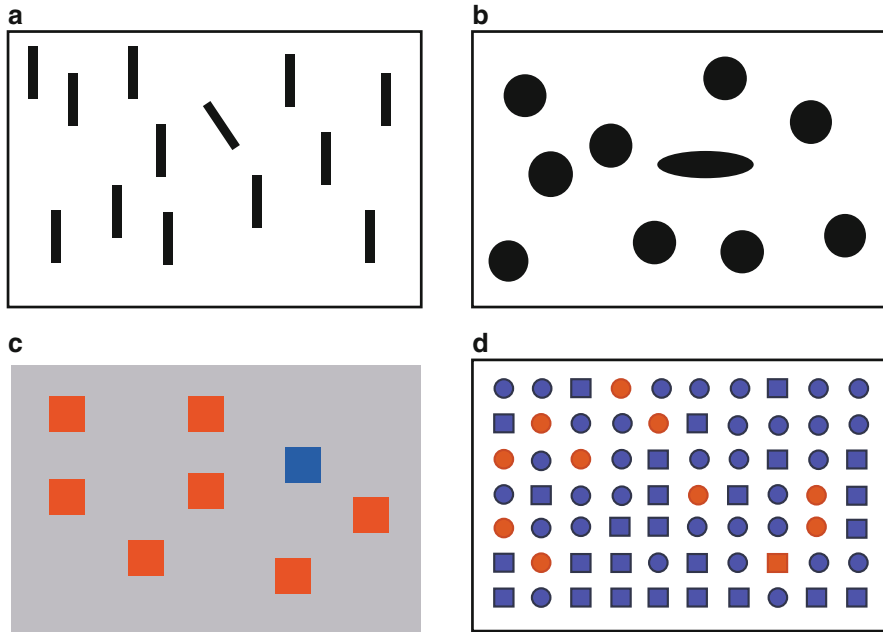
Fig. 1 Steps in the visualization process

3. **Attributes:** It should be ascertained, that all the attributes of the data can be conveyed through the visualized object. For example, the marks of two students can be differentiated by using two circles of different colors and making the size of the circles in proportion to the marks that were obtained.
4. **Addition of Interactive Options:** Interactive options such as zooming, filtering, and panning can be added. Interactive options help in better exploration of the data and increase engagement with the user.
5. **Rendering:** The visualized object is finally rendered on screen after setting parameters like camera perspective, lighting, and depth in the case of multidimensional data.

It must be noted that the pipeline is an iterative process that requires the scene to be updated whenever the user modifies the objects in the scene. Figure 1 depicts the different steps in the visualization process.

### 3 Visualization Techniques

Visualization techniques are useful for showing an overview of the data, followed by details of the object as and when the user explores the visualized scene. In this section, we will discuss the important attributes required for good representation of the visualized objects and for interaction with these objects in the visualized scene (Fig. 2).



**Fig. 2** Importance of using different colors and shapes [5]. (a) Different orientations are easier to notice. (b) Different shapes are easier to notice. (c) Different colors are easier to notice. (d) Not as easy to notice the orange square amidst other objects with similar shapes and sizes

### 3.1 Representative Visualization Techniques

Some of the representative visualization techniques that will enhance the quality of information that is delivered to the user are given in the following [6]:

1. **Shape:** A suitable shape should be used to convey the information. For example, different shapes can be used to signify different objects. For example, the marks of boy students can be represented by a square and that of girl students using a circle in a scatter-plot. Variety of shapes have been experimented with and used to build a visualization model and each of these shapes is better suited for conveying a specific type of information. The iBlogVis system uses a diamond to view the content of a blog and a line to represent the number of characters in a blog [7].
2. **Size:** The size of the objects can be used to indicate the importance or the value of the object in proportion to other objects that are visualized in the scene. The size of the objects refers to the length and width of the object.
3. **Position:** The position of the object in a 2D scene can also be used to represent a parameter. For example, time could be set as the X-axis that would help to find the time at which the event occurred.

4. **Color:** Instead of using different shapes, we can also use different colors (hues) or different intensities of the same color to signify different events or objects.
5. **Orientation:** The orientation of the object in a scene is also easily noticeable.
6. **Depth:** The illusion of depth can be given to objects to visually represent the increasing/decreasing rate of an object.
7. **Texture:** Different textures can be used in different parts of the same object or in different objects to distinguish the objects. It is also used as an indicator of the material of the object. For example, the presence of different types of rocks in a region can be differentiated using textures to mark out the places in a map; it can be used in the medical field for visualizing the anatomical structure [4].
8. **Opacity:** The data that is represented on screen can be represented with different levels of transparency to indicate different levels of saturation of the item. For example, different shades of red can be used for displaying different levels of severity of an event.
9. **Labeling:** The data should be labeled with clear descriptions that will enhance the understanding of the user. For example, the axes should be labeled properly in the case of chart representations.
10. **Layout:** The model is often visualized in different layouts that depend on the type of the data. Some of the popular layouts are **Tree, Radial, and Graph Layouts**. Often, models are created that combine the strengths of these layouts to form a stacked model with different layers [9]:
  - (a) **Tree Layout:** This layout is best suited for viewing hierarchical data [10, 13] (e.g., Organization Chart, Family Tree). A simple type of tree layout is used for constructing Pedigree Trees that display a family’s ancestral roots, as depicted in Fig. 3. It can be modified to add information regarding the birth and death

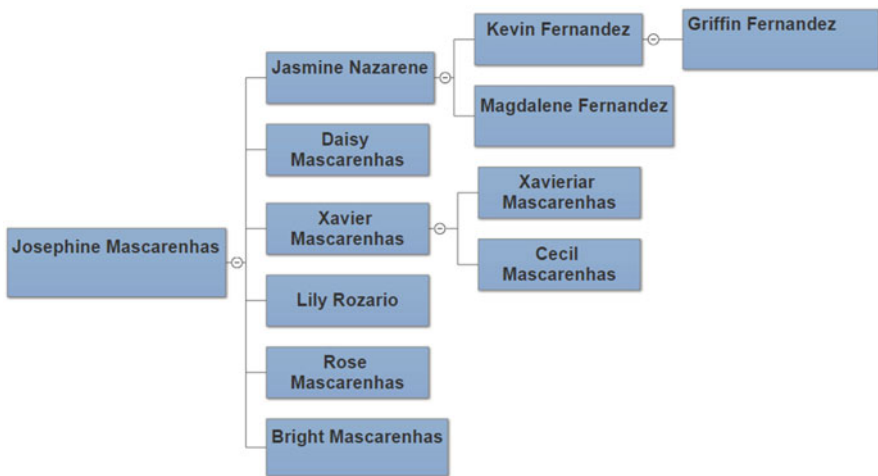


Fig. 3 Pedigree Tree denoting the ancestral roots of a user Josephine

dates of each person. However, these trees expand largely in size as the number of generations increase and are later replaced by Fan Charts, which follow a circular layout that makes them more compact than Pedigree Trees [10].

Fish eye tree views can be used for representing hierarchical textual information. The data in different levels of the hierarchy are depicted by varying degrees of indentations. The data can be viewed in different stages by collapsing or expanding the data in different levels [12].

A variation of trees called Tree-Maps are used for visualizing highly dimensional data (e.g., Medical data) [11, 14]. Tree-Maps are used for visualization of hierarchically *structured* information in a compact rectangular region that is completely filled with information. The region is partitioned into a set of boxes that accommodate nodes. The area occupied by a node in a box depends on the weight or importance of the node.

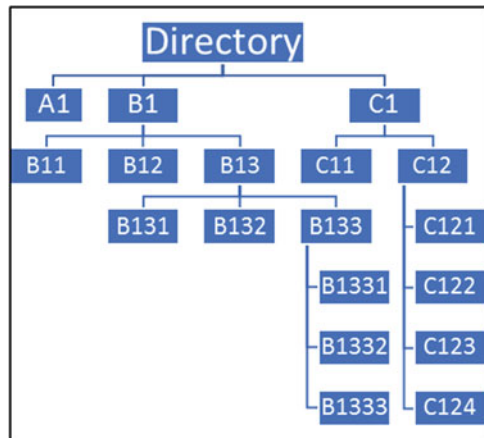
Tree-Maps are capable of conveying two types of information; *structural information* about the hierarchy and some *content information* associated with each node. Consider the following example that uses a Tree structure (Fig. 4) and nested Tree-Maps (Fig. 5) to visualize a directory structure.

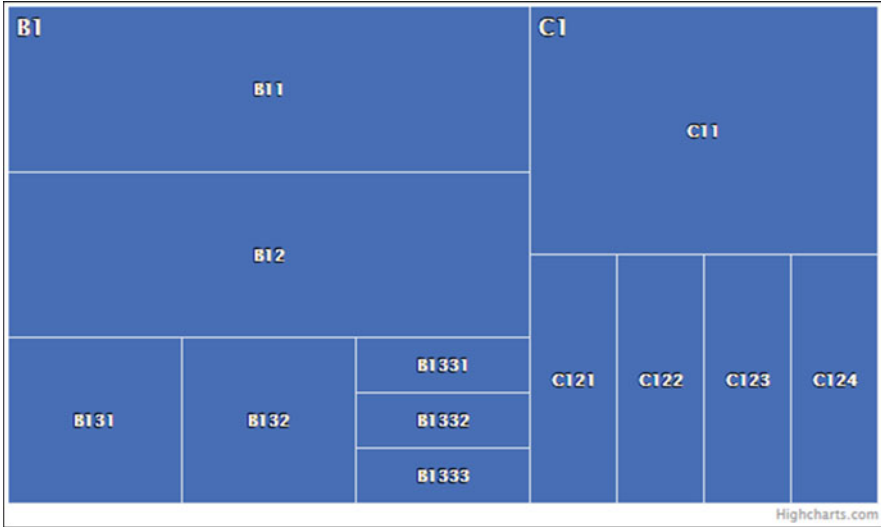
The Tree-Map is adept at visualizing the structure of files inside the various folders in a directory. In a colored Tree-Map, the boxes at the same level in a hierarchy share the same color. In our example, folders B11, B12, B13, C11, and C12 would share the same color (Fig. 5).

While Tree-Maps are 2D structures, Beamtrees on the other hand are 3D in nature. They have the added information of the depth of a node. Beamtrees are created by scaling Tree-Maps.

As Tree-Maps already make use of all the available space in a rectangle, all the boxes inside the rectangle are scaled down in size to a single-dimension structure, which is referred to as the width. After all the boxes have been scaled, all the leaves are assigned to their parent boxes [15].

**Fig. 4** Tree representation of a file system





**Fig. 5** Tree-Map representation of a file system

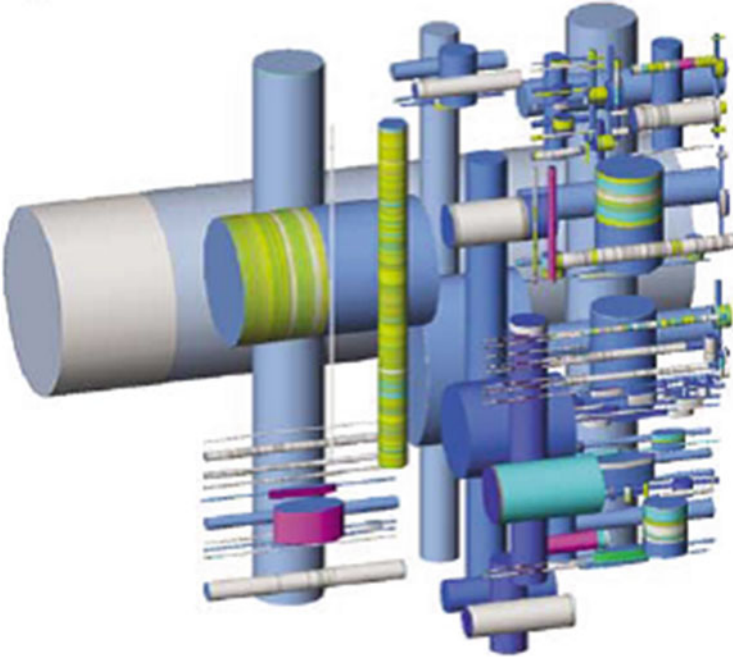
Figure 6 depicts a 3D view of the file system in which all the files in the same color are at the same level in the hierarchy.

- (b) **Radial Arrangement:** It is a highly versatile model, using which the user can visualize multidimensional data using slices. It is also possible to visualize the different layers of a system using concentric circles. For example, each concentric circle could represent different levels of security [12, 17].

Radial visualization refers to the practice of displaying information within a circular structure. The most popular form of radial displays is the Pie Chart, in which the chart is divided into sectors to visualize the proportions of each element. However, finding the bigger “slice in the pie” is quite difficult. To overcome this difficulty, a pie chart variant that uses concentric circles instead of sectors is used [12].

A combination of these two forms of pie charts is used to visualize events in a file system [17]. In the impromptu system (shown in Figure 7), each slice of the pie is used to represent a single user and the model uses a different color for each user. These slices collapse as a user logs out of the system and the files accessed by each user are visualized as pixels in the slice allocated to the user. Each concentric circle represents a level of permission that is given to the users for accessing a set of files. For example, files in the outermost circle can only be read, whereas the files in the innermost circle can be both read and written on.

Another interesting method is followed in the VisAlert system, which visualizes the network events in a radial enclosure. In this system, an event is described by three attributes; *What*, *When*, and *Where*. Each event is mapped to a 2D space representing the *What* and *Where* attributes as they are finite.



**Fig. 6** 3D Beamtree representation of a file system using a [15]

A straight line is drawn from this point to the event's *When* attribute to represent the relationship between the three attributes. Figure 8 shows the VisAlert system after the events were visualized [18].

The Visual Thesaurus is an interactive dictionary that displays the selected word in the center and its related words in a radial fashion around the selected word [19]. This helps in specifying the synonyms of the word in a concise manner within a small area. When one of the related words is clicked, the entire arrangement shuffles around to bring the clicked word to the center and its related words around it. Figure 9 represents the Thesaurus structure for the word "information." This approach could be used to visualize related research papers [24].

The Fan Chart (shown in Fig. 10) is used in genealogy to display the ancestors of a person or in simpler words to display the family tree of the person. The outer concentric circle, which is the biggest, contains a crowded list of ancestors. The size of the circle grows with each generation of ancestors. Each ancestor is given the same space as every node in a concentric ring. An interactive variant of the fan charts was proposed by the authors in [20]. In this type of fan charts, clicking on a less important ancestor collapses the central node and removes the node's ancestors from the chart as well. The space freed by the collapsed node is taken over by the neighbor nodes.



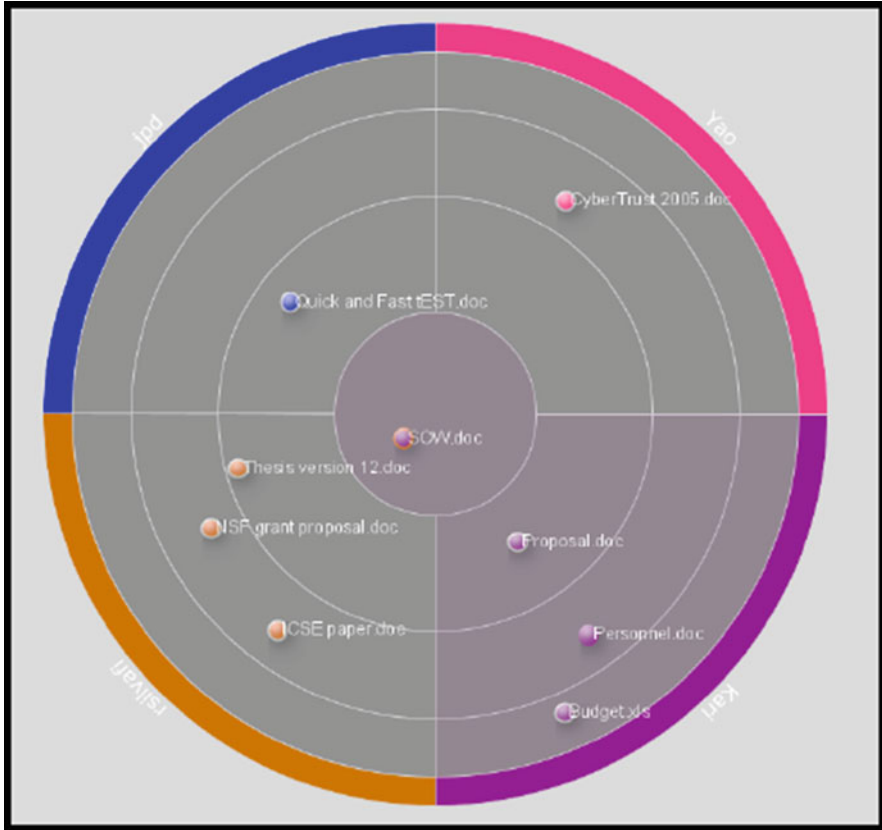


Fig. 7 Visualization interface in the impromptu system [17]

The authors in [21] also used a radial model to detect if a file is a malware by classifying all downloads into good downloads (not malware), bad downloads (malware), and ugly downloads (downloads that we are not sure whether they are good or bad). To analyze if the ugly download is either good or bad, the authors visualize the attributes of the download in a radial fashion. All downloads are plotted in axes inside the radial model, where each axis represents an attribute. The security administrator can then compare the similarities between previous downloads and the current “ugly” download. In Fig. 11, we can see the list of attributes of the download on the left and the radial model with the downloads visualized on the right. The green and red dots in the radial model refer to the good and bad downloads, respectively [21].

- (c) **Graph Layout:** Visualization of relationships and interdependencies (e.g., social media networks) and measures of an object or process (e.g., marks, height, accuracy) [11].

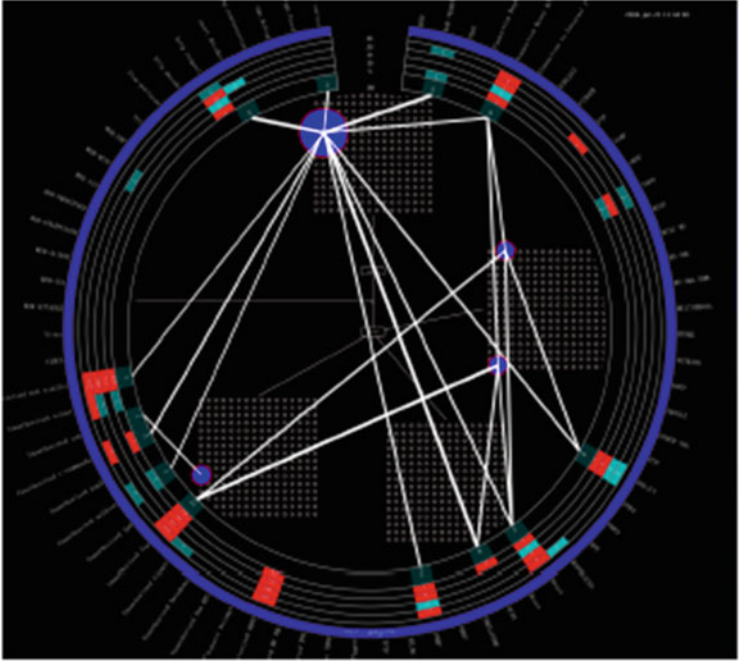


Fig. 8 The VisAlert System [18]

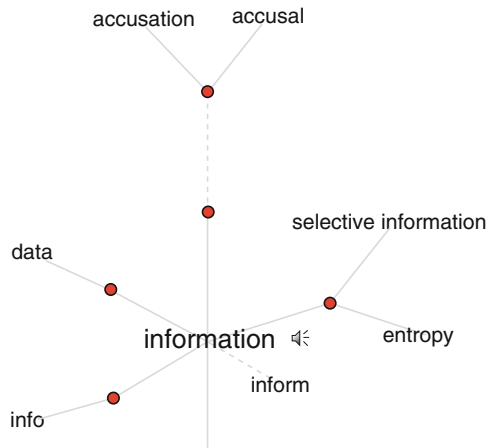


Fig. 9 Visual Thesaurus: Synonyms for the word “information” [19]

Graphs are mostly used to visualize relationships between a set of objects. The objects are represented as nodes and their relationships are represented using edges. A single graph can have multiple types of nodes to represent different sets of objects. Edges are of two types (Fig. 12):

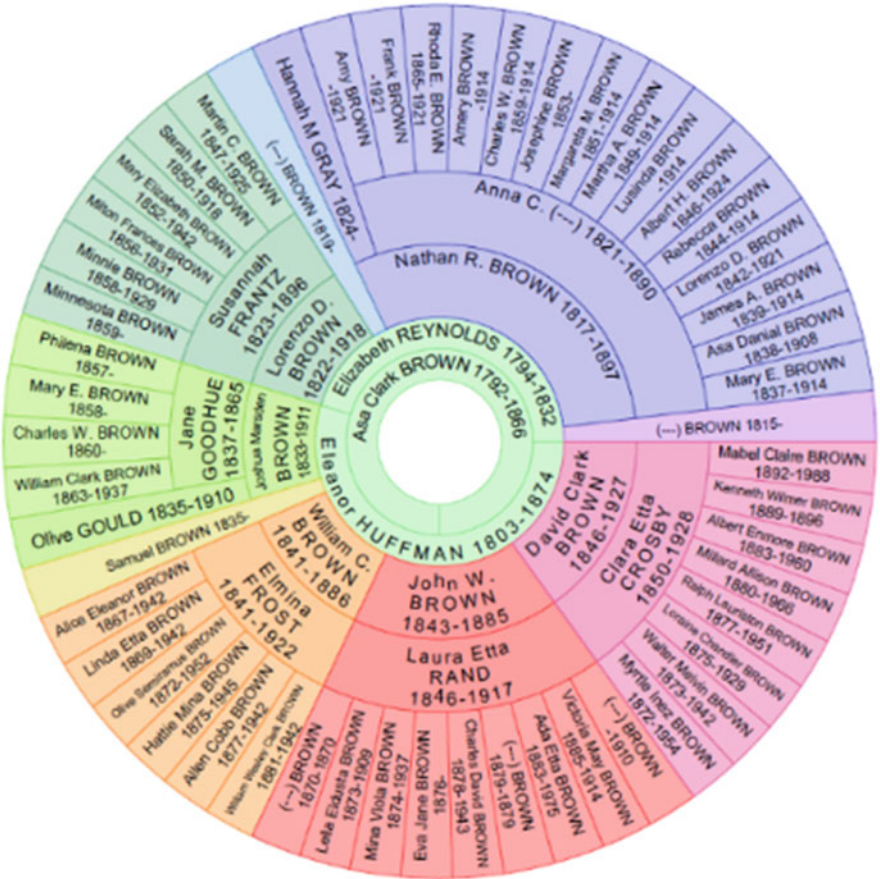
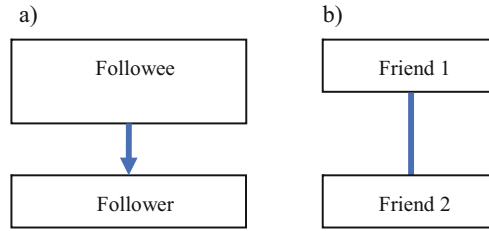


Fig. 10 Fan Chart [20]



Fig. 11 Visualization of attributes relating to the downloads [21]

**Fig. 12** (a) Directed graph;  
(b) Undirected graph



1. **Directed edge:** Indicates a one-way relationship, that is, the edge can be traversed in one direction only. For example, the relationship between followers and the person who is being followed.
2. **Undirected edge:** Indicates a two-way relationship, where the edges can be traversed both ways. For example, these types of edges are used for representing mutual friends in a social media network.

Edges can be manipulated to convey further information. The two common ways edges can be manipulated are:

1. **Weighted edge:** Indicates the strength of the relationship between two nodes. For example, the number of messages exchanged between people in a mutual friend network can be visualized by assigning a weight to it. The higher the number of messages, the thicker is the edge.
2. **Colored edge:** To represent different types of relationships, the edges are colored.

The authors in [22] discuss a bimodal graph (a graph that uses two types of nodes). The graph represents the relationship between community members (represented by circular nodes) and the forums (represented by rectangular nodes) they have posted to. The node size indicates the number of projects that were completed by each member and the weighted edge represents the number of posts that they posted to the forum.

The authors in [23] used a graph to represent the social media network of three Fortune 100s companies to illustrate how these companies are managing their social media platforms—Twitter, Facebook, blogs, and client-hosted forums. Figure 13 indicates how their social media and websites are interlinked.

- (d) **Layered Structures:** This layout is good for viewing different layers of an application, levels of security access, etc. This type of visualization is used in the Tudumi system (shown in Fig. 14) [16]. These methods use the visualization techniques described before, but use a set of these methods to form a layered structure to represent more information.

The Tudumi system visualizes user's activities, which are recorded in logs. These events are visualized using a layer of concentric disks, in which the lowest disk represents the user substitution information and the other disks represent other details like the log-in information. The system uses different shapes to identify the different types of users in the system.



Fig. 13 Graph representing how Walmart’s social media presence is linked [23]

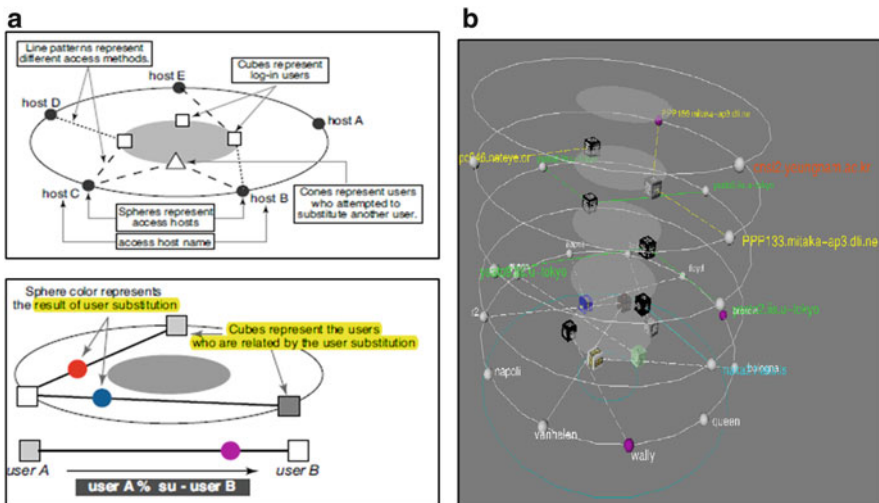


Fig. 14 The Tudumi system. (a) The Visualization paradigm that is followed to visualize events in each layer of security. (b) The visualized model that follows a layered structure [16]

A cube represents the logged-in users, spheres represent access hosts, and red cones represent users who attempted to substitute other users. Lines are drawn to represent the relation between access hosts and logged-in users. The color of the cones changes to show successful/unsuccessful substitution.

### 3.2 *Interactive Visualization Techniques*

Interactive visualization provides users with more control over the objects in the visualized scene. This helps the user to explore the data and to better understand the results of rigorous experiments. Such techniques not only boost cognition, they also increase user engagement. Users interact with the visualized scene for four main reasons [11]:

1. **Extract more details from an object (Drill-down):** The scene should provide an overview of the data. If the user notices an interesting pattern or object, then the user should be able to drill-down to access more details about the selected pattern or object.
2. **Show less information (Drill-up):** Once the interesting pattern or object has been explored in detail, the user should be able to drill-up to see the overview of the data.
3. **Modify the values of parameters:** The users should be able to modify the default values given to parameters by interacting with the scene.
4. **Customize the scene:** It leads to a much better experience for the user, as interaction is all about the user's preferences. The user should be able to change the color, shape of the objects, etc., in the scene.

Some of the interactive visualization techniques that will enhance the quality of information that is delivered to the user are given below. They are in accordance with Shneiderman's visualization task list [27].

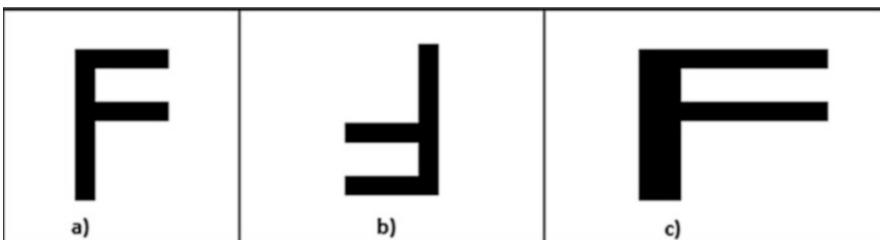
1. **Select:** When there are multiple items on the screen, it may be difficult for the user to keep track of items of potential interest. To overcome this, the user should be able to select an individual item or portions of the data that he or she finds interesting. This enables the user to identify the selected items easily that may be hidden among scores of other items. For example, the user might want to select the outliers in a pattern.
2. **Filter:** The system should allow for filtering data based on the user's choice. It is similar to "select" but the data here is filtered based on a condition as opposed to random selection of events. For example, the user may be interested in only viewing events that occurred during a specified time period [26].
3. **Zooming:** The user should be able to zoom into a pattern or a set of pixels in a dense pixel chart for more clarity. This helps in viewing the items from a low-level area, which provides more information when the patterns are observed from a closer view. The zoomed view can be provided as a pop-up or in the ancillary display to provide extra information regarding the selected pixels [25].

4. **Glossing:** The system should provide some basic details about the component when the cursor hovers on top of it. Some basic information like the time an event occurred, number of components, etc. could be displayed in the tooltip.
5. **Reconfiguration:** In some application, the user might want to focus on some principal event and view its relationship with other secondary events. In this case, the system should support reconfiguration to allow for changing the focus on different events [22]. Consider the example of the Visual Thesaurus, which is an interactive dictionary that displays the selected word in the center and its related words in a radial fashion around the selected word. When one of the related words is clicked, the entire arrangement shuffles around to bring the clicked word to the center and its related words around it. Figure 9 represents the thesaurus structure for the word “information.”
6. **Blinking:** The change in the status of any component, in the case of real-time visualization, should be communicated to the user by rapid flickering of the component to grab the attention of the user.
7. **Distortion:** This allows us to focus on a period of time, which provides a zoomed view of the selected period [26].

## 4 Visualization Options for 3D Scenes

Some of the following options are optional in a 2D setting; however, they are necessary for rendering a 3D scene.

1. **Perspective:** It refers to the angle at which the camera is set with respect to the object. Different perspectives can discover different dimensions of the object. Hence, an appropriate perspective that offers the best insight into the data should be selected.
2. **Panning:** Once the perspective is decided, that is, the camera is fixed at a location, the scene can be viewed from different angles. This helps in getting a better idea of the position and placement of all the objects in the scene by moving across different views [26].
3. **Rotation:** The object can be rotated about any axis to get complete information about the object. In Fig. 15(b), the letter “F” has been rotated by 180° along the y-axis.



**Fig. 15** Effects of rotation and scaling: (a) Normal view of letter “F” (b) 90° Rotated “F” (c) Horizontally scaled “F”



4. **Scaling:** The object can be scaled in size, along any of the axis. In Fig. 15(c) we can see that the letter “F” has been scaled to double its original size along the x-axis.
5. **Lighting:** The objects in 3D are often present in an environment that is “lighted.” This allows the user to view the surface of the object with an illusion of depth.

## 5 Popular Tools for Creating Visualization Models

In this section, we will look at some of the popular visualization tools that are easy to use and are properly documented. Table 1 shows some of the tools that help in visualizing the data in the form of charts or maps, whereas other tools allow the user to build their own visualization models.

Google Maps and Kartograph help in visualizing geographical data using maps. Tools like Google Charts, Fusion Charts, ZingCharts help in visualizing the data in the form of charts and the user can choose from the various predefined charts. Most of the tools provide highly interactive charts that are easy to work with. ZingCharts and Fusion Charts also help in creating dashboards that help to present the data to the user in a highly concise and compact manner.

Tableau is also used for chart modeling. It is easy to use mainly because of the ease with which you can plug in your data and choose from the different charts that are available. In addition, it provides an extremely easy drag-and-drop feature to select the visualization attributes.

Tools like Processing, D3, and Three.js are used to create their own visualization models in 2D/3D by coding in the supported languages. Some of these tools also provide various features that help in enhancing the interactivity of the created model. Fabric.js is another easy-to-use tool that is used for creating simple visualization models with more support for animation.

**Table 1** Popular visualization tools

Tool	Can create new visualization models?	Supports 3D?	Visualization type
Tableau	No	Yes	Charts
Processing	Yes	Yes	User-built model
D3	Yes	Yes	User-built model
Three.js	Yes	Yes	User-built model
Fabric.js	Yes	No	User-built model
Google Charts	No	Yes	Charts
ZingCharts	No	Yes	Charts
Kartograph	No	Yes	Maps



## 6 Conclusion

As data is growing at a rapid pace, it is important to visualize the data in a manner that is easy to use, navigate, and understand. It also lays the need for tracking multiple attributes of the data at every second, which leads to the need for creating dashboards that present concise and compact displays of the data. In this chapter, we presented a list of interactive and representative visualization techniques that enhance the quality of the visualization model that can be created by the user. We have also provided a simple visualization pipeline that outlines the steps to be followed to create a model that couples representative and interactive techniques. IV techniques that are specific to 3D visualization models were also presented.

We have also created a list of the popular visualization tools that are available today. A short comparison between some of these popular tools was also presented to help the user in selecting a tool that best suits his or her interests.

**Acknowledgments** This work was made possible by QUCP-CENG-CSE-15/16-2 grant from the Qatar University Fund. The statements made herein are solely the responsibility of the authors.

## References

1. Card, Stuart K., Jock D. Mackinlay, and Ben Shneiderman. *Readings in information visualization: using vision to think*. Morgan Kaufmann, 1999.
2. Chegini, M., Shao, L., Lehmann, D. J., Andrews, K., & Schreck, T. Interaction Concepts for Collaborative Visual Analysis of Scatterplots on Large Vertically-Mounted High-Resolution Multi-Touch Displays.
3. Zudilova-Seinstra, Elena, Tony Adriaansen, and Robert Van Liere. "Overview of Interactive Visualisation." *Trends in Interactive Visualization*. Springer London, 2009. 3–15.
4. Brodbeck, Dominique, Riccardo Mazza, and Denis Lalanne. "Interactive visualization-A survey." *Human machine interaction*. Springer Berlin Heidelberg, 2009. 27–46.
5. Hicks, M. (2009). Perceptual and design principles for effective interactive visualisations. In *Trends in Interactive Visualization* (pp. 155–174). Springer London.
6. Pham, B., Streit, A., & Brown, R. (2009). Visualisation of Information Uncertainty: Progress and Challenges. In *Trends in interactive visualization* (pp. 19–48). Springer London.
7. Vassileva, Julita, and Carl Gutwin. "Exploring blog archives with interactive visualization." *Proceedings of the working conference on Advanced visual interfaces*. ACM, 2008.
8. Chi, E. H. H., & Riedl, J. T. (1998, October). An operator interaction framework for visualization systems. In *Information Visualization, 1998. Proceedings. IEEE Symposium on* (pp. 63–70). IEEE.
9. Heer, Jeffrey, Stuart K. Card, and James A. Landay. "Prefuse: a toolkit for interactive information visualization." *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2005.
10. Tuttle, C., Nonato, L. G., & Silva, C. (2010). PedVis: a structured, space-efficient technique for pedigree visualization. *IEEE transactions on visualization and computer graphics*, 16(6), 1063–1072.
11. Keim, Daniel A. "Information visualization and visual data mining." *IEEE transactions on Visualization and Computer Graphics* 8.1 (2002): 1–8.

12. Wu, Yingcai, et al. "OpinionSeer: interactive visualization of hotel customer feedback." *IEEE transactions on visualization and computer graphics* 16.6 (2010): 1109–1118.
13. Nguyen, Quang Vinh, and Mao Lin Huang. "EncCon: an approach to constructing interactive visualization of large hierarchical data." *Information Visualization* 4.1 (2005): 1–21.
14. Fernandez, Rachael, and Noora Fetais. "Framework for Visualizing Browsing Patterns Captured in Computer Logs Using Data Mining Techniques." *International Journal of Computing & Information Sciences* 12.1 (2016): 83.
15. Van Ham, F., & van Wijk, J. J. (2003). Beamtrees: Compact visualization of large hierarchies. *Information Visualization*, 2(1), 31–39.
16. Takada, Tetsuji, and Hideki Koike. "Tudumi: Information visualization system for monitoring and auditing computer logs." *Information Visualisation, 2002. Proceedings. Sixth International Conference on. IEEE, 2002.*
17. De Paula, Rogerio, et al. "In the eye of the beholder: a visualization-based approach to information system security." *International Journal of Human-Computer Studies* 63.1 (2005): 5–24.
18. Livnat, Y., Agutter, J., Moon, S., Erbacher, R. F., & Foresti, S. (2005, June). A visualization paradigm for network intrusion detection. In *Information Assurance Workshop, 2005. IAW'05. Proceedings from the Sixth Annual IEEE SMC (pp. 92–99). IEEE.*
19. Thinkmap Inc., "Thinkmap Visual Thesaurus," <http://www.visualthesaurus.com>
20. Draper, G. M., & Riesenfeld, R. F. (2008). Interactive fan charts: A space-saving technique for genealogical graph exploration. In *Proceedings of the 8th Annual Workshop on Technology for Family History and Genealogical Research (FHTW 2008).*
21. Angelini, M., Aniello, L., Lenti, S., Santucci, G., & Ucci, D. (2017, October). The goods, the bads and the uglies: Supporting decisions in malware detection through visual analytics. In *Visualization for Cyber Security (VizSec), 2017 IEEE Symposium on (pp. 1–8). IEEE.*
22. Hansen, D. L., Rotman, D., Bonsignore, E., Milic-Frayling, N., Rodrigues, E. M., Smith, M., & Shneiderman, B. (2012, December). Do You Know the Way to SNA?: A process model for analyzing and visualizing social media network data. In *Social Informatics (SocialInformatics), 2012 International Conference on (pp. 304–313). IEEE.*
23. Culnan, M. J., McHugh, P. J., & Zubillaga, J. I. (2010). How large US companies can use Twitter and other social media to gain business value. *MIS Quarterly Executive*, 9(4).
24. Yi, Ji Soo, Youn ah Kang, and John Stasko. "Toward a deeper understanding of the role of interaction in information visualization." *IEEE transactions on visualization and computer graphics* 13.6 (2007): 1224–1231.
25. Carlis, John V., and Joseph A. Konstan. "Interactive visualization of serial periodic data." *Proceedings of the 11th annual ACM symposium on User interface software and technology. ACM, 1998.*
26. Bade, Ragnar, Stefan Schlechtweg, and Silvia Miksch. "Connecting time-oriented data and information to a coherent interactive visualization." *Proceedings of the SIGCHI conference on Human factors in computing systems. ACM, 2004.*
27. Shneiderman, Ben. "The eyes have it: A task by data type taxonomy for information visualizations." *Visual Languages, 1996. Proceedings., IEEE Symposium on. IEEE, 1996.*