Case Study: Potential Vulnerabilities in Information Visualization

Niusen Chen
Depart of Computer Science
Michigan Technological University

Abstract

Information visualization is widely used nowadays. The visualized information are more convenient for people to understand the nature of data. People benefit a lot from several information visualization software such Google Maps and Zoho Reports. However, with the large amount of data posted to the public, potential attacks may also target on those information visualization system. In this paper, we use a real world attack on Google Maps as a case study to discuss a potential vulnerability. Then, we propose two preliminary solutions to mitigate this potential attack. We finally discuss the problems and challenges in the future work.

1 Introduction

With the development of Internet, more and more people prefer to uploading their data to web and sharing them with others. Because of the vast amounts of data posted to the public, information visualization becomes a popular way for companies or users to display data. Information visualization takes advantage of the highest bandwidth human input device, vision, and human perceptual capabilities. Information visualization can be used for exploration, discovery, decision making, and to communicate complex ideas to others [1]. For example, a boxplot is a kind of chart which can visualize the distribution of data. Treemap is a type of chart which is used to display the contents in different categories. 3D Scatter Plot is often used to visualize the relationship between different variables [2]. Recently, some visualization techniques and tools are proposed. Voyager (K. Wongsuphasawat, et al., 2016), a mixed-initiative system that supports faceted browsing of recommended charts chosen according to statistical and perceptual measures [3]. LineUP (S. Gratzl, et al., 2013), a novel and scalable visualization technique that uses bar charts. This interactive technique supports the ranking of items based on multiple heterogeneous attributes with different scales and semantics [4]. Information visualization will be more frequently used in the future.

However, information visualization system are also suffered from potential attacks. Since data are explored to public, there is a possibility that some private data may be leaked to attackers unintentionally. Hackers can also attack an information visualization system through uploading some fake data to confuse other users. Those potential vulnerabilities should be considered when company starts to design an information visualization system. In the remaining part of this paper, we first introduce some basic knowledge about information visualization. Then a concrete a attacking

example is analyzed to show the vulnerabilities in an information visualization system. Finally, some suggestions are given to mitigate the potential attacks.

2 Background

2.1 Data

Information visualization starts from data [5]. Data is a high level and abstract concept. A digit number is data, an image is data, a text message is also data. Data can be classified as continuous data and discrete data. Discrete data can only take certain values, but continuous data can take any value. In computer field, all data is discrete since computer can only deal with digits. Data will be transformed to digits before they can be processed by computer. In real world, some additional equipment are needed to do translation. For example, voltage or current are translated to digit value through a analog to digit converter after they are collected, image is represented as pixels in computer, etc. But actually, they are all binary values when processed by CPU.

2.2 Data Visualization

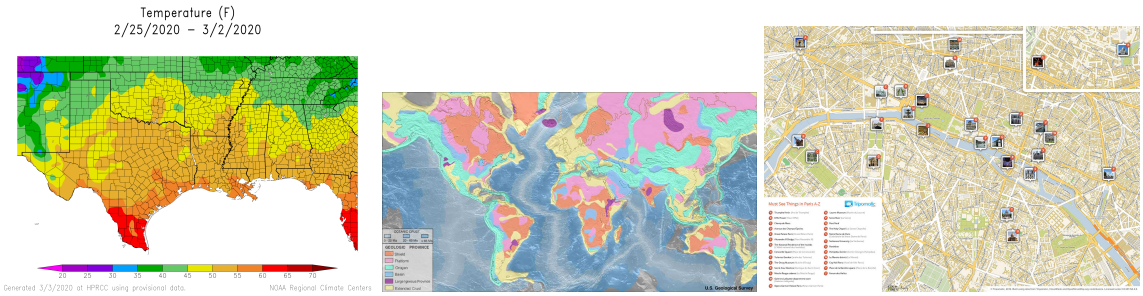
Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends and patterns in data [6]. The process of data visualization consists of 5 steps. First step is **Mapping**, which means how to encode data into a visual form. Second step is **Selection**. Selection can treated as a filter, which removes useless data. The third step is **Presentation**. Presentation requires managing and organizing data on the screen effectively. The fourth step is **Interactivity**, which requires a user-friendly visualization system. The last step is **Evaluation**. Evaluation is an important part since it can detect some potential bugs or vulnerabilities of the visualization system [7].

2.3 Charts

In this part, some frequently used simple charts are introduced, such as line chart, bar chart and pie chart. Then, other visualization system such as maps are introduced.

2.3.1 Simple Charts

The most and the easiest way to visualize data is charts. There are different types of charts, such as pies, bars, line charts and 3D charts. Most of these charts are used to display one dimension data, except 3D chart. Pie charts are best to used to compare the proportions or percentage between different values. It is straightforward for people to observe the percentage distribution. But the drawback is, it can only represent a small number of different values. If there are too much values, the pie chart will be messy. Bar charts are often used to show how frequently each value occurs. People can directly get the frequency of each value by looking at the height of the bar charts [8]. Bar charts can better clarify trends than tables and estimate estimate key values at a glance. But bar charts require additional written or explanation [9]. Line charts can be used to show the trend



(a) (b) (c)

Figure 1: Different Types of Maps.

of discrete or continuous values. It also can be used to show the frequency of different values. Line chart can contain multiple lines in one graph, but too many lines may also confuse readers. 3D bar charts can be seen as an extension of bar charts. 3D bar charts can represent two categorical data dimensions simultaneously. But 3D bar charts are not suggested to be used frequently, occlusion is one of the problems [8]. Some bars in the front may hide the bars in the back, which is difficult for users to read. Figure 2 gives 4 examples of different simple charts. (a) is a pie chart, it displays the proportions between rice, milk and bread. (b) is a bar chart, it shows the frequency of different values. (c) is a line chart, it illustrates the temperature trend form Monday to Friday. (d) [10] is a 3D bar chart. It shows the number sold of different fruits from April to June. Besides, there are some variations of simple charts, such as stacked charts, histogram and scatter plots. All of them add some additional components or make some changes on the original simple charts in order to fit the different situations. Details about those charts will not be discussed here.

2.3.2 Maps

Maps are used for a very long time. Map is a symbolic depiction emphasizing relationships between elements of some space, such as objects, regions, or themes [11]. In ancient times, maps were static and were made of paper mostly. Nowadays, with the development of technology, maps already become dynamic and interactive. Google Maps is a dynamic map which is widely used nowadays. In addition to conventional road maps, it also offers aerial and satellite views of many places. Information visualization is part of Google Maps. The technique stack to support Google Maps involves database, network, machine learning, etc. Some other kinds of maps such as climatic map, geologic map, resource map and star map are also have an application in their own field. Figure 1 lists 3 different maps. (a) is a climate map. (b) is geologic map and (c) is a street map.

2.3.3 Treemaps

Treemaps organize data in a tree-structured way. Each branch of the tree is given a rectangle, which is then tiled with smaller rectangles representing sub-branches [12]. Leaf nodes are ones that have no children. Root node locates on the top of the tree. Treemaps are useful when comparing nodes and subtrees at varying depths in the tree to find patterns and exceptions.

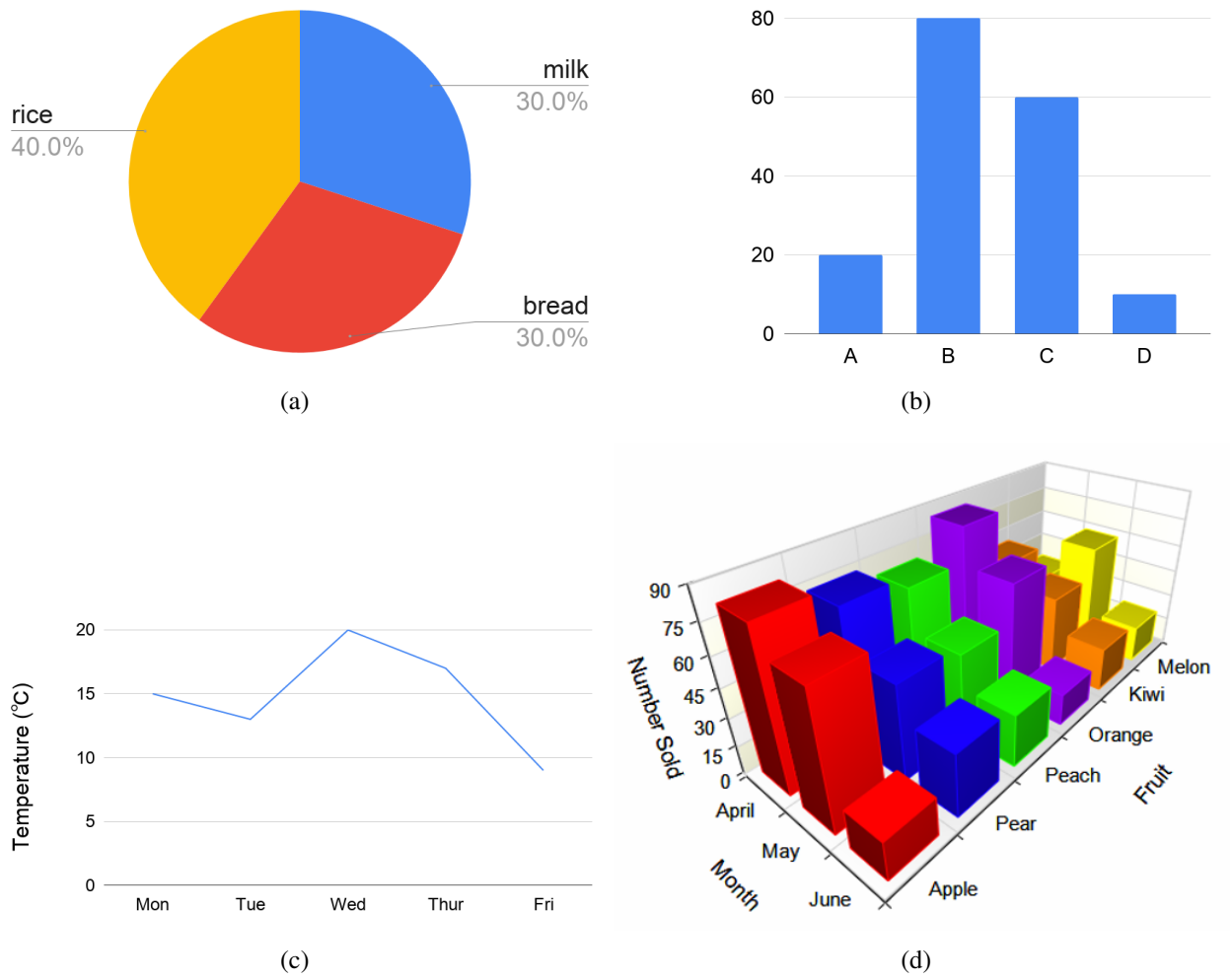


Figure 2: Different Types of Charts.

2.3.4 Others

There are some other ways to represent data. Parallel coordinates can be used to visualize many dimensions in one graph. Link graphs is best for visualizing relationship and communication patterns. Three-Dimensional graphs can provide more angels for the viewers, thus viewers are able to get more details of the data.

3 Attacking Google Map

In this part, we introduce a real world attacking example on Google Map. A German artist illustrated how to forge a traffic jam in Google Map by walking around the street with 99 cell phones.

3.1 System Model

Google Maps uses GPS in cell phones to determine whether the traffic is crowded or not. The traffic situation is visualized in Google Map with different colors. An example of the traffic situation in Chicago can be seen in Figure 3. Red color on the road indicates the traffic is slow and green indicates fast. Based on the traffic maps, drivers can schedule their routes to avoid traffic jam.

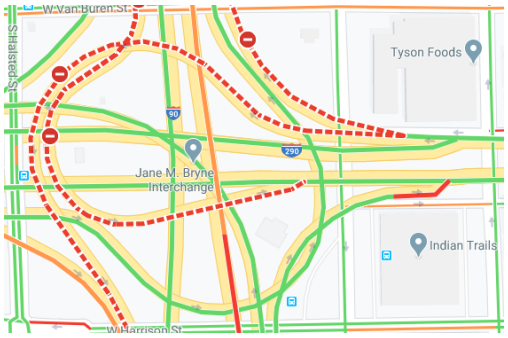


Figure 3: Traffic Situation in Chicago.

3.2 Attack Scenario

An artist from Germany attacked the google map by forging some fake traffic jam. This artist rented 99 android phones, then he opened Google Maps in each cell phone. He walked on a street back and forth. This street turn to red after while, which indicates this street is busy. But actually, this street is not as busy as people see in Google Maps.

4 Preliminary Solutions

In this part, we propose two preliminary solutions to mitigate this attack on Google Maps. Due to the limitations, these two solutions are not experimentally verified yet.

4.1 Camera-based Traffic Flow Detection

The first preliminary solution is to use cameras to detect traffic flow. The camera can calculate the number of vehicles in a street. Then, data collected from cameras will be uploaded to the Google Maps. Those data will be visualized as instant traffic situation in user's app. The system architecture is shown in Figure 4.

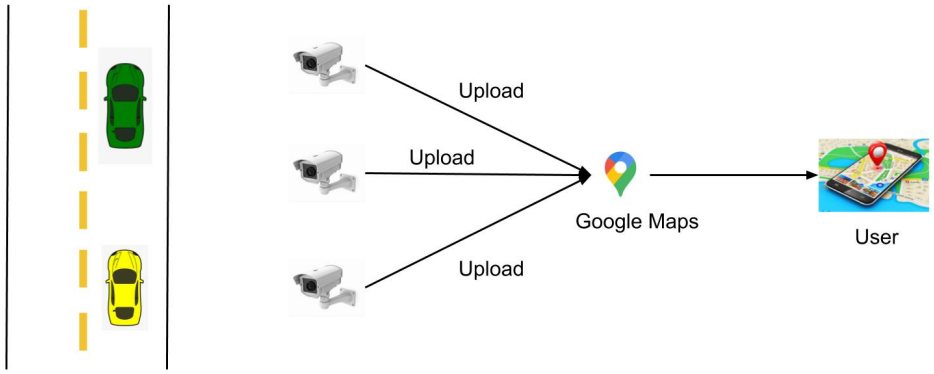


Figure 4: Camera-based Traffic Flow Detection.

In the figure above, each camera can cover partial street. Periodically, each camera will calculate the number of vehicles in its own range, then uploads data to the server. Google Maps server will visualize those information and publish to the user. We use a simple model to explain the details about this system. A simplest model of this system can be described as in Figure 5.

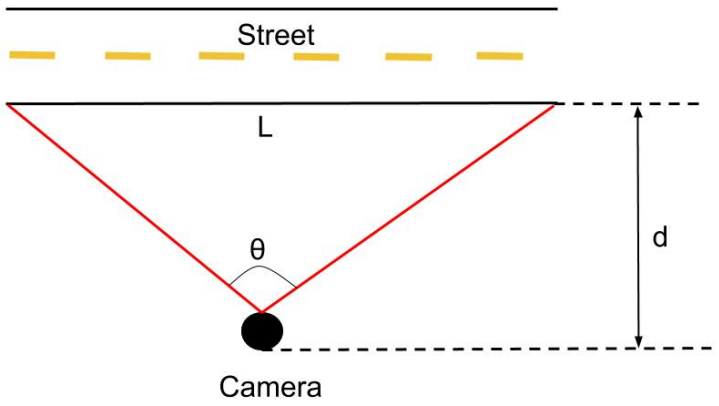


Figure 5: A Simple Model.

In the image above, L is the length which a single camera can catch. d is the distance between the camera and the street. θ represents the max angle a camera can cover. Therefore, length L can be calculated by following equation:

$$L = 2d \tan \frac{\theta}{2} \tag{1}$$

Suppose the road is straight and the total length is S, the total number of cameras needed for this road is n, therefore, we can get the following equation:

$$nL \geq S \tag{2}$$

We combine equation (1) and equation (2), we can get the number of cameras n needed for the street:

$$n \geq \frac{S}{2d \tan \frac{\theta}{2}} \tag{3}$$

There are several issues need to be addressed. First, previous analysis is a simple model of the system. We assume the road is straight. However, in real world, it is not possible that roads/streets are always straight. For those complicated road situations, equations above can not be applied. Second, above analysis only provides a method to calculate the number of cameras needed, another significant problem is how to detect the vehicles. There are several moving object detection techniques, such as *Point Tracking*, *Kernel Tracking* and *Silhouette Tracking* [13]. Details about those algorithms will not be discussed here. But different from the traditional object tracking, corner cases should be considered carefully. For example, if a vehicle locates in the middle of two cameras, each camera only detects half of the vehicle, as shown in Figure 6. In this situation, only one camera should count it. Third, how often should the cameras upload the data. It is better if cameras can upload data instantly. However, this will waste a lot of resources and unnecessary. A potential strategy could be, the frequency of uploading data should be dynamic instead of static. For example, the frequency should be high in the morning, since at that time, people start to go to work. Informing them of the traffic situation instantly will save their time. But in the evening, the frequency should be low, since most people stay home in the evening. This is an optimization problem. More details need to be discussed.

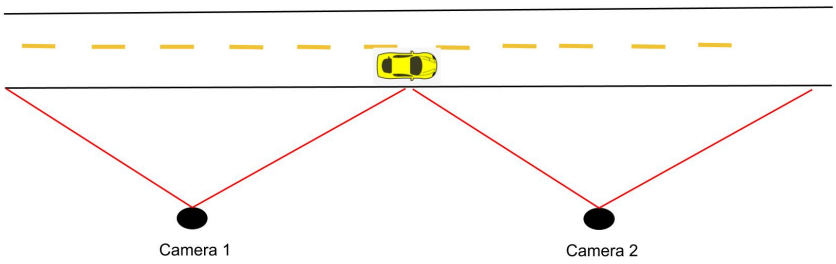


Figure 6: A Corner Case.

4.2 Vehicle-GPS based Traffic Flow Detection

The other preliminary solution can be embedding GPS into vehicles. In this way, the attack described in Section 3 is almost impossible to occur. But how to integrate the Google Maps service into vehicle's system becomes the new challenge. GM and Volvo have already considered embedding Google Maps into their next generation vehicles [14, 15].

4.3 Discussion

Compare to the existing Google Maps, the solutions we proposed can mitigate the attack in Section 3. But for Camera-based Flow Detection, how to cover as much of the area as possible with a minimum number of camera is a challenging problem. Otherwise, a huge amount of cameras will

be installed. For Vehicle-GPS based Traffic Flow Detection, the most difficult work is to embed Google Maps service into vehicle's control system. Since vehicle's control system is a simple embedded system, resources are limited. Integrating Google Maps will bring additional overhead to the system. Another problem about Vehicle-GPS based Traffic Flow Detection is the security of the car itself. Autonomous driving thrives rapidly, it relies on V2X technologies. But what if autonomous driving system is compromised by attacker. In this situation, attacker can modify any data he/she wants. System security and network security also should be taken into consideration.

5 Conclusion

In this paper, we use a real world example to illustrate the potential vulnerabilities in Google Maps. Then we propose two preliminary solutions to mitigate the attack. Due to some limitations, they are not verified experimentally. We also list the potential problems and challenges of these two proposed solutions. The problems and details should be clarified in the future work. Based on this case study, we believe there are still a lot of potential vulnerabilities in information visualization system. Designers and users should pay more attention to the information security.

References

[1] J. R. Goodall, "Introduction to visualization for computer security," in *VizSEC 2007*. Springer, 2008, pp. 1–17.

[2] O. Velarde, "19 innovative ways to use information visualization across a variety of fields," https://visme.co/blog/information-visualization/, accessed March 1, 2020.

[3] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer, "Voyager: Exploratory analysis via faceted browsing of visualization recommendations," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 1, pp. 649–658, 2015.

[4] S. Gratzl, A. Lex, N. Gehlenborg, H. Pfister, and M. Streit, "Lineup: Visual analysis of multi-attribute rankings," *IEEE transactions on visualization and computer graphics*, vol. 19, no. 12, pp. 2277–2286, 2013.

[5] S. K. Card and J. Mackinlay, "The structure of the information visualization design space," in *Proceedings of VIZ'97: Visualization Conference, Information Visualization Symposium and Parallel Rendering Symposium*. IEEE, 1997, pp. 92–99.

[6] https://www.tableau.com/learn/articles/data-visualization.

[7] M. Khan and S. S. Khan, "Data and information visualization methods, and interactive mechanisms: A survey," *International Journal of Computer Applications*, vol. 34, no. 1, pp. 1–14, 2011.

[8] R. Marty, *Applied security visualization*. Addison-Wesley Upper Saddle River, 2009.

[9] "Advantages and disadvantages of different types of graphs," http://www.kmrom.com/Site-En/Articles/ViewArticle.aspx?ArticleID=416, accessed March 2, 2020.

[10] "Ncss plots and graphs," https://www.ncss.com/software/ncss/ncss-plots-and-graphs/, accessed March 2, 2020.

[11] https://en.wikipedia.org/wiki/Map.

[12] https://en.wikipedia.org/wiki/Treemapping.

[13] K. A. Joshi and D. G. Thakore, "A survey on moving object detection and tracking in video surveillance system," *International Journal of Soft Computing and Engineering*, vol. 2, no. 3, pp. 44–48, 2012.

[14] https://www.media.volvocars.com/global/en-gb/media/pressreleases/228639/volvo-cars-to-embed-google-assistant-google-play-store-and-google-maps-in-next-generation-infotainme.

[15] https://www.zdnet.com/article/gm-wants-to-put-google-maps-google-assistant-in-your-car/.