Case Study: Potential Vulnerabilities in Information Visualization

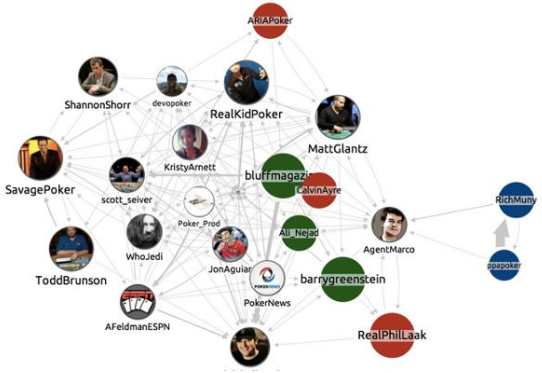
Niusen Chen (PhD Student)
Department of Computer Science
Michigan Technological University

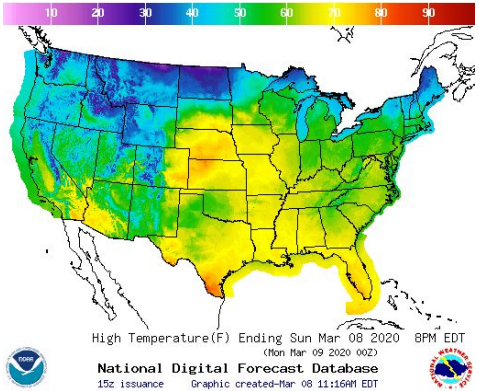
Outline

- Background
- Process of Visualization
- Case Study: Vulnerabilities in Google Maps
- Conclusions

Information Visualization

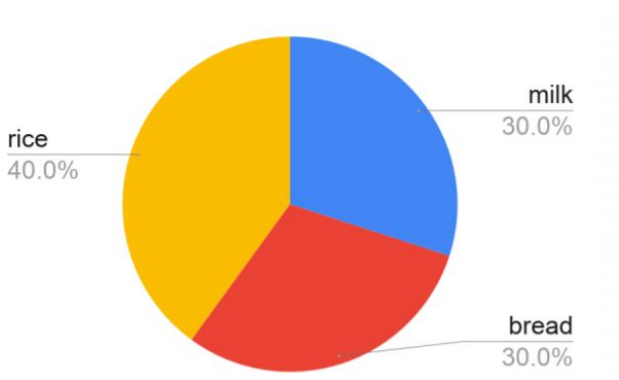
Information visualization is the study of (interactive) visual representations of abstract data to reinforce human cognition. [1].



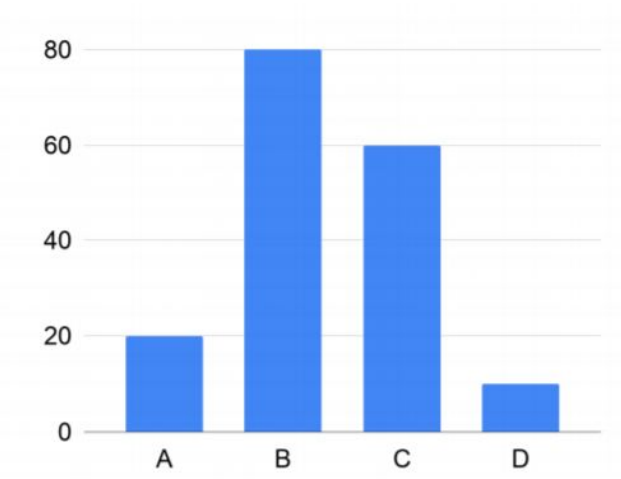


Simple Charts

Pie chart

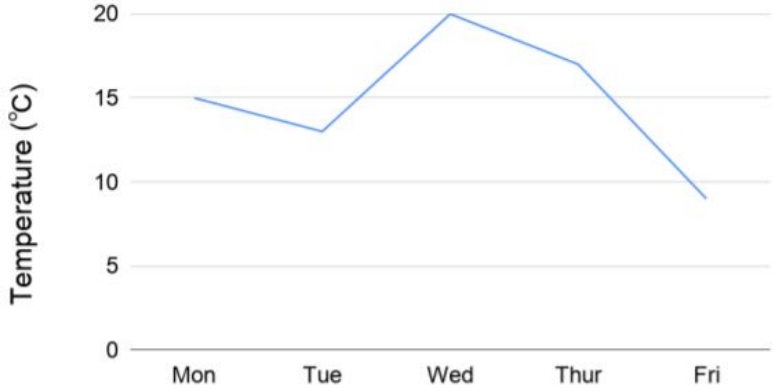


Bar chart

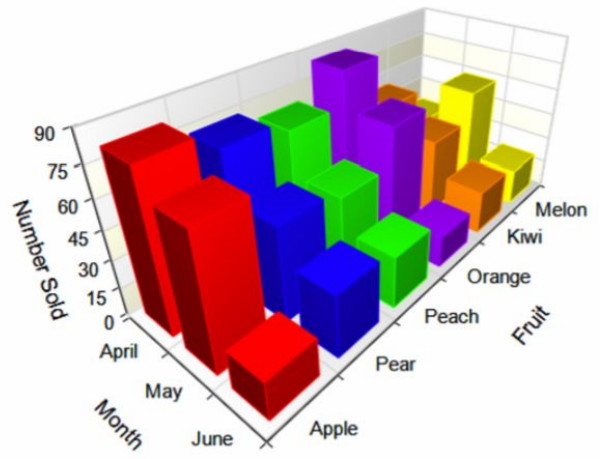


Simple Chart (cont.)

Line chart



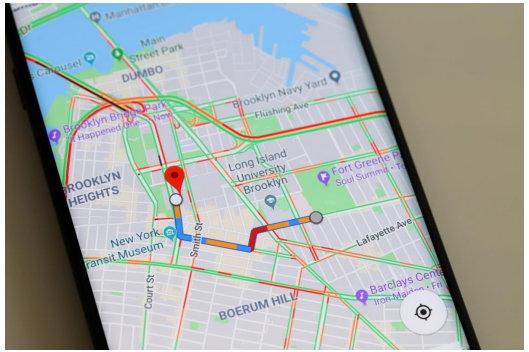
3D-Bar chart



Maps

A map is a symbolic depiction emphasizing relationships between elements of some space, such as objects, regions, or themes.

- Static Maps
- Dynamic Maps



Others

- Treemaps
- Link-graphs
- Three-dimensional graphs

Process of Visualization

Mapping: Encode data into a visual form

Selection: Remove useless data

Presentation: Manage and organize data on the screen effectively

Interactivity: Design an user-friendly visualization system

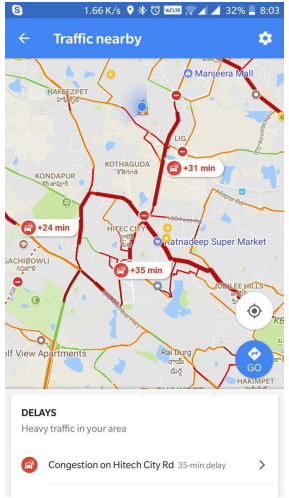
Evaluation: Detect potential bugs or vulnerabilities [2]

Case Study: Vulnerabilities in Google Maps

Google Maps: A web and mobile application. It offers satellite imagery, street maps, 360° panoramic views of streets (Street View), real-time traffic conditions (Google Traffic), and route planning for traveling by foot, car, bicycle and air (in beta), or public transportation. Systems such as Global Positioning System (GPS) and Geographical Information System (GIS) are used in Google Maps.

Case Study: Vulnerabilities in Google Maps

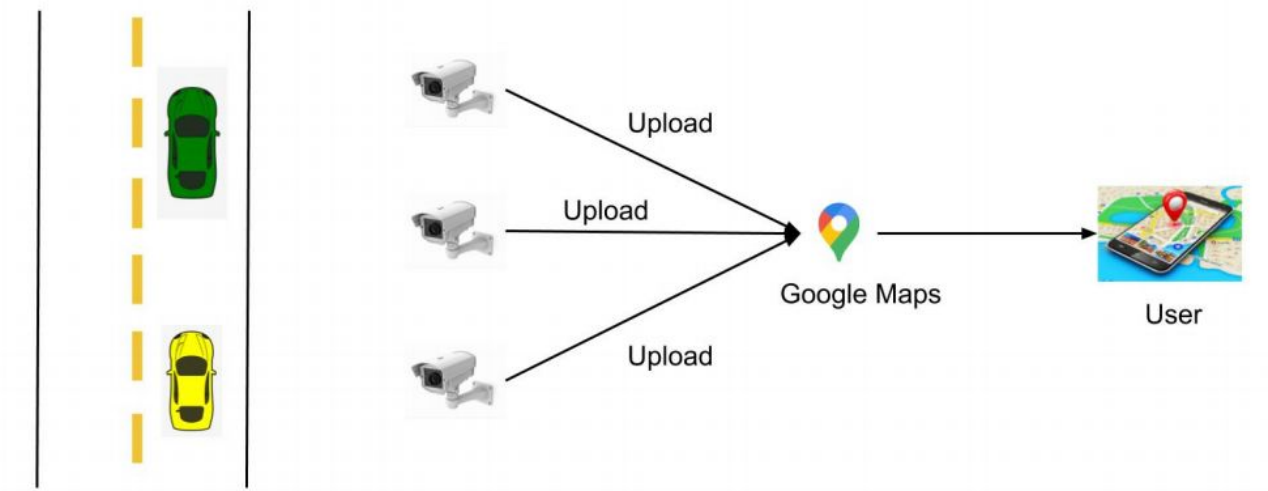
Attacks: A German artist illustrated how to forge a traffic jam in Google Map by walking around the street with 99 cell phones



https://www.youtube.com/watch?v=k5eL_al_m7Q

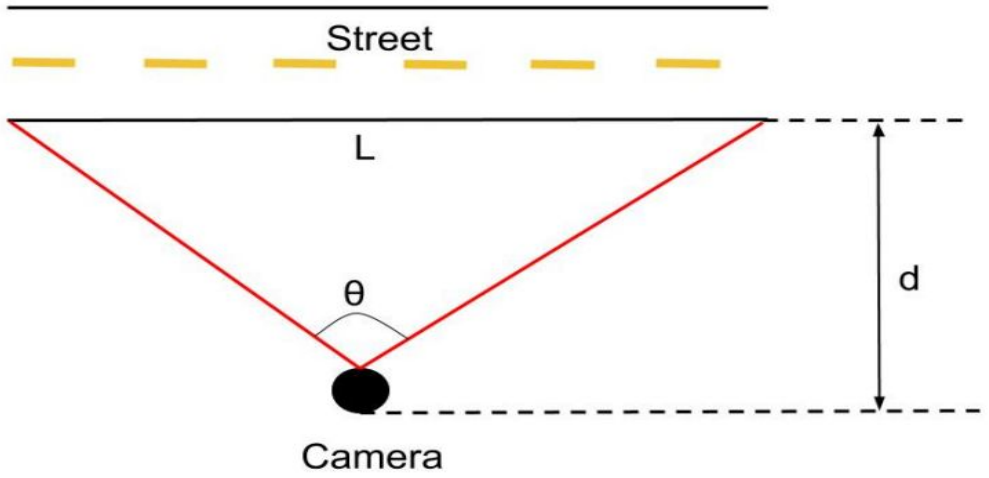
Preliminary Solution

Solution 1: Camera-based Traffic Flow Detection



Preliminary Solution (cont.)

Solution 1: Camera-based Traffic Flow Detection



S: Total length of street

L: Length which one camera can cover

d: Distance between camera and street

n: Number of cameras required for the street

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

Preliminary Solution (cont.)

Solution 1: Camera-based Traffic Flow Detection

- How to handle more complicated road situations?
  - Curves, Intersections, etc.
- How to detect vehicles
  - Point Tracking, Kernel Tracking and Silhouette Tracking [3]
- Update frequency
  - Static
  - Dynamic
    - Morning: frequently update
    - Evening: Infrequently update

Preliminary Solution (cont.)

Solution 2: Vehicle-GPS based Traffic Flow Detection

Embed Google Maps into vehicle’s system

Preliminary Solution (cont.)

Solution 2: Vehicle-GPS based Traffic Flow Detection

- Bring additional overhead to vehicle control system
  - Resources are limited
- Require vehicle companies to update their systems
  - Expensive and time consuming
- Security of autonomous driving

Conclusion

- Information Visualization system could be compromised
- Future research should focus on this area

References

[1] https://en.wikipedia.org/wiki/Information_visualization

[2] 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.

[3] 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