Project Title: Visualizing Large-Scale Binary Data Sets

Team Members: Kwuang Tang, Kevin Jin

Proj repo: <https://github.com/cktang88/viz-proj>

Paper: <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6876026>

**Background and Motivation:** What are your motivations for choosing this project? Is the project related to your research interests, or something we covered in the lectures?

Visualizing data with high cardinality in sets and elements per set using traditional visualization techniques can be difficult, since Venn Diagrams and Euler Diagrams and other visualizations quickly become overly crowded and complicated. For the OnSet data visualization method, this is not the case because there can be a very large number of sets and also large numbers of elements in each set in the dataset and the resulting visualization would still be comprehensible. This is because each set is represented independently through a PixelLayer, which can in turn hold a large quantity of elements. Usually a user would only want to compare several sets at once to find relationships, which is to facilitate through interactive operations and the MultiLayer, which can either be an OR operation or AND operation on multiple PixelLayers.

The OnSet visualization uses the spatial visual channel quite uniquely compared to other types of visualizations as well. Rather than having the X and Y axis represent some specific quantitative or categorical attribute, the coordinates of each pixel locations are algorithmically generated to group hierarchies together.

Not only is the OnSet paper very effective in scaling data elements by size and number of categorical attributes, the OnSet paper uses its visual channels in very different ways compared to the basic visualization techniques we’ve seen in lecture. The OnSet visualization technique relies heavily on interactive operations, which is common to modern day visualization techniques.

We’re interested in choosing this project because the OnSet visualization technique seems to be a very effective way to scale visualization for large numbers of data elements and categories. We also believe the unique usage of visual channels and modern-day interactivity relied upon in this visualization would be educational to us by broadening our perspectives with regards to creating visualization techniques.

**Objectives:** What do you hope to achieve in the project? What do you intend to learn as part of the outcome?

With regards to this project, we plan to effectively implement the OnSet visualization technique to a degree to which the research paper authors would agree that the original goal was satisfied. The OnSet visualization technique we implement does not have to be exactly as in the research paper due to our time constraints of 4 weeks for this project, but the visualization should scale well for thousands of data points and many categories/sets, to see the relationship between categories/sets.

What we intend to learn as part of the outcome is how to create a visualization that scales for many categories. Because in real world datasets, an element in the data will very often belong to many categories and there would be likewise, many categories to a single element. Hopefully through the process of implementing the OnSet visualization technique, we can get ideas for a new visualization technique that would also scale well for many categories.

**Data:** From where and how are you collecting data? If appropriate, provide a link to your data sources.

In the paper, two case studies were presented - one with whale shark blood data, and the other is a calendar visualization. The data we plan to use will be similar in format to these in order to fairly utilize the technique as it is intended to be applied to target data. Note that many data sets can be transformed to binary data sets by simply asking specific yes/no questions about presence of certain attributes.

Good source of binary data sets: <https://dtai.cs.kuleuven.be/CP4IM/datasets/>

1. Animals: <https://dtai.cs.kuleuven.be/CP4IM/datasets/data/zoo-1.txt>

PixelLayer - one per attribute (eg. hair, etc.)

Element - represents a single animal, is it a mammal AND has attribute (yes/no)

2. Voting: <https://dtai.cs.kuleuven.be/CP4IM/datasets/data/vote.txt>

PixelLayer - one per vote issue (eg. immigration)

Element - represents a single congressperson, did they vote “yes” AND they are a democrat (yes/no)

There are probably other data sets we could explore if these two turn out to be bad candidates.

**Data Processing:** How much do you intend to devote to typical data processing tasks, such as data cleanup, data aggregation, etc.. How much time do you intend to devote to this?

Two or three days have been set aside to do data cleanup and aggregation, in order to display the basic data set as a matrix (PixelLayer) by the end of the first week. Data processing is pretty easy with the OnSet technique - all that is needed is data that is put in a nested array (one array per set). The placement of cells in the array is simply computed via modulos. As an optional feature, if hierarchy is supported keys may need to be used.

In the data above, the `.txt` files will be converted to `.csv` via a tool, which will be read into the program and be converted to arrays as in previous class projects. This is not expected to be overly difficult.

**Must-Have Features:** List the features that are absolutely necessary for the project to be successful.

* Display each set of binary data as a colored matrix (PixelLayer)
* Highlight corresponding pixel in other Layers of pixel being hovered over
* Drag and Drop for “and” and “or” operations (creates a MultiLayer)
* Each Layer needs a label

**Optional Features:** List the features that you think would be nice to have, but not critical.

* Group by hierarchy
* On highlight pixel, show hierarchy group that it belongs to
* Ability to add new data, to create new PixelLayers
* Thickness band - band between two layers that show similarity between the layers

**Project Schedule:** Plan out the schedule for the project, on a weekly basis. Show how work will be delegated amongst the team members.

* **Proposal:** November 7
* Week 1: collect one or several data sets to work with, and display them each as sets (Pixel Layers)
* **Update 1:** November 14
* Week 2: Highlight corresponding pixels in other layers + Add labels for each layer + Write up Process Book PDF
* **Prototype Submission:** November 21
* Week 3: Add ability to drag/drop for and/or operations to create MultiLayer
* **Update 2:** November 28
* Week 4: Tackle one optional feature + make project presentation + documentation (README)
* **Presentation, Final Project Submission:** December 7

### 

**Visualization Technique:** Provide a link to the paper you are going to implement, and a detailed description of how you plan on implementing it. What data analysis is necessary? How are you going to visually encode the data? If you are only going to consider parts of the paper, please specify which parts. If you plan on extending certain aspects of the paper, using other techniques, please include these details.

Paper: <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6876026>

We will implement the OnSet visualization technique by tackling core features one step at a time. Of course the first task at hand would be to create a PixelLayer from a dataset. This PixelLayer would have pixels arranged initially according to the order they were inputted into the dataset (later on we would have pixel locations be algorithmically computed to account for hierarchies in data, elements within the same hierarchy would be clustered together). We will probably implement this using d3js’ scaleBand, another thing to look out for is keeping aspect ratio of the PixelLayer and each element of the matrix to be as close to each other as possible. The code to create a PixelLayer would be wrapped in a javascript function, called createPixelLayer.

The next step would be to create multiple PixelLayers upon loading a dataset, one for each category/set, which we can implement with a for loop, calling the createPixelLayer() function each time for every category/set.

Mouseover on a pixel should highlight that same pixel among all PixelLayers, we need the d3js mouseover eventListener to implement this and also to reload each PixelLayer when a new pixel is mouseover.

Drag-and-drop will also be implemented. The initial plan is as follows: upon mouse click, a boolean variable is set to true, and the PixelLayer will start following the mouse by transforming the x and y location attributes of the graph to be (mouse current location - mouse start location). A generated MultiLayer by dragging one or more PixelLayers onto another is really just another PixelLayer with data that we algorithmically filter out. For example, for AND operations, we only select pixels that were in all of the child PixelLayers. For OR operations, we accept all pixels as long as they were in at least one of the child PixelLayers.

Adding labels to the PixelLayers and MultiLayers is trivial and d3js has methods to do that. The necessary data analysis is category/set relationships, which is the point of OnSet visualizations. The visual encodings for the data will be color of the pixels/data elements, and relative position of the pixels to other pixels (can represent hierarchies).

We will only consider certain parts of the paper as described in the Must-Have features section as well as in the detailed technique implementation above, but will try to get pixel location hierarchies, which is an Optional Features, to be done as well since it is of value.

**Analysis:** Describe how you are going to analyze the technique with the data that you collect.

The technique will be effective if it is able to effectively visualize the data sets we use in the way that the paper suggested. In particular, the strengths of the technique compared to traditional visualizations should be apparent, and channels should be clear. This technique will hopefully be tested with multiple data sets as described previously.