Two Zooming/Navigation Problems

- First paper tackles clutter when zooming, by maintaining constant information density.
- Second paper attempts to address context loss when zooming in, but completely ignores (and abuses) information density.

Problem Domain

- Zooming in or out changes the effective area displayed on screen, changing the number of visible objects.
- Reducing density in multi-scale data (i.e., maps) has been shown to improve performance and visual appeal.
- Well-formed applications conform to the Principle of Constant Information Density (cartographic literature).
- To maintain constant information density at all zoom levels:
  - Show more object information when zooming in.
  - Show more objects when zooming in.
  - Opposite when zooming out: reduce information, aggregate objects.

DataSplash

- DataSplash database visualizer:
  - Create interactive zoomable interfaces
  - Associate object representation per layer
  - Objects change representation as elevation is zoomed in/out

DataSplash Details

- Visual objects associated with rows of the table:
  - x,y coordinates pulled from the table (i.e., longitude, latitude, but not limited to map data).
  - Generate a scatter plot per layer.
- Each object is part of one layer, each layer is associated with one database table.
- Interactively assign visibility of layers depending on elevation:
  - Resize and move the layer visibility bars.
**DataSplash Details**

- Associate columns of table with different display properties
  - Height, width, radius, colour, rotation
- Portals, or windows into other canvases:
  - i.e. City objects have portals into the city’s map
  - Portal history allows going back and forth between canvases

**Problems with DataSplash**

- Difficult to create visualizations with appropriate density and details at all elevations
- Process is time consuming, since all elevation layers must be manually verified whenever a change is introduced
- No feedback on information density

**Improved DataSplash**

- Provides density feedback at all layers and elevations
- Note the parabolic shape of the density, due to the quadratic relation between zoom and displayed area
- Tick mark colouring represents the aggregate density of all layers.
- Application extensions for custom density measurement functions.

**Visual Feedback of Density**

- Interacting with the layer visibility bars
  - automatically changes their width, reflecting their density at the new elevation levels
  - updates the ticks’ colour, showing aggregate density
- Joining or aggregating table rows in a layer also updates the bars and ticks, communicating the new density
- Teaches users about the relationship between zooming and number of visible objects

**Semi-Automated Layer Density**

- User drags sides of layer bar
- System applies several density modification functions to layer that fulfill requested density target
- Presents resulting canvases to the user through portals

**Modification Functions**

- Applied to a layer, to modify its intensity
  - Operate on data (aggregating rows)
  - Operate on the visual representation
- Examples:
  - Selecting (cities with population > n)
  - Aggregating (cities by state)
  - Changing shape of glyph (triangle = less ink)
  - Changing size of glyph
  - Changing colour
Method critique

- Hiding information, may be misleading
- Applies global density calculation, even to sparse areas, hiding information when unnecessary
- Abrupt shifts between layers, may cause popping effects

Paper critique

- Well written and detailed, providing justifications for implementation decisions
- Informal user study
  - Web based java applet
  - Evaluate user response to density variance
  - Uncontrolled study, some users said task was confusing
  - Results influenced by speed of different machines (avoiding dense layers due to responsiveness)

Questions?

Problem Domain

- Attempts to solve the problem of context loss when zooming in for more details
- Introduces the macroscope: overlapping, translucent zoomed-in and zoomed-out layers!
- Allegedly applicable to:
  - Charts
  - Maps
  - File browsers
  - Etc

The Macroscope
Would it work?

- Claims that multiple layers can still retain clarity when overlapped by using:
  - Translucency
  - Focus, blurring
  - Dynamic interaction, movement of layers
- Especially suited for multiple-resolution data, that has different representations at different zoom scales
  - Difference in features helps enhance the visual distinction between layers
- Claims that the human visual system is adept at discerning features at different scales, and separating the layers, even when superimposed

The Macroscope in practice

- Three layers, World to Country to City

Details

- The user can create viewports at any layer and can:
  - Resize the viewport (zooming the corresponding layer)
  - Move the viewport (panning the layer)
  - Change translucency of layers, highlighting the zoomed-out context, or the zoomed-in focus of interest
- The viewer is always oriented in space, since all zoom layers are visible
- Layers are highlighted when corresponding viewport is selected.

Some more map examples

- Viewport rectangle fades with the corresponding layer, to reduce clutter
- Maps are particularly suited for the macroscope, due to their sparse, high-contrast features (road lines, city dots, text labels). Really?

File browser example

- No opening/closing folders, just zooming

Method critique

- Cluttered mess
- Ambiguous what information comes from what layer
- By combining layers at different scales, false features can be introduced
- Anything past two layers is practically useless
  - To reduce visual density, zoomed out layers would have to be very translucent
  - Thus not very good file-browser replacement
  - Even worse for maps, introduces fake features
  - Does not adequately achieve its goal of maintaining context
Paper critique

- No analysis of information density, and perceptual effects of the overlapped cluster.
- Picture descriptions did not attempt in any way to address the most obvious drawback, instead: "see, you can still sort of tell the different layers apart" (paraphrased)
- No user study because the system was too slow (running on an old Macintosh 9500/200, realtime SGI system was apparently in the works).
- Future work: true 3D stereo viewing and 3D input device will fix it!

Questions?