What makes a good vis system?

- Infovis is based on the human perceptual system, so humans should be central in the design process.
What makes a good vis system?

- Infovis is based on the human perceptual system, so humans should be central in the design process.

- User testing should be central to infovis research.
What makes a good vis system?

- Infovis is based on the human perceptual system, so humans should be central in the design process.

- User testing should be central to infovis research.

- Without testing on real users in real situations, we have little basis for determining the effectiveness of visualisation tools.
The Need For Metrics In Visual Information Analysis

[Miller, Hetzler, Nakamura, and Whitney 1997]

- Compares four systems for “visualising the thematic content of large document collections.”
The Need For Metrics In Visual Information Analysis

[Miller, Hetzler, Nakamura, and Whitney 1997]

- Compares four systems for “visualising the thematic content of large document collections.”

- “How can we measure the “goodness” of a particular or combined visualization?”
• Each visualisation has its strengths and weaknesses, varying by user and task. Interaction is key, so metrics developed for static systems will not capture what we wish to measure.
• Each visualisation has its strengths and weaknesses, varying by user and task. Interaction is key, so metrics developed for static systems will not capture what we wish to measure.

• What is the point of this paper?
Graphical Encoding for Information Visualization: An Empirical Study

[Nowell, Schulman, and Hix 2002]

- How effective are colour, shape, and size in encoding information? The literature contains “inconclusive and often conflicting viewpoints.”
How effective are colour, shape, and size in encoding information? The literature contains “inconclusive and often conflicting viewpoints.”

The task: counting objects matching specified criteria - unidimensional and redundant codings. (Low-level task;
pop-out.)
pop-out.)

- Three sizes, three shapes, three colours. (Three because only three sizes were possible - but this is always a problem when using size...)
pop-out.)

- Three sizes, three shapes, three colours. (Three because only three sizes were possible - but this is always a problem when using size...)

- Nominal and quantitative data types (but discretized so really just ordinal? Is “document relevance” really quantitative?)
pop-out.)

- Three sizes, three shapes, three colours. (Three because only three sizes were possible - but this is always a problem when using size...)

- Nominal and quantitative data types (but discretized so really just ordinal? Is “document relevance” really quantitative?)

- Measured accuracy and time, and subjective “cognitive difficulty” and “desirability”.
Results: no surprise: colour best, shape marginally better than size. Redundant coding helps; colour + one is best; triple-redundancy often doesn’t help.
• Results: no surprise: colour best, shape marginally better than size. Redundant coding helps; colour + one is best; triple-redundancy often doesn’t help.

• Issues:
  ★ Discriminability - how many categories can be distinguished for each coding?
• Results: no surprise: colour best, shape marginally better than size. Redundant coding helps; colour + one is best; triple-redundancy often doesn’t help.

• Issues:
  ★ Discriminability - how many categories can be distinguished for each coding?
  ★ Task - pop-out / counting, measurement, comparison, large-scale pattern-finding, navigation, . . .
• Results: no surprise: colour best, shape marginally better than size. Redundant coding helps; colour + one is best; triple-redundancy often doesn’t help.

• Issues:

★ Discriminability - how many categories can be distinguished for each coding?
★ Task - pop-out / counting, measurement, comparison, large-scale pattern-finding, navigation, . . .
★ Measure - speed, accuracy, ease of use, ...
Results: no surprise: colour best, shape marginally better than size. Redundant coding helps; colour + one is best; triple-redundancy often doesn’t help.

Issues:

★ Discriminability - how many categories can be distinguished for each coding?
★ Task - pop-out / counting, measurement, comparison, large-scale pattern-finding, navigation, . . .
★ Measure - speed, accuracy, ease of use, ...
★ Extraneous information - can be detrimental.
An Empirical Comparison of Three Commercial Information Visualization Systems

[Kobsa 2001]

- **Eureka** (aka TableLens) - table.
- **InfoZoom** (aka Focus) - sideways table; compressed; overview mode.
- **SpotFire** - scatterplot, others.
“ecologically relevant” tasks that took 80 to 110 seconds (mean) - fairly complex.
• “ecologically relevant” tasks that took 80 to 110 seconds (mean) - fairly complex.

• InfoZoom is fast but bad for finding correlations - mode problems.
A key point:

“Keeping tasks simple makes it easier to attribute differences in task performance directly to the different types of visualization, and helps eliminate confounding factors. A drawback of studies with low-level tasks is however their unclear ecological relevance: how frequently do these low-level tasks actually occur in real-world tasks, and how significant are they in the overall task solution process?”
• Higher-level (problem-solving) tasks - choosing type of visualisation; variable selection; navigation; filtering. General user interface usability is important in determining how quickly and effectively users can solve problems.
Higher-level (problem-solving) tasks - choosing type of visualisation; variable selection; navigation; filtering. General user interface usability is important in determining how quickly and effectively users can solve problems.

What about more experienced users?
Snap-together visualization: can users construct and operate coordinated visualizations?

[North and Shneiderman, 2000]

- **Snap** “enables users to rapidly and dynamically construct coordinated-visualization interfaces, customized for their data, without programming.”
“data-savvy users successfully, enthusiastically and rapidly constructed powerful coordinated-visualization interfaces of their own.”
“data-savvy users successfully, enthusiastically and rapidly constructed powerful coordinated-visualization interfaces of their own.”

The concept:

★ Different tools (types of visualisations) should be used for different levels of data exploration.
“data-savvy users successfully, enthusiastically and rapidly constructed powerful coordinated-visualization interfaces of their own.”

The concept:

★ Different tools (types of visualisations) should be used for different levels of data exploration.
★ These tools should be linked.
• “data-savvy users successfully, enthusiastically and rapidly constructed powerful coordinated-visualization interfaces of their own.”

• The concept:
  ★ Different tools (types of visualisations) should be used for different levels of data exploration.
  ★ These tools should be linked.
  ★ The number of possible combinations is too large for the programmer to design everything in advance.
“data-savvy users successfully, enthusiastically and rapidly constructed powerful coordinated-visualization interfaces of their own.”

The concept:
- Different tools (types of visualisations) should be used for different levels of data exploration.
- These tools should be linked.
- The number of possible combinations is too large for the programmer to design everything in advance.
- Let the users do it!
Evaluate:

- Do users understand coordination?
Evaluate:

- Do users understand coordination?
- Can they build coordinated visualisations?
• Evaluate:
  ★ Do users understand coordination?
  ★ Can they build coordinated visualisations?
  ★ “what aspect of ... coordinated visualizations caused improved performance [?] Was it the additional information displayed in the multiple visualizations or the interactive coordination between them?”
• Evaluate:

- Do users understand coordination?
- Can they build coordinated visualisations?
- “what aspect of ... coordinated visualizations caused improved performance [?] Was it the additional information displayed in the multiple visualizations or the interactive coordination between them?”

• Snap places a “Snap” button in each vis window. Drag-and-dropping between Snap buttons opens a Snap dialog, in which users can specify the coordination between the visualisations.
• The Snap dialog looks clunky.
• The Snap dialog looks clunky.

• Two studies:
  ✫ “can users successfully construct their own coordinated-visualization interfaces?”
• The Snap dialog looks clunky.

• Two studies:
  ✮ “can users successfully construct their own coordinated-visualization interfaces?”
  ✮ “can users then operate the constructed coordinated-visualization interfaces to explore information beneficially?”
• The Snap dialog looks clunky.

• Two studies:
  ✓ “can users successfully construct their own coordinated-visualization interfaces?”
  ✓ “can users then operate the constructed coordinated-visualization interfaces to explore information beneficially?”

• First study: test subjects reported a “sense of satisfaction and power in being able to ... quickly snap powerful exploration environments together and ... see the many parts operate as
a whole.” Snapping together interfaces took them from 2 to 15 minutes.
a whole.” Snapping together interfaces took them from 2 to 15 minutes.

- Second study: coordinated vs. multiple uncoordinated vs. single visualisations.
a whole.” Snapping together interfaces took them from 2 to 15 minutes.

- Second study: coordinated vs. multiple uncoordinated vs. single visualisations.

- Coordinated wins, especially for more complex tasks. Users like coordination.
Discussion

- We are exploring *evaluation* of infovis systems.
Discussion

- We are exploring *evaluation* of infovis systems.

- We have seen different approaches in the papers:
Discussion

• We are exploring *evaluation* of infovis systems.

• We have seen different approaches in the papers:
  ✫ ‘Head-to-head’ [Kobsa] - compare user performance on high-level tasks using established systems.
Discussion

- We are exploring *evaluation* of infovis systems.

- We have seen different approaches in the papers:
  - ‘Head-to-head’ [Kobsa] - compare user performance on high-level tasks using established systems.
  - ‘Feasibility’ [North & Shneiderman, study #1] - can users Snap?
Discussion

- We are exploring *evaluation* of infovis systems.

- We have seen different approaches in the papers:
  - ‘Situated psychophysical’ [Nowell *et al.*] - verification of low-level glyph interpretation guidelines.
  - ‘Head-to-head’ [Kobsa] - compare user performance on high-level tasks using established systems.
  - ‘Feasibility’ [North & Shneiderman, study #1] - can users Snap?
‘Head-to-head’ [North & Shneiderman, study #2] - compare ‘snapped’ interfaces to vanilla ones; high-level tasks.
‘Head-to-head’ [North & Shneiderman, study #2] - compare ‘snapped’ interfaces to vanilla ones; high-level tasks.

- It is tempting to try bottom-up psychophysical-style evaluations that yield solid guidelines.
‘Head-to-head’ [North & Shneiderman, study #2] - compare ‘snapped’ interfaces to vanilla ones; high-level tasks.

- It is tempting to try bottom-up psychophysical-style evaluations that yield solid guidelines.

- But it is difficult to devise ‘abstract tasks’ - the details always seem to be important.
‘Head-to-head’ [North & Shneiderman, study #2] - compare ‘snapped’ interfaces to vanilla ones; high-level tasks.

- It is tempting to try bottom-up psychophysical-style evaluations that yield solid guidelines.

- But it is difficult to devise ‘abstract tasks’ - the details always seem to be important.

- Good low-level design can not compensate for clunky high-
level interfaces.
level interfaces.

- Visualisations are grounded in a GUI context - without a good GUI, even good visualisation strategies cannot be used effectively.
level interfaces.

- Visualisations are grounded in a GUI context - without a good GUI, even good visualisation strategies cannot be used effectively.

- **Top-down** testing then seems to be the way to go.
But:

- Who tests systems (bias - task selection, data sets, users, ...)?
But:

- Who tests systems (bias - task selection, data sets, users, ...)?
- How does training effect results?
But:

★ Who tests systems (bias - task selection, data sets, users, ...)?
★ How does training effect results?
★ Measures (speed, accuracy, ease of use, ...)?
• But:
  ★ Who tests systems (bias - task selection, data sets, users, ...)?
  ★ How does training effect results?
  ★ Measures (speed, accuracy, ease of use, ...)?
  ★ Can we generalise the results?
• But:

★ Who tests systems (bias - task selection, data sets, users, ...)?
★ How does training effect results?
★ Measures (speed, accuracy, ease of use, ...)?
★ Can we generalise the results?
★ How much is good vis and how much is good general GUI design?
Fin

Thanks!