Seminar Software Composition: Project P6

How are Software Visualizations Evaluated?

Presented by: Lukas Imstepf
lukas.imstepf@unifr.ch

Department of Informatics
University of Fribourg

June 27, 2017
Research Method
Systematic Literature Review

Research Method

- Following Kitchenham’s guidelines for systematic literature reviews in software engineering.

B. Kitchenham (2004): “Procedures for Performing Systematic Reviews”
1. Search for:

"software visualization" OR
"software visualisation"
Search Strategy and Data Sources

Research Method

1. Search for:
   "software visualization" OR "software visualisation"

2. in three scientific online databases:
   - ACM Digital Library
   - IEEE Xplore DL
   - ScienceDirect
1. Search for:
   "software visualization" OR "software visualisation"

2. in three scientific online databases:
   - ACM Digital Library
   - IEEE Xplore DL
   - ScienceDirect

3. Download the search results as BibTeX/CSV files, converting CSV to BibTeX with bibsani
Data Scraping and Early Exclusion Criteria

Research Method

- ACM Digital Library
  - search
  - BibTeX

- IEEE Xplore DL
  - search
  - csvtobib
  - CSV
  - BibTeX

- ScienceDirect
  - search
  - bibsani
  - BibTeX
  - BibTeX

- pdfdbscrap
  - download

Exclusion criteria: data sanitization (incomplete entries, duplicates: −100)
Exclusion criteria: scraping failures (−66, +7 manual downloads)
Data Scraping and Early Exclusion Criteria

Research Method

- ACM Digital Library
  - search
  - BibTeX

- IEEE Xplore DL
  - search
  - csvtoibib
  - CSV
  - BibTeX

- ScienceDirect
  - search
  - bibsani
  - BibTeX

- pdfdbscrap
  - download

- Exclusion criteria: data sanitization (incomplete entries, duplicates: -100)
- Exclusion criteria: scraping failures (-66, +7 manual downloads)

- 1289 search results
Data Scraping and Early Exclusion Criteria

Research Method

- 1289 search results
- Exclusion criteria: data sanitization (incomplete entries, duplicates: −100)
Data Scraping and Early Exclusion Criteria

Research Method

- 1289 search results
- Exclusion criteria: data sanitization (incomplete entries, duplicates: $-100$)
- Exclusion criteria: scraping failures ($-66$, $+7$ manual downloads)
Exclusion Criteria

Research Method

- Subtotal: 1130 scrapped PDF files

Further exclusion criteria:

1. Fewer than five pages (−279, subtotal = 851)
2. InfoVis (medical/geographical) papers (−318, subtotal = 533)
3. Exclusion by paper type:
   3.1 Technique papers (novel algorithms)
   3.2 Design study papers (particular domain problems)
   3.3 Systems papers (architectural choices)
   3.4 Evaluation papers
   3.5 Model papers (taxonomy, formalisms, commentary)

Exclusion Criteria

Research Method

- Subtotal: 1130 scrapped PDF files

Further exclusion criteria:

1. Fewer than five pages

(−279, subtotal = 851)
Exclusion Criteria

Further exclusion criteria:

1. Fewer than five pages \((-279, \text{ subtotal } = 851)\)
2. InfoVis (medical/geographical) papers \((-318, \text{ subtotal } = 533)\)
Exclusion Criteria

Research Method

Subtotal: 1130 scrapped PDF files

Further exclusion criteria:

1. Fewer than five pages \((-279, \text{subtotal} = 851)\)
2. InfoVis (medical/geographical) papers \((-318, \text{subtotal} = 533)\)
3. Exclusion by paper type:
   3.1 Technique papers (novel algorithms)
   3.2 Design study papers (particular domain problems)
   3.3 Systems papers (architectural choices)
   3.4 Evaluation papers
   3.5 Model papers (taxonomy, formalisms, commentary)

Exclusion Criteria: Paper Type

SoftVis papers from 1992 to 2017

- TECH ($\Sigma = 165$)
- SYS ($\Sigma = 160$)
- MODEL ($\Sigma = 71$)
- EVAL ($\Sigma = 50$)
- DESIGN ($\Sigma = 62$)
- EXCL ($\Sigma = 25$)

Yearly published papers from 1995 to 2017.
Exclusion Criteria: Recap

Research Method

Search results: 1289
Exclusion Criteria: Recap

Research Method

- Search results: 1189
- Data sanitization: 100
Exclusion Criteria: Recap

Research Method

- Search results: 1130
- Data sanitization: 100
- Scraping failures: 59
Exclusion Criteria: Recap

Research Method

- Search results: 851
- Data sanitization: 100
- Scraping failures: 59
- Fewer than five pages: 279
Exclusion Criteria: Recap

Research Method

- SoftVis papers: 533
- Data sanitization: 100
- Scraping failures: 59
- Fewer than five pages: 279
- InfoVis papers: 318
Exclusion Criteria: Recap

- **SoftVis papers**: 508
- **InfoVis papers**: 318
- **Scraping failures**: 59
- **Data sanitization**: 100
- **Fewer than five pages**: 279

Paper type: exclusions
Exclusion Criteria: Recap

Research Method

- SoftVis papers: 343
- Data sanitization: 100
- Scraping failures: 59
- Fewer than five pages: 279
- InfoVis papers: 318
- Paper type: exclusions: 25
- Paper type: technique: 165
Exclusion Criteria: Recap

Research Method

- Scraping failures: 100
- Data sanitization: 100
- Fewer than five pages: 279
- SoftVis papers: 183
- Paper type: exclusions: 25
- Paper type: technique: 165
- Paper type: system: 160

Total: 733
Exclusion Criteria: Recap

Research Method

- Fewer than five pages: 279
- Scraping failures: 59
- Data sanitization: 100
- Paper type: design study: 62
- Paper type: exclusions: 25
- Paper type: technique: 165
- Paper type: system: 160
- Paper type: model: 71
- Paper type: evaluation: 50
- InfoVis papers: 318
Exclusion Criteria: Recap

Research Method

- Fewer than five pages: 279
- Scraping failures: 100
- Data sanitization: 59
- Paper type: design study: 62
- Paper type: technique: 165
- Paper type: system: 160
- Paper type: model: 71
- Paper type: evaluation: 50
- InfoVis papers: 318
62 Design studies
Venues

Design studies, $N = 62$
Target audience

Design studies, $N = 62$

![Graph showing the number of design studies by year for different categories: professional ($\Sigma = 42$), education ($\Sigma = 10$), academic/professional ($\Sigma = 9$), academic ($\Sigma = 1$).](image)
Programming Paradigms

Design studies, $N = 62$

- other ($\Sigma = 20$)
- none ($\Sigma = 24$)
- Object-oriented programming ($\Sigma = 18$)
Visualizations and evaluations

Overview

- Number of visualizations: $\Sigma = 121$
- Number of evaluations: $\Sigma = 79$
- Per selected design study ($N = 62$)
121 Visualizations of 62 design studies
Usage of visualization frameworks

Design study papers, $N = 62$

- Roassal/Moose: 7
- d3.js: 2
- not identified: 50
- other*: 3
- professional: 42
- education: 10
- academic: 1
- academic/professional: 9

*other visualization frameworks: Graphplace, Flatland, Sovis
Presentation of a new visualization tool?

Design study papers, $N = 62$

- professional: 42
- academic/professional: 9
- academic: 1
- education: 10
- yes: 48
- update: 6
- no: 8
Who needs visualizations, and why?

**Programming paradigms vs. programming languages**

- None: 24
- Constraint programming (declarative): 2
- Procedural programming: 7
- Parallel computing: 3
- Aspect-oriented programming (OOP): 1
-Automata-based programming: 1

<table>
<thead>
<tr>
<th>Programming Paradigm</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Imperative programming</td>
<td>1</td>
</tr>
<tr>
<td>Procedural programming</td>
<td>7</td>
</tr>
<tr>
<td>None</td>
<td>24</td>
</tr>
<tr>
<td>Distributed programming</td>
<td>4</td>
</tr>
<tr>
<td>Object-oriented programming</td>
<td>17</td>
</tr>
<tr>
<td>Parallel computing</td>
<td>3</td>
</tr>
<tr>
<td>Constraint programming (declarative)</td>
<td>2</td>
</tr>
<tr>
<td>Concurrent programming</td>
<td>2</td>
</tr>
<tr>
<td>Aspect-oriented programming (OOP)</td>
<td>1</td>
</tr>
<tr>
<td>Automata-based programming</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Language</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>6</td>
</tr>
<tr>
<td>C++</td>
<td>2</td>
</tr>
<tr>
<td>C/C++</td>
<td>3</td>
</tr>
<tr>
<td>Not specified</td>
<td>3</td>
</tr>
<tr>
<td>Python</td>
<td>2</td>
</tr>
<tr>
<td>Java</td>
<td>14</td>
</tr>
<tr>
<td>C++</td>
<td>2</td>
</tr>
<tr>
<td>Visual Basic .NET</td>
<td>1</td>
</tr>
<tr>
<td>Pharo</td>
<td>1</td>
</tr>
<tr>
<td>Language independent</td>
<td>29</td>
</tr>
<tr>
<td>Assembly language</td>
<td>1</td>
</tr>
<tr>
<td>Assembly language</td>
<td>1</td>
</tr>
</tbody>
</table>
Who needs visualizations, and why?

Programming paradigms vs. programming languages

- None: 24
- Constraint programming (declarative): 2
- Procedural programming: 7
- Parallel computing: 3
- Aspect-oriented programming (OOP): 1
- Concurrent programming: 2
- Automata-based programming: 1
- Distributed programming: 4
- Object-oriented programming: 17
- Parallel computing: 3
- Assembly language: 1
- Python: 2
- C++: 2
- C/C++: 3
- Not specified: 3
- Java: 14
- C: 6
- Python: 2
- C++: 2
- Visual Basic .NET: 1
- Language independent: 29
- Java: 14
- C: 6
- Pharo: 1
Who needs visualizations, and why?

Programming paradigms vs. programming languages

- None: 24
- Constraint programming (declarative): 2
- Procedural programming: 7
- Parallel computing: 3
- Aspect-oriented programming (OOP): 1
- Distributed programming: 4
- Concurrency programming: 2
- Automata-based programming: 1
- Imperative programming: 1
- Assembly language: 1
- Python: 2
- C++: 2
- C/C++: 3
- Not specified: 3
- Visual Basic .NET: 1
- Java: 14
- C: 6
- Python: 2
- Java: 14
- Pharo: 1
- Language independent: 29
Who needs visualizations, and why?
Target audience vs. problem domain visualization contribution
Who needs visualizations, and why?

Target audience vs. problem domain visualization contribution

- Professional: 111
- Education: 16
- Academic/professional: 22
- Academic: 2

- Exploring change in software over time: 18
- Defining and maintaining requirements: 14
- Managing software projects: 4
- Understanding software execution: 51
- Understanding software structure: 58
- How software is developed: 6
Who needs visualizations, and why?

Target audience vs. problem domain visualization contribution

- Exploring change in software over time: 18
- Defining and maintaining requirements: 14
- Managing software projects: 4
- Understanding software execution: 51
- Understanding software structure: 58
- How software is developed: 6

- Professional: 111
- Education: 16
- Academic/professional: 22
- Academic: 2
Who needs visualizations, and why?
Target audience vs. problem domain visualization contribution

- Exploring change in software over time: 18
- Defining and maintaining requirements: 14
- Managing software projects: 4
- Understanding software execution: 51
- Understanding software structure: 58
- How software is developed: 6

professional: 111
education: 16
academic/professional: 22
academic: 2
Who needs visualizations, and why?
Target audience vs. problem domain visualization contribution

- Exploring change in software over time: 18
- Defining and maintaining requirements: 14
- Managing software projects: 4
- Understanding software execution: 51
- Understanding software structure: 58
- How software is developed: 6

professional: 111
education: 16
academic/professional: 22
academic: 2
Who needs visualizations, and why?

Target audience vs. data visualization questions

Who needs visualizations, and why?

Target audience vs. data visualization questions

Who needs visualizations, and why?

Target audience vs. data visualization questions

Who needs visualizations, and why?

Target audience vs. data visualization questions

Who needs visualizations, and why?

Target audience vs. data visualization questions

What are the sources for which visualizations?

Visualization sources vs. visualization paradigms

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What are the sources for which visualizations?

Visualization sources vs. visualization paradigms

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What are the sources for which visualizations?

Visualization sources vs. visualization paradigms

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What are the sources for which visualizations?

Visualization sources vs. visualization paradigms

- Issue management data: 3
- Version control system data: 37
- Sourcecode: 25
- Static code analysis data: 95
- Software execution data: 83
- Mailing list: 6

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What are the sources for which visualizations?

Visualization sources vs. visualization paradigms

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
### What are the sources for which visualizations?

#### Visualization sources vs. visualization paradigms

<table>
<thead>
<tr>
<th>Source Type</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Issue management data</td>
<td>3</td>
</tr>
<tr>
<td>Version control system data</td>
<td>37</td>
</tr>
<tr>
<td>Sourcecode</td>
<td>25</td>
</tr>
<tr>
<td>Static code analysis data</td>
<td>95</td>
</tr>
<tr>
<td>Software execution data</td>
<td>83</td>
</tr>
<tr>
<td>Mailing list</td>
<td>6</td>
</tr>
<tr>
<td>Source code highlighting</td>
<td>22</td>
</tr>
</tbody>
</table>

#### Visualization Paradigms
- Hierarchical and Graph-Based Techniques: 115
- 3D techniques: 19
- Info graphics: 20
- Animation: 18
- Pixel-oriented techniques: 5
- Source code highlighting: 22
- Icon-based techniques / Icon displays: 17
- Timelines: 30
- Geometric projection techniques: 3
- 3D techniques: 19

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What are the sources for which visualizations?

Visualzation sources vs. visualization paradigms

- Static code analysis data: 95
- Version control system data: 37
- Sourcecode: 25
- Static code analysis data: 95
- Software execution data: 83
- Issue management data: 3
- Mailing list: 6

Geometric projection techniques: 3
Hierarchical and Graph-Based Techniques: 115
Icon-based techniques / Icon displays: 17
3D techniques: 19
Timelines: 30
Info graphics: 20
Animation: 18
Pixel-oriented techniques: 5
Source code highlighting: 22

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What is visualized?

Problem domain visualization contribution vs. visualization paradigms

Managing software projects: 5
Understanding software execution: 85
How software is developed: 16
Exploring change in software over time: 33
Understanding software structure: 81
Defining and maintaining requirements: 21
Managing software projects: 5
Understanding software execution: 85
How software is developed: 16
Exploring change in software over time: 33
Understanding software structure: 81
Defining and maintaining requirements: 21
Info graphics: 20
Geometric projection techniques: 4
Pixel-oriented techniques: 5
Timelines: 29
Hierarchical and Graph-Based Techniques: 113
Animation: 17
3D techniques: 19
Source code highlighting: 17
Icon-based techniques / Icon displays: 17

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What is visualized?

Problem domain visualization contribution vs. visualization paradigms

Managing software projects: 5
Understanding software execution: 85
How software is developed: 16
Exploring change in software over time: 33
Understanding software structure: 81
Defining and maintaining requirements: 21
Info graphics: 20
Geometric projection techniques: 4
Pixel-oriented techniques: 5
Timelines: 29
Hierarchical and Graph-Based Techniques: 113
Animation: 17
3D techniques: 19
Source code highlighting: 17
Icon-based techniques / Icon displays: 17

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What is visualized?

Problem domain visualization contribution vs. visualization paradigms

- Managing software projects: 5
- Understanding software execution: 85
- How software is developed: 16
- Exploring change in software over time: 33
- Understanding software structure: 81
- Defining and maintaining requirements: 21
- Timelines: 29
- Info graphics: 20
- Geometric projection techniques: 4
- Pixel-oriented techniques: 5
- Hierarchical and Graph-Based Techniques: 113
- 3D techniques: 19
- Animation: 17
- Source code highlighting: 17
- Icon-based techniques / Icon displays: 17

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What is visualized?

Problem domain visualization contribution vs. visualization paradigms

- Managing software projects: 5
- Understanding software execution: 85
- How software is developed: 16
- Exploring change in software over time: 33
- Understanding software structure: 81
- Defining and maintaining requirements: 21
- Understanding software execution: 85
- How software is developed: 16
- Exploring change in software over time: 33
- Understanding software structure: 81
- Defining and maintaining requirements: 21

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What is visualized?

Problem domain visualization contribution vs. visualization paradigms

- Managing software projects: 5
- Understanding software execution: 85
- Exploring change in software over time: 33
- Understanding software structure: 81
- Defining and maintaining requirements: 21
- How software is developed: 16
- Managing software projects: 5
- Understanding software execution: 85
- Exploring change in software over time: 33
- Understanding software structure: 81
- Defining and maintaining requirements: 21

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What is visualized?

Problem domain visualization contribution vs. visualization paradigms

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
What is visualized?

Problem domain visualization contribution vs. visualization paradigms

Visualization Paradigms by Keim, Kriegel (1996): “Visualization Techniques for Mining Large Databases: A Comparison”
79 Evaluations of 62 design studies
To evaluate, or not to evaluate...

- **Evaluation presented**: 77.4%
- **Evaluation planned**: 11.3%
- **No evaluation**: 11.3%

Percentage of selected design studies ($N = 62$)
What is evaluated, and when?

Evaluation scope vs. evaluation aspects

- Validating functionality of a visualization tool: 3
- Evaluating visualization algorithms: 21
- Evaluating user experience: 28
- Evaluating user performance; time and accuracy: 6
- Understanding environments and work practices: 15
- Evaluating visual data analysis and reasoning: 41
- Evaluating communication through visualization: 4
- Evaluating collaborative data analysis: 1

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and when?

Evaluation scope vs. evaluation aspects

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and when?

Evaluation scope vs. evaluation aspects

- Pre-design: 2
- Design: 88
- Prototype: 104
- Deployment: 44
- Validating functionality of a visualization tool: 6
- Evaluating visualization algorithms: 42
- Evaluating user performance; time and accuracy: 12
- Evaluating user experience: 56
- Understanding environments and work practices: 30
- Evaluating visual data analysis and reasoning: 82
- Evaluating communication through visualization: 8
- Evaluating collaborative data analysis: 2

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and when?

Evaluation scope vs. evaluation aspects

Pre-design: 1
Design: 44
Prototype: 52
Deployment: 22
Validating functionality of a visualization tool: 3
Evaluating visualization algorithms: 21
Evaluating user performance; time and accuracy: 6
Understanding environments and work practices: 15
Evaluating communication through visualization: 4
Evaluating visual data analysis and reasoning: 41
Evaluating user experience: 28
Evaluating collaborative data analysis: 1

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and when?
Evaluation scope vs. evaluation aspects

- Validating functionality of a visualization tool: 3
- Evaluating visualization algorithms: 21
- Evaluating user experience: 28
- Evaluating user performance; time and accuracy: 6
- Understanding environments and work practices: 15
- Evaluating visual data analysis and reasoning: 41
- Evaluating communication through visualization: 4
- Evaluating collaborative data analysis: 1

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and when?

Evaluation scope vs. evaluation methods

Pre-design: 1
Design: 32
Prototype: 36
Deployment: 13
Pilot (or exploratory) study: 5
Informal evaluation: 16
Interview: 1
Laboratory observation: 3
Heuristic evaluation: 1
Questionnaire/Questionary: 12
Pilot (or exploratory) study: 5
Usability test: 2
Case study: 18
Comparative study (concurrent control): 8
Algorithmic performance: 5
Field observation: 10
Log analysis: 1
What is evaluated, and when?

Evaluation scope vs. evaluation methods

- Pre-design: 1
- Design: 32
- Prototype: 36
- Deployment: 13
- Pilot (or exploratory) study: 5
- Informal evaluation: 16
- Interview: 1
- Laboratory observation: 3
- Heuristic evaluation: 1
- Questionnaire/Questionary: 12
- Field observation: 10
- Laboratory observation: 3
- Usability test: 2
- Log analysis: 1
- Comparative study (concurrent control): 8
- Usability test: 2
- Algorithmic performance: 5
- Case study: 18
- Field observation: 10
- Log analysis: 1
What is evaluated, and when?

Evaluation scope vs. evaluation methods

- Pre-design: 1
- Design: 32
- Prototype: 36
- Deployment: 13
- Pilot (or exploratory) study: 5
- Informal evaluation: 16
- Interview: 1
- Laboratory observation: 3
- Questionnaire/Questionary: 12
- Pilot (or exploratory) study: 5
- Usability test: 2
- Case study: 18
- Comparative study (concurrent control): 8
- Algorithmic performance: 5
- Field observation: 10
- Log analysis: 1

24/33
What is evaluated, and when?

Evaluation scope vs. evaluation methods

- Pre-design: 1
- Design: 32
- Prototype: 36
- Deployment: 13
- Pilot (or exploratory) study: 5
- Informal evaluation: 16
- Interview: 1
- Laboratory observation: 3
- Usability test: 2
- Heuristic evaluation: 1
- Questionnaire/Questionary: 12
- Comparative study (concurrent control): 8
- Field observation: 10
- Case study: 18
- Algorithmic performance: 5
- Log analysis: 1
What is evaluated, and when?

Evaluation scope vs. evaluation methods
What is evaluated, and how?

Evaluation aspects vs. evaluation methods

Evaluating user experience: 30
Evaluating visual data analysis and reasoning: 42
Evaluating communication through visualization: 4
Validating functionality of a visualization tool: 4
Evaluating visualization algorithms: 22
Evaluating collaborative data analysis: 1
Evaluating user performance; time and accuracy: 6
Understanding environments and work practices: 15

Interview: 1
Usability test: 2
Questionnaire/Questionary: 17
Field observation: 18
Pilot (or exploratory) study: 10
Informal evaluation: 21
Case study: 31
Laboratory observation: 4
Log analysis: 1
Comparative study (concurrent control): 12
Heuristic evaluation: 2
Algorithmic performance: 5

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and how?

Evaluation aspects vs. evaluation methods

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and how?

Evaluation aspects vs. evaluation methods

Evaluating user experience: 30
Understanding environments and work practices: 15
Evaluating visual data analysis and reasoning: 42
Evaluating communication through visualization: 4
Validating functionality of a visualization tool: 4
Evaluating visualization algorithms: 22
Evaluating collaborative data analysis: 1
Evaluating user performance; time and accuracy: 6

Interview: 1
Usability test: 2
Questionnaire/Questionary: 17
Field observation: 18
Pilot (or exploratory) study: 10
Informal evaluation: 21
Case study: 31
Laboratory observation: 4
Log analysis: 1
Comparative study (concurrent control): 12
Heuristic evaluation: 2
Algorithmic performance: 5

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and how?

Evaluation aspects vs. evaluation methods

- Validating functionality of a visualization tool: 4
- Evaluating visualization algorithms: 22
- Evaluating user performance; time and accuracy: 6
- Evaluating collaborative data analysis: 1
- Evaluating user experience: 30
- Understanding environments and work practices: 15
- Evaluating visual data analysis and reasoning: 42
- Evaluating communication through visualization: 4
- Comparative study (concurrent control): 12
- Usability test: 2
- Heuristic evaluation: 2
- Log analysis: 1
- Algorithmic performance: 5
- Pilot (or exploratory) study: 10
- Informal evaluation: 21
- Field observation: 18
- Case study: 31
- Laboratory observation: 4

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and how?

Evaluation aspects vs. evaluation methods

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and how?

Evaluation aspects vs. evaluation methods

Validating functionality of a visualization tool: 4
Evaluating visualization algorithms: 22
Evaluating user experience: 30
Evaluating collaborative data analysis: 1
Evaluating user performance; time and accuracy: 6
Evaluating visual data analysis and reasoning: 42
Evaluating communication through visualization: 4
Understanding environments and work practices: 15
Evaluating visualization algorithms: 22
Pilot (or exploratory) study: 10
Case study: 31
Informal evaluation: 21
Laboratory observation: 4
Comparative study (concurrent control): 12
Heuristic evaluation: 2
Algorithmic performance: 5
Log analysis: 1
Usability test: 2
Questionnaire/Questionary: 17
Field observation: 18
Interview: 1

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and how?

Evaluation aspects vs. evaluation methods

Evaluating user experience: 30
Understanding environments and work practices: 15
Evaluating visual data analysis and reasoning: 42
Evaluating communication through visualization: 4
Validating functionality of a visualization tool: 4
Evaluating visualization algorithms: 22
Evaluating collaborative data analysis: 1
Evaluating user performance; time and accuracy: 6

Interview: 1
Usability test: 2
Questionnaire/Questionary: 17
Field observation: 18
Pilot (or exploratory) study: 10
Informal evaluation: 21
Case study: 31
Laboratory observation: 4
Log analysis: 1
Comparative study (concurrent control): 12
Heuristic evaluation: 2
Algorithmic performance: 5

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
What is evaluated, and how?

Evaluation aspects vs. evaluation methods

- Evaluating user experience: 30
- Understanding environments and work practices: 15
- Evaluating visual data analysis and reasoning: 42
- Evaluating communication through visualization: 4
- Validating functionality of a visualization tool: 4
- Evaluating visualization algorithms: 22
- Evaluating collaborative data analysis: 1
- Evaluating user performance; time and accuracy: 6

Evaluation aspects by Lam et al. (2012): "Empirical Studies in Information Visualization: Seven Scenarios"
What is evaluated, and how?

Evaluation aspects vs. evaluation methods

Validating functionality of a visualization tool: 4
Evaluating visualization algorithms: 22
Evaluating user performance; time and accuracy: 6
Evaluating collaborative data analysis: 1
Evaluating visual data analysis and reasoning: 42
Evaluating communication through visualization: 4
Validating functionality of a visualization tool: 4
Evaluating user experience: 30
Understanding environments and work practices: 15

Evaluation aspects by Lam et al. (2012): “Empirical Studies in Information Visualization: Seven Scenarios”
Who are the evaluation subjects?

Target audience vs. evaluation subjects

- Professional: 42
- Novice user: 19
- Education: 10
- Academic/professional: 9
- Academic: 1
- Professional user: 15
- Academic user: 28
- Novice user: 19
- Education: 10
And the winner is...

Evaluation score by target audience

Evaluation method ranking roughly based on B. Kitchenham’s “Study design hierarchy for Software Engineering”
And the winner is... 

Evaluation score by venue

Evaluation method ranking roughly based on B. Kitchenham’s “Study design hierarchy for Software Engineering”
Does the evaluation score improve over time?

Evaluation score by year

Evaluation method ranking roughly based on B. Kitchenham’s “Study design hierarchy for Software Engineering”
Discussion
Results

Discussion

- Widely different software visualization needs based on:
  - the target audience (e.g. professional vs. education)
  - the programming language/paradigm
Results

Discussion

- Widely different software visualization needs based on:
  - the target audience (e.g. professional vs. education)
  - the programming language/paradigm

- Naturally different evaluation aspects and methods depending on evaluation scope (or stage):
  - Informal evaluation methods in early evaluation scopes
  - More formal evaluation methods with later evaluation scopes
  - Overall rather low evaluation effort
  - Minimum standard seems to have been established
Results

Discussion

- Widely different software visualization needs based on:
  - the target audience (e.g. professional vs. education)
  - the programming language/paradigm

- Naturally different evaluation aspects and methods depending on evaluation scope (or stage):
  - Informal evaluation methods in early evaluation scopes
  - More formal evaluation methods with later evaluation scopes
  - Overall rather low evaluation effort
  - Minimum standard seems to have been established

- Consolidation from other venues to SOFTVIS/VISSOFT
Limitations/threats to validity

Discussion

- Selection bias (ACM, IEEE, SD)
Limitations/threats to validity

Discussion

- Selection bias (ACM, IEEE, SD)
- Scope of a seminar project vs. systematic literature review
  - Iterative process (expertise and time)
  - Establishing consensus (peer-review/repeatable?)
Limitations/threats to validity

Discussion

- Selection bias (ACM, IEEE, SD)
- Scope of a seminar project vs. systematic literature review
  - Iterative process (expertise and time)
  - Establishing consensus (peer-review/repeatable?)
- Paper classification (What is a design study?)
Limitations/threats to validity

Discussion

- Selection bias (ACM, IEEE, SD)
- Scope of a seminar project vs. systematic literature review
  - Iterative process (expertise and time)
  - Establishing consensus (peer-review/repeatable?)
- Paper classification (What is a design study?)
- Data extraction
Thanks for listening!
Thanks for listening! Questions?