Final presentations timing
• final presentations timing
  – Original plan: 1-5 Tue (26)
  – ML Fus: 12-17:20-20:30
  – Best availability: 3-7 Tue (26)
  – Worse Mon (21), Wed (24), Thu (20)
• reminder
  – we do have class next time (Tue Dec 3), since started a week late

• peer review 2
  – do remember to submit your peer review dates
  – for this one, also upload notes as comments

To day
• finalize final presentation slot; Tue Dec 10 3-7pm
• presentations
• final papers and final presentations
  – course paper vs research paper expectations
• writing info's papers: pitfalls to avoid
• other research pitfalls and process
  – review reading; reviewing, conference talks
  – reproducible and repeatable research

Sample outlines: Research Papers and Process
Tamara Munzner
Department of Computer Science
University of British Columbia

CPC 547. Information Visualization
26 November 2019

http://www.cs.ubc.ca/~tmm/courses/547-19

Wrapup: Research Papers and Process

PDF use InfoVis templates http://plasticpublishing.org/research/courses/munzner.html
• no length cap: illustrate freely with screenshoots! – design study / technique / aim for at least 6-8 pages
– analysis / survey: aim for at least 15-20 pages
– ok to re-use text from proposal, interim writeup
– encourage looking at my writing correct and style guidelines – http://www.cs.ubc.ca/~tmm/writing.html
– strongly encourage looking at previous examples – www.cs.ubc.ca/~tmm/courses/547-19/projectdesc.html#examp
– Example Past Projects
  – browse 2015, 2014,… reports

Course requirements vs research paper standards
• research novelty not required
• model your choice to use Literature/Whatssoever
– high level: topics like how did you deal
– medium level: what pre-existing features did you implement
– ok to re-use text from proposal, interim writeup
– encourage looking at my writing correct and style guidelines
– strongly encourage looking at previous examples
– should you actually do a review writing?
– strongly encourage looking at previous examples

Sample outlines: Design study II
• data and task abstractions
– analyze your domain problem
  – build up ideas
  – high through low level: what before how
  – paper level
  – motivation why should I care
  – overview what did you do
  – details how did you do it
• technique level
– overview then details
– sometimes subsection or paragraph level

Sample outlines: Other types
• see page for implementation project types

Sample outlines: T echnique (diffs)
• Abstract, introduction (same as above)
• Related/Work
– big focus on similar solutions, some discussion of similar problems (same task/datasets combo)
• Data and Task Abstractions
– more emphasis on characterizing the breadth of possible uses
– describe your solution idiom (visual encoding and interaction)
– analysis according to book framework (how)
– justify your design choices with respect to alternatives
– if significant algorithm work, discuss algorithms and data structures

Report structure: General
• low level necessary but not sufficient
• mid-level: order of explanations
– build up ideas
– high through low level: what before how
• paper level
– motivation why should I care
– overview what did you do
– details how did you do it
• section level
– overview then details
– sometimes subsection or paragraph level

Sample outlines: Survey (diffs)
• Abstract (same as above)
• Introduction
– discuss the scope of what you’re covering, why it’s interesting/rational/important compared to visualization as a whole
• Related/Work
– only previous surveys
– focus on how your work is similar to vs different from them: especially work coverage
• Main
– break up sections based on your own synthesis of themes of work covered
• Discussion
– relevant vocabulary/ideas, your own background connection
• Data/Task Abstraction, Related Work (same as above)
• Methods & Tools
– how it is prescribed/normally analyzed
– explain what idioms you chose and justify those choices; same for tools
• Analysis
– present results of your visual data analysis, including screenshots of tools in action
– you might want a Background section as the start if domain-focused survey
– when there’s important vocabulary/idioms to establish before diving into non-discussion
• strengths/weaknesses of your approach (idioms and tools)
• can be interleaved or in separate section
• Discussion / Future Work, Conclusions, Bibliography (same as above)

Sample outlines: Survey (diffs)
• Abstract (same as above)
• Introduction
– discuss background section as the start if domain-focused survey
– when there’s important vocabulary/idioms to establish before diving into non-discussion
• analyze visualizations presented in these papers in terms of what/why/how framework
• main
– break up sections based on your own synthesis of themes of work covered
• discussion
– relevant vocabulary/ideas, your own background connection

Sample outlines: Analysis (diffs)
• Abstract (same as above)
• Introduction
– discuss background section as the start if domain-focused survey
• Methods & Tools
– how it is prescribed/normally analyzed
– explain what idioms you chose and justify those choices; same for tools
• Analysis
– present results of your visual data analysis, including screenshots of tools in action
– you might want a Background section as the start if domain-focused survey
– when there’s important vocabulary/idioms to establish before diving into non-discussion
• strengths/weaknesses of your approach (idioms and tools)
– can be interleaved or in separate section
• Discussion / Future Work, Conclusions, Bibliography (same as above)

Sample outlines: Design study III
• implementation
– medium-level implementation description
– specific details of what you wrote vs what existing library/cookie/components do
• results of what did what work
• findings
– include findings of your analysis (eg deployment to target users, computational benchmarks)
– if necessary, should be pg (lossless compression) not png (lossy compression)
• discussion and future work
– reflect on your approach: strengths, weaknesses, limitations
– lessons learned: what do you know now that you didn’t when you started?
– future work: what would you do if you had more time?

Sample outlines: Design study IV
• conclusions
– summarize what you’ve done
• different than abstract since reader has seen all the details
• bibliography
– review reading, review writing, conference talks
• more emphasis on characterizing the breadth of possible uses
– big focus on similar solutions, some discussion of similar problems (same task/datasets combo)
• Data and Task Abstractions
– more emphasis on characterizing the breadth of possible uses
– describe your solution idiom (visual encoding and interaction)
– analysis according to book framework (how)
– justify your design choices with respect to alternatives
– if significant algorithm work, discuss algorithms and data structures

Sample outlines: Other types
• see page for implementation project types

Sample outlines: Final Papers & Presentations
• www.cs.ubc.ca/~tmm/courses/547-19/projectdesc.html#examp
• abstract
– concisely summarize your project
– do not include citations
• introduction
– high starting point: overview of what you’ll do
– low level: necessary but not sufficient
• mid-level: order of explanations
– build up ideas
– high through low level: what before how
• paper level
– motivation why should I care
– overview what did you do
– details how did you do it
• section level
– overview then details
– sometimes subsection or paragraph level

Sample outlines: Design study I
• implementation
– medium-level implementation description
– specific details of what you wrote vs what existing library/cookie/components do
• results
– include findings of your analysis (eg deployment to target users, computational benchmarks)
• discussion and future work
– reflect on your approach: strengths, weaknesses, limitations
– lessons learned: what do you know now that you didn’t when you started?
– future work: what would you do if you had more time?

Sample outlines: Technical (diffs)
• Abstract, introduction (same as above)
• Related/Work
– big focus on similar solutions, some discussion of similar problems (same task/datasets combo)
• Data and Task Abstractions
– more emphasis on characterizing the breadth of possible uses
– big focus on similar solutions, some discussion of similar problems (same task/datasets combo)
• Methods & Tools
– how it is prescribed/normally analyzed
– explain what idioms you chose and justify those choices; same for tools
• Analysis
– present results of your visual data analysis, including screenshots of tools in action
– you might want a Background section as the start if domain-focused survey
– when there’s important vocabulary/idioms to establish before diving into non-discussion
• strengths/weaknesses of your approach (idioms and tools)
– can be interleaved or in separate section
• Discussion / Future Work, Conclusions, Bibliography (same as above)

Sample outlines: Technical (diffs)
• Abstract, introduction (same as above)
• Related/Work
– big focus on similar solutions, some discussion of similar problems (same task/datasets combo)
• Data and Task Abstractions
– more emphasis on characterizing the breadth of possible uses
– big focus on similar solutions, some discussion of similar problems (same task/datasets combo)
• Methods & Tools
– how it is prescribed/normally analyzed
– explain what idioms you chose and justify those choices; same for tools
• Analysis
– present results of your visual data analysis, including screenshots of tools in action
– you might want a Background section as the start if domain-focused survey
– when there’s important vocabulary/idioms to establish before diving into non-discussion
• strengths/weaknesses of your approach (idioms and tools)
– can be interleaved or in separate section
• Discussion / Future Work, Conclusions, Bibliography (same as above)
Contributions in research papers

• what are your research contributions?
– what can we do that wasn't possible in reading of previous work
– goal is clarity, not overselling (limitations typically later, in discussion section)
– don't necessarily expect your code compiles on my machine
• encouraged but not required
– submit live demo URL
– open-source your code (if so, fine to just send me that URL)
– submit supporting video
– with or without voiceover
– very nice to have tutorial, software bits/notes doesn't last forever!
• can be same or different from what you show in final presentation

Code / Video

• required submit your code
– so I can see what you've done, but I will not post
  – include README file as root with brief roadmap/overview of organization
  – which parts are your code vs libraries
  – how to compile and run
– I do not necessarily expect your code compiles on my machine
• encouraged but not required
– submit live demo URL
– open-source your code (if so, fine to just send me that URL)
– submit supporting video
– with or without voiceover
– very nice to have tutorial, software bits/notes doesn't last forever!
• can be same or different from what you show in final presentation

Showcase image

• showcase image for projects page
  – 300x300 image
  – call it showcase.png or showcase.jpg

Logistics

• Assignments: Final Presentations on Canvas
  – upload due 1:30 for final presentations if using my laptop
  – upload live demo URL video

Final presentations

• context
  – CS department will be invited, also feel free to invite others
  – refreshments will be served, two short breaks
  – order alphabetical by first name
• code freeze
  – no additional work on project after presentation deadline
  – additional three days to get it all written down coherently for final report

Final presentations: Tue Dec 13 3-7 (!) FSC 2300A

• length (19 projects)
  – 14 min for 3-person teams, 13 min for 2-person teams, 10 min for 1-person teams
  – includes questions for 1 min (brief questions only)
  – session structure
  – order alphabetically by first name as on project page (with or without)
  – demo or video encouraged
  – don’t cram in so much content that can’t explain why/what/how
  – diverged from original goals, in retrospect

• Timing:
  – submit live demo URL
  – open-source your code (if so, fine to just send me that URL)
  – submit supporting video
  – with or without voiceover
  – very nice to have tutorial, software bits/notes doesn’t last forever!
• can be same or different from what you show in final presentation

Process & Pitfalls for InfoVis Papers

Idiom pitfalls

• Unjustified Visual Encoding
  – should justify why visual encoding design choices appropriate for problem
  – preconceived common statement of problem and encoding!
• Hammer In Search of Nail
  – should characterize capabilities of new technique if proposed in paper
• Color Cacophony
  – avoid blatant disregard for basic color perception issues
  – huge areas of highly saturated color
  – categorical color coding for 15+ category levels
  – nodogram without luminance differences
  – encoding 3 separate attributes with RGB
• Rainbows Just Like In The Sky
  – avoid hue for ordered attributes, perceptual nonlinearity along rainbow gradient

Contributions in research papers

• what are your research contributions?
– what can we do that wasn’t possible before? how can we do something better than before?
– how do we know that was unique or unclear before?
• determines everything
– how to frame message which details worth including
  – often not obvious
  – diverged from original goals, in retrospect
  – state these explicitly and clearly in the introduction
  – don’t hope reviewer or reader will fill them in for you
  – don’t leave unclear should be obvious after close reading of previous work
– goal is clarity, not overelaborating (limitations typically later, in discussion section)

Later pitfalls: Tactics

• Stealth Contributions
  – definitely not appropriate
  – think it’s your job to tell reader explicitly!
  – consider carefully, often different from original project goals
• I Am So Unique
  – don’t ignore previous work
  – from high-level problems and with similar solutions
• Enumeration Without Justification
  – “X did it” 70%
  – must say why previous work doesn’t solve your problem
  – what limitations of their does your approach fill?
• I Am Utterly Perfect
  – no you’re not; discussion of limitations makes paper stronger!

Later pitfalls: Results

• Unしたらted By Time
  – choose level of detail for performance numbers
  – detailed graphs for technical papers, high-level for design & eval papers
• Straw Man Comparison
  – compare appropriately against state-of-the-art algorithms
  – mutually exhaustive and mutually exclusive (run benchmarks yourself, all on same machine)
• Tiny Toy Datasets
  – compare against state-of-the-art datasets sizes for technique (small ok for eval)
• But My Friends Liked It
  – asking lameness of course not convincing to general audience is domain experts
• Unjustified Tasks
  – use ecologically valid user study tasks convincing abstraction of real-world use

Later pitfalls: Style

• Deadly Detail Dump
  – think only of after what and why; provide high-level framing before low-level detail
• Story-Free Captions
  – opt for flow-through pictures skimming
• My Picture Speaks For Itself
  – explicitly walk them through images with discussion
• Grammar Is Optional
  – good low-level flow is necessary (but not sufficient), native speaker check good in ESL
• Mistakes Were Made
  – don’t use passive voice, leaves ambiguity about actor
  – your research contribution or done by others?

Final pitfalls: Tactics

• Stealth Contributions
  – don’t leave implicit, it’s your job to tell reader explicitly!
  – consider carefully, often different from original project goals

Final pitfalls: Style

• deadly detail dump
  – explain only of after what and why; provide high-level framing before low-level detail
• story-free captions
  – opt for flow-through pictures skimming
• my picture speaks for itself
  – explicitly walk them through images with discussion
• grammar is optional
  – good low-level flow is necessary (but not sufficient), native speaker check good in ESL
• mistakes were made
  – don’t use passive voice, leaves ambiguity about actor
  – your research contribution or done by others?
When and how will this storm hit visualization?

• they're ahead of us
  – they have some paper retractions
  – they agonize about difficulty of getting failure-to-replicate papers accepted
  – we hardly ever even try to do such work.
  – they are a much older field
    – are younger, might our power hierarchies thus be less entrenched?…
    – they have higher profile
    – we don't have as rich a fabric of blogs as major drivers of discussion
  – crosscutting traditional power hierarchies
  – we have far fewer active bloggers

• replication crisis was focus of BELIV 2018 workshop at IEEE VIS
  – evaluation and BEyond - methodoLogIcal approaches for Visualization
    – http://beliv.cs.univie.ac.at/