Interactive Visual Data Analysis

11 – Data navigation and interactive lenses

Objectives

- How can we navigate data beyond zooming: Learn additional data navigation techniques
- How can we adapt visualizations dynamically: Learn about visualization adjustment via interactive lenses

Overview

Last lecture:

- Basic techniques

Today:

- Additional data navigation techniques
  - Beyond zooming in two dimensions
  - Multi-scale input for precise navigation
- Visualization adjustment
  - Interactive lenses

Beyond zooming in two dimensions

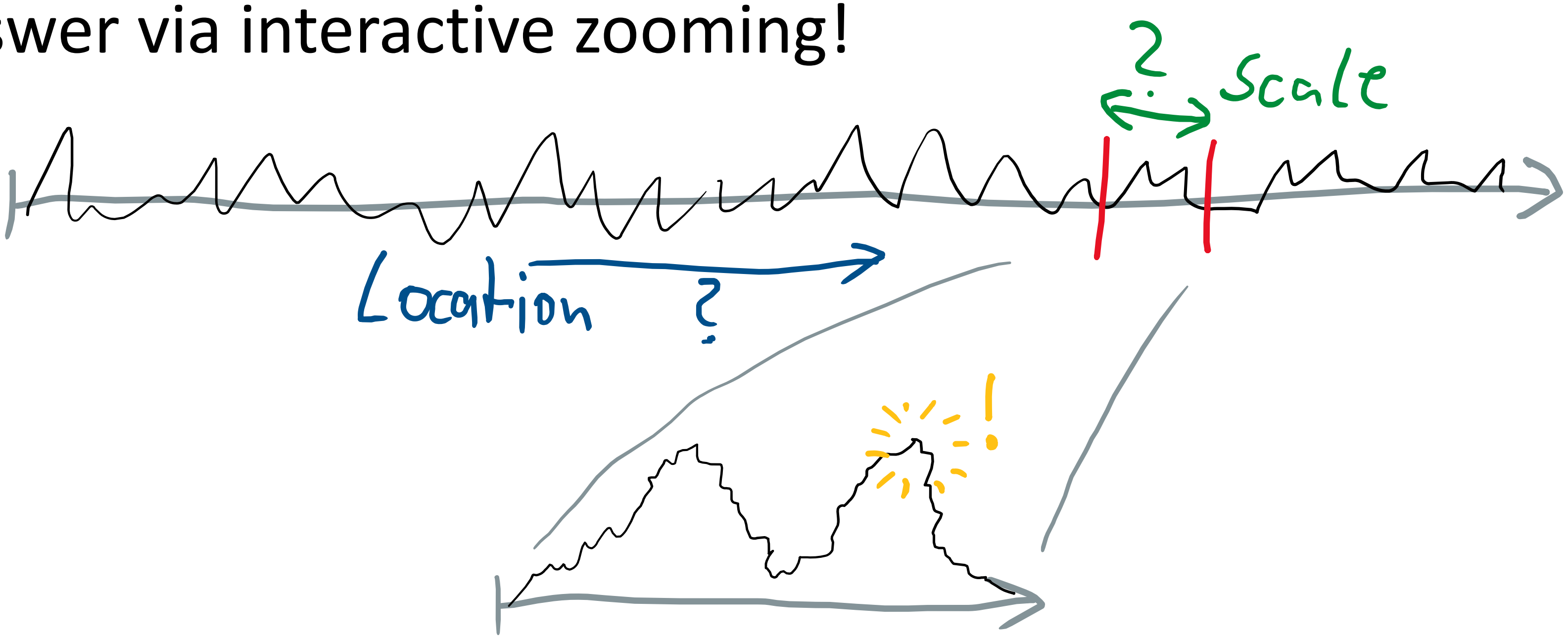
- So far, we considered zooming as a means to support the navigation of data shown as a 2D visual representation
- Yet, zooming is a general concept applicable beyond just 2D

- We look at two examples
  - **1D zooming** for univariate temporal data
  - ***n*D zooming** for multivariate (temporal) data

Beyond zooming in two dimensions

General case of zooming

- **Where (location)** and **how much (scale)** of a data variable should be studied to find interesting patterns?
- Find answer via interactive zooming!

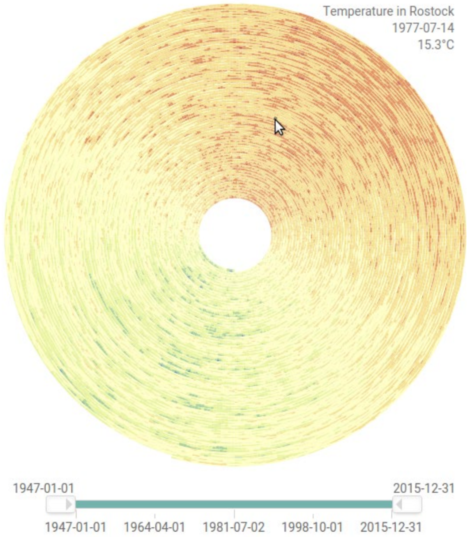


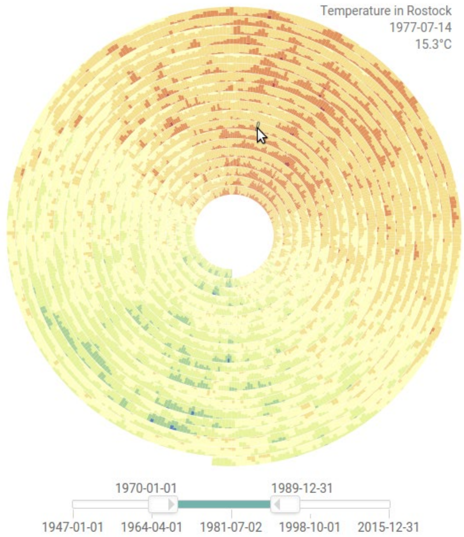
Beyond zooming in two dimensions

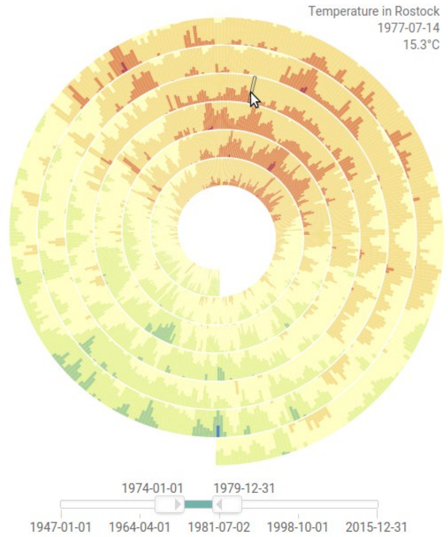
Example: 1D zooming for univariate temporal data

- Explore temperature data on a spiral display
- Interactive **range slider** for zooming and panning along the 1D time axis

Temporal data visualization was the topic of Lecture 6.





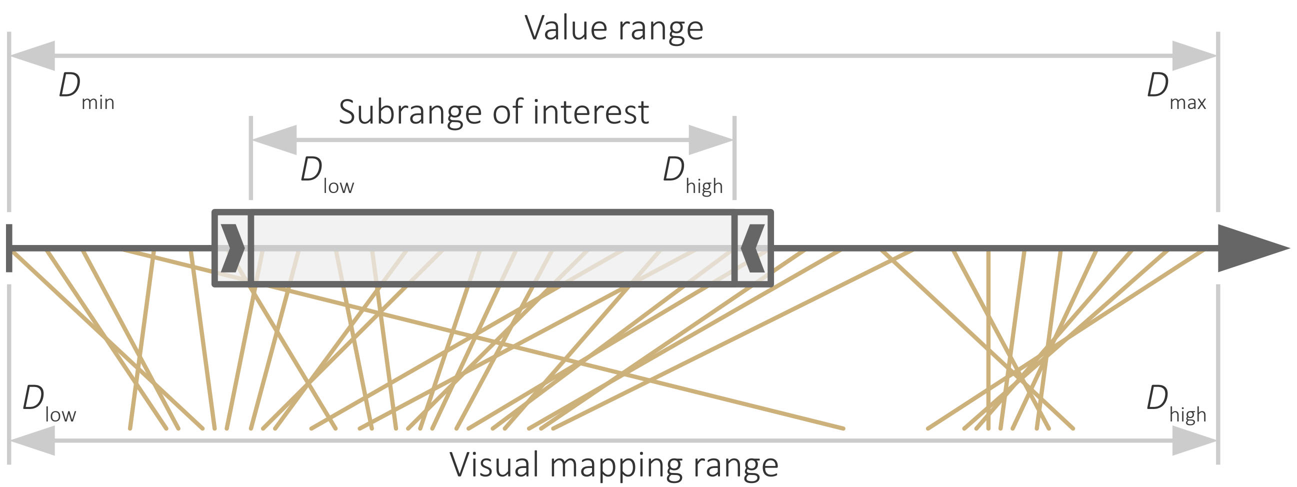


https://vcg.informatik.uni-rostock.de/~ct/software/EnhancedSpiral.js/index.html

Beyond zooming in two dimensions

General case of *n*D zooming for multivariate data

- Question: How to zoom *n*D data?
- Answer: Combine multiple 1D **zoomable axes**

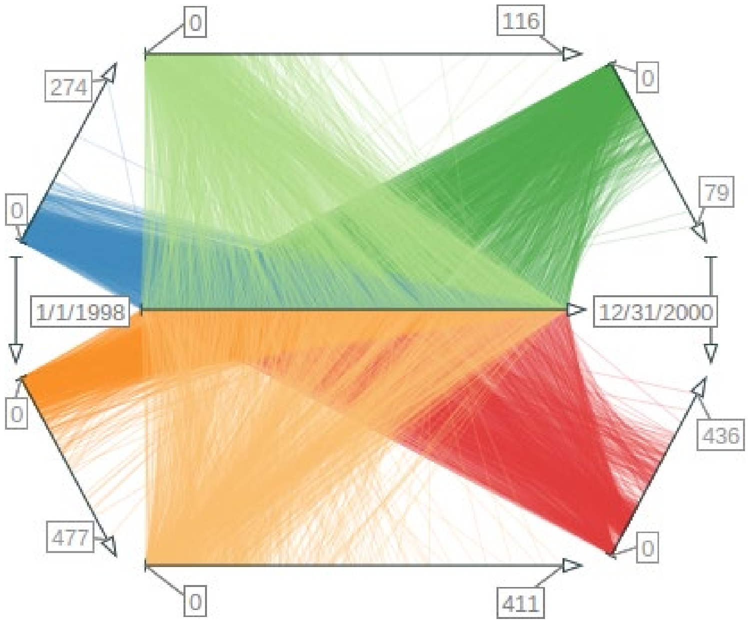


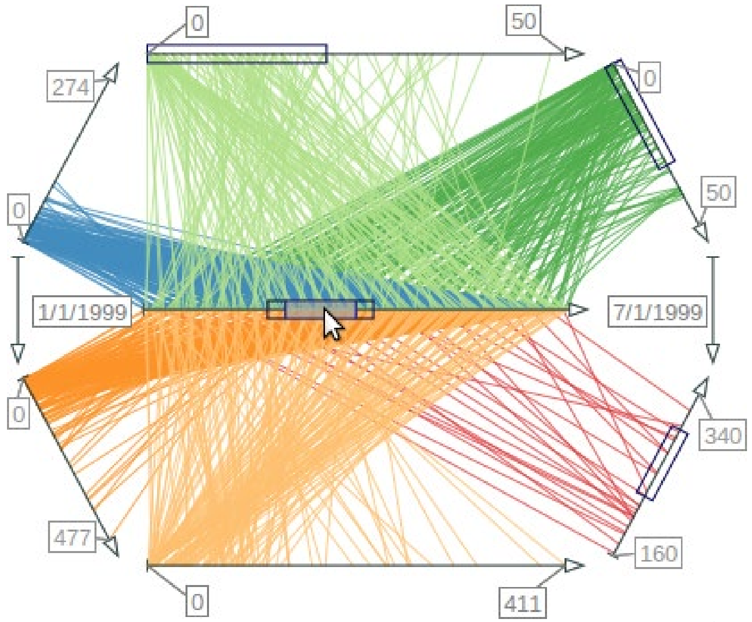
- Think about: What about interaction costs, physical and mental?

Beyond zooming in two dimensions

Example: *n*D zooming for the TimeWheel

- Central axis for time, radial axes for time-dependent variables
- Each axis can be zoomed independently or in a linked-fashion → nD zoom





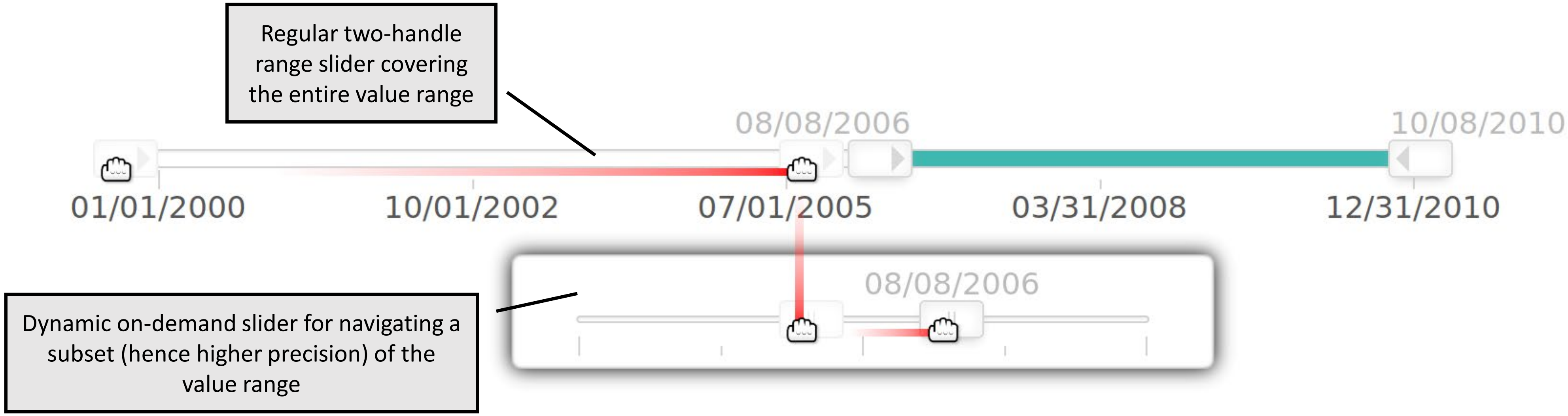
Multi-scale input for precise navigation

- Problem: When navigating larger value ranges, we may have **insufficient interaction precision** due to too-low pixel resolution
- Example:
  - Time series with 25.000 days would require 25.000 pixels for sufficient precision
  - Realistically, if we assume a slider with 1.000 pixels (i.e., 25.000 days are mapped to 1.000 pixels), each pixel corresponds to 25 days.
  - Drag interaction by 1 pixel would navigate by 25 days, essentially skipping 24 days
  - There is no way to access the skipped days 😒
- Think about: How to navigate precisely when resolution is limited?
- Solution: **Interaction** not only on single scale, but **on two or more scales**

Multi-scale input for precise navigation

Dual-scale slider

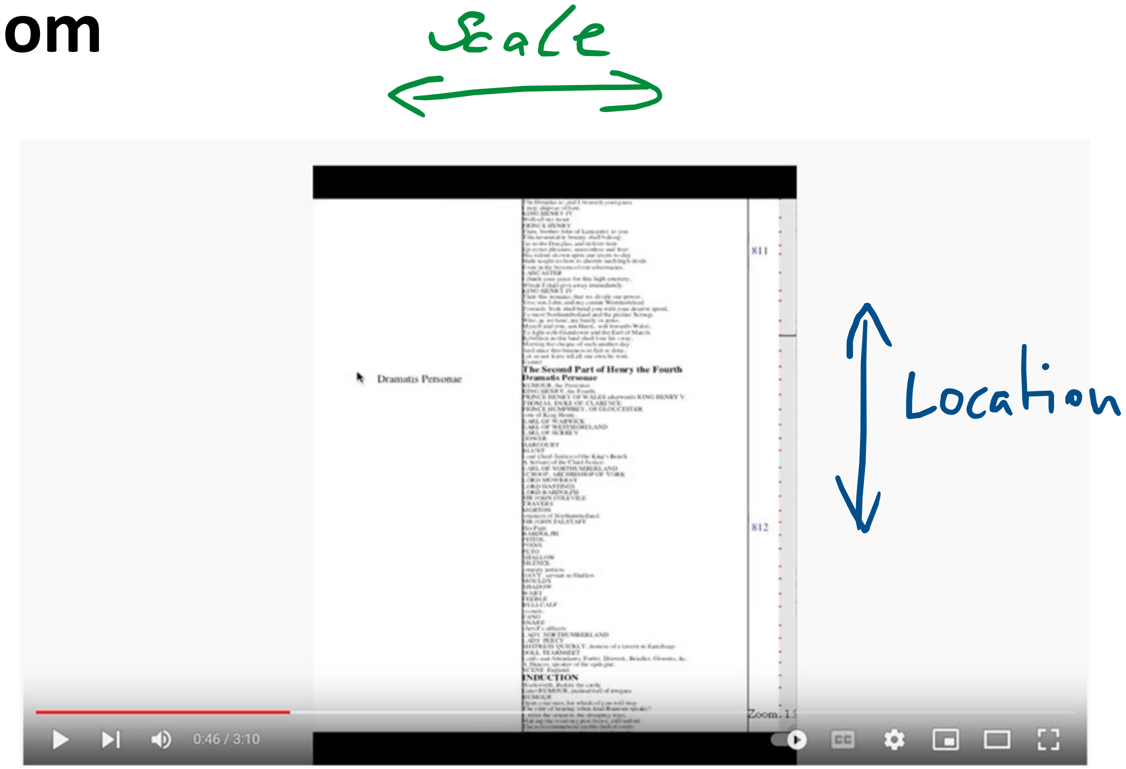
- Regular slider for coarse and fast navigation
- On-demand high-precision slider for fine but slow navigation



Multi-scale input for precise navigation

Multi-scale navigation with OrthoZoom

- Zooming via drag operation
- Drag semantics depend on direction
    - Vertical direction determines location in the data (where)
    - Horizontal direction determines how much data (scale)



https://www.youtube.com/watch?v=fwz04BNRQQQ

Overview

Last lecture:

- Basic techniques

Today:

- Additional data navigation techniques
  - Beyond zooming in two dimensions
  - Multi-scale input for precise navigation
- Visualization adjustment
  - Interactive lenses

Motivation

We learned about **interaction intents** in Lecture 9.

Interaction intents (Yi et al., 2007)

So far

- Fundamental picking, interactive selection, and accentuation
  - *Mark* something as interesting
  - *Show me* something conditionally
- Navigating zoomable visualizations
  - *Show me* something else
  - *Show me* more or less detail

Motivation

Interaction intents (Yi et al., 2007)

Now

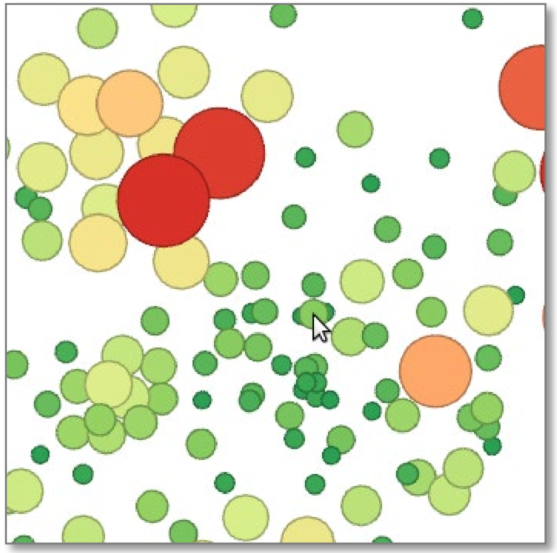
- Interactive lenses
  - *Show me* a different arrangement
  - *Show me* a different representation

Motivation

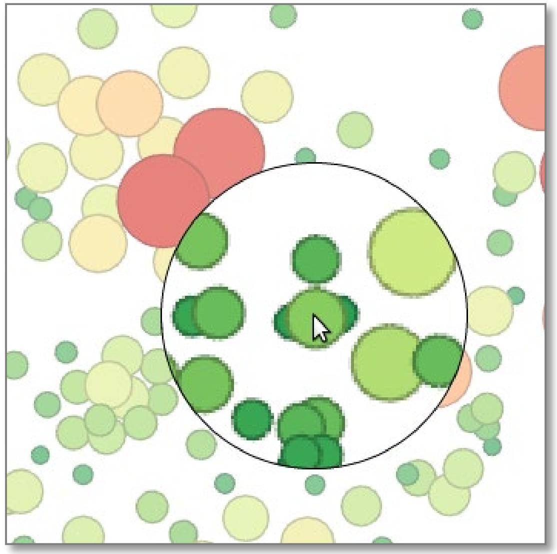
- Standard interaction
  - Performed using graphical user interface
  - Leads to global and permanent change of visual representation
- Implications
  - Action and response spatially separated → Effect can be difficult to comprehend (What did just happen?)
  - Global effect → Context can be difficult to maintain (How did it look like before?)
  - Permanent effect → Difficult to return to previous view (How can I undo the change?)
- Elegant alternative: **Interactive lenses**

Interactive Lenses

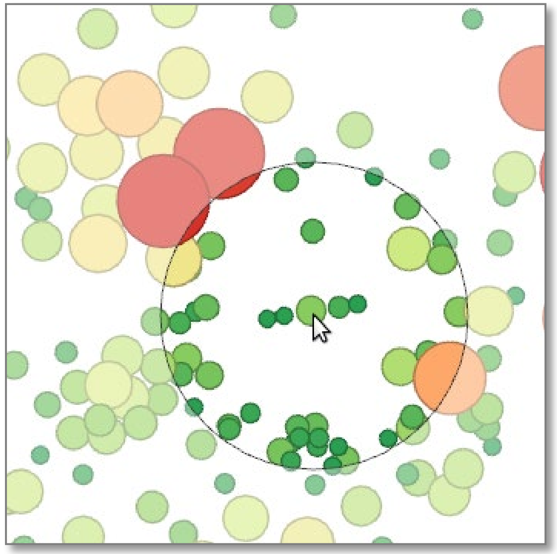
A first example: Access details in visual representations



Regular visualization



Simple magnification



Fisheye distortion

Interactive Lenses

Next, we look at

- Definition
- Conceptual model
- Adjustable properties

Interactive Lenses

Definition

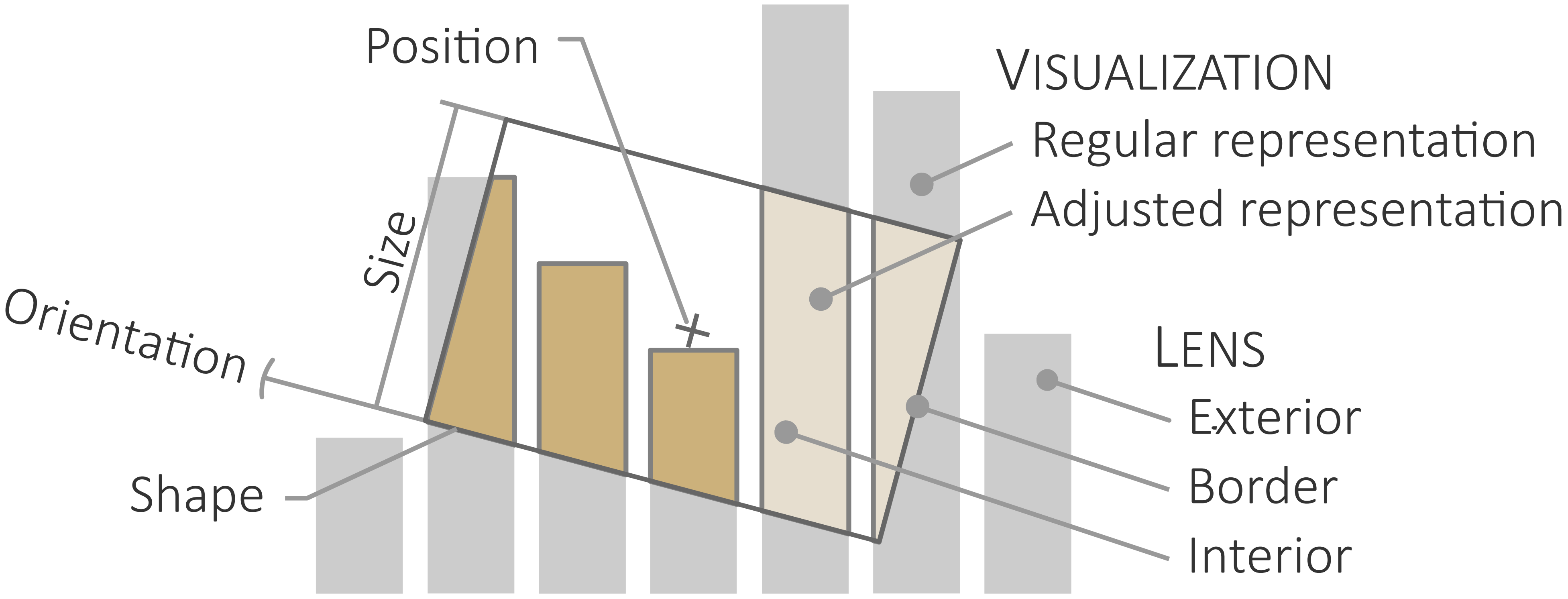
Lens techniques were originally introduced by Bier et al. (1993) under the label **Magic Lenses**.

> “An interactive lens is a lightweight tool to solve a localized visualization problem by temporarily altering a selected part of the visual representation of the data.
>
> — Tominski et al., 2017

> “Lenses are lightweight exploration tools that can be added to a visualization on demand. […] A key characteristic is that lenses produce *local and transient* changes in the visualization. That is, the visual representation is adjusted only in selected parts and its original state is restored once the lens is dismissed.
>
> — Tominski & Schumann, 2020

Interactive Lenses

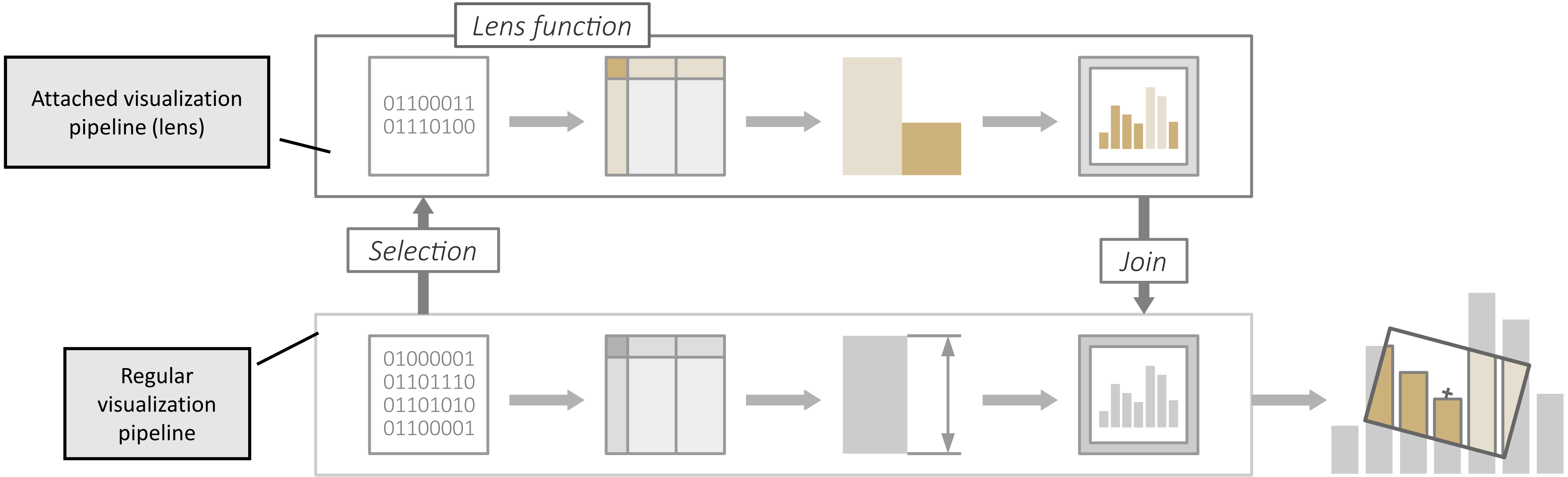
Conceptual model



Interactive Lenses

Conceptual model

- Interactive lenses can be modeled as additional visualization pipeline attached to the regular visualization pipeline



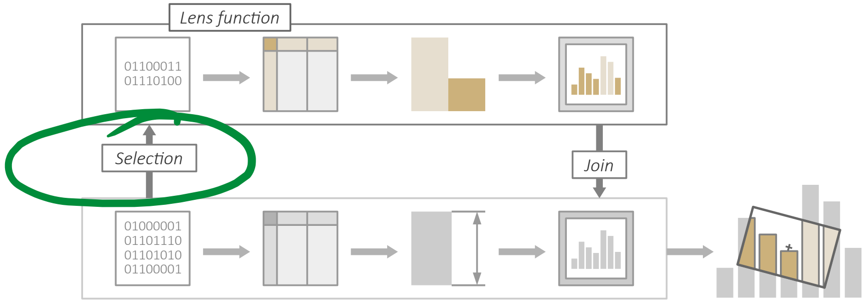
Interactive Lenses

Conceptual model

Lens techniques consist of three main ingredients

- Input data: **Selection** determines what data are to be affected by the lens
- Data processing: **Lens function** defines how the visual representation of the selected data looks like
- Result output: **Join** merges the changed visual representation with the base visualization

Interactive Lenses



Selection: What is to be affected

- Key to the “local” nature of the interaction:
  Selection → smaller data subset → local effect

- Defines part of the visual representation to be changed
- Typically corresponds to content under lens
- Selected content can be from any stage of the visualization pipeline (pixels, geometry, or underlying data)
- Based on selected content, the lens function computes a lens effect

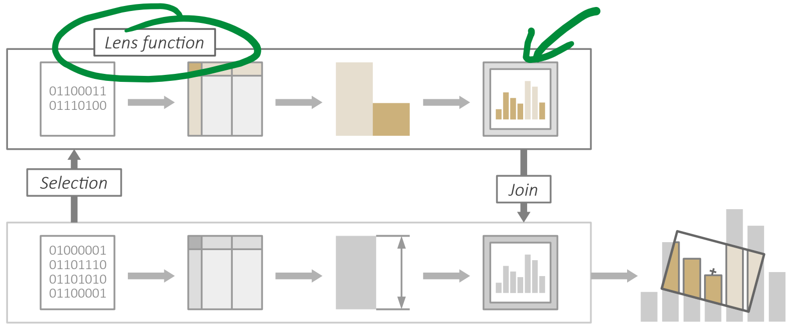
Interactive Lenses

Lens function: How is the visual representation changed

- Processes selected content to generate lens effect
- Depending on stage where selection took place, only particular processing steps needed
- Selection usually much smaller than entire dataset: Advanced and potentially costly operations possible on reduced subset

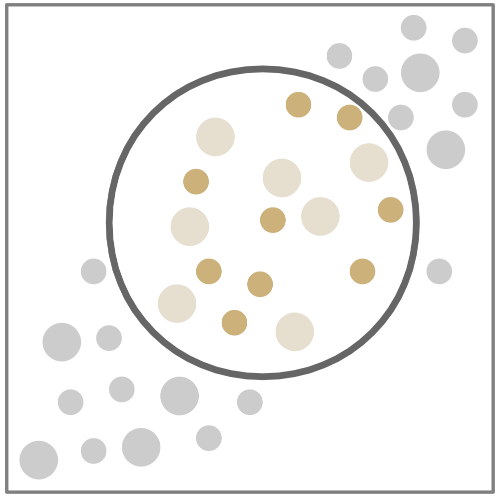
- Large variety of lens function in literature
  → **Three basic lens effects**

Interactive Lenses

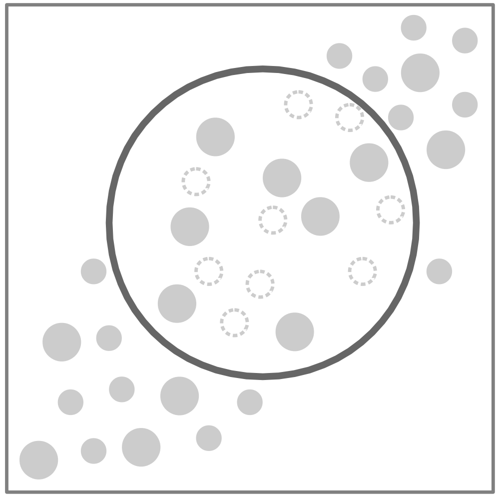


Lens effects: What types of changes are possible

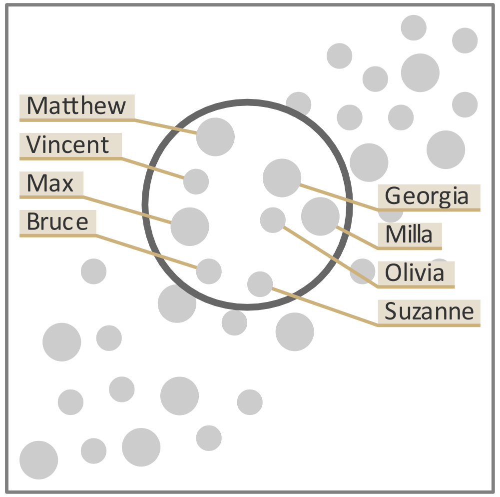
- Alteration: Change existing content
- Suppression: Remove existing content
- Enrichment: Add new content



Alter: Change color



Suppress: Filter dots

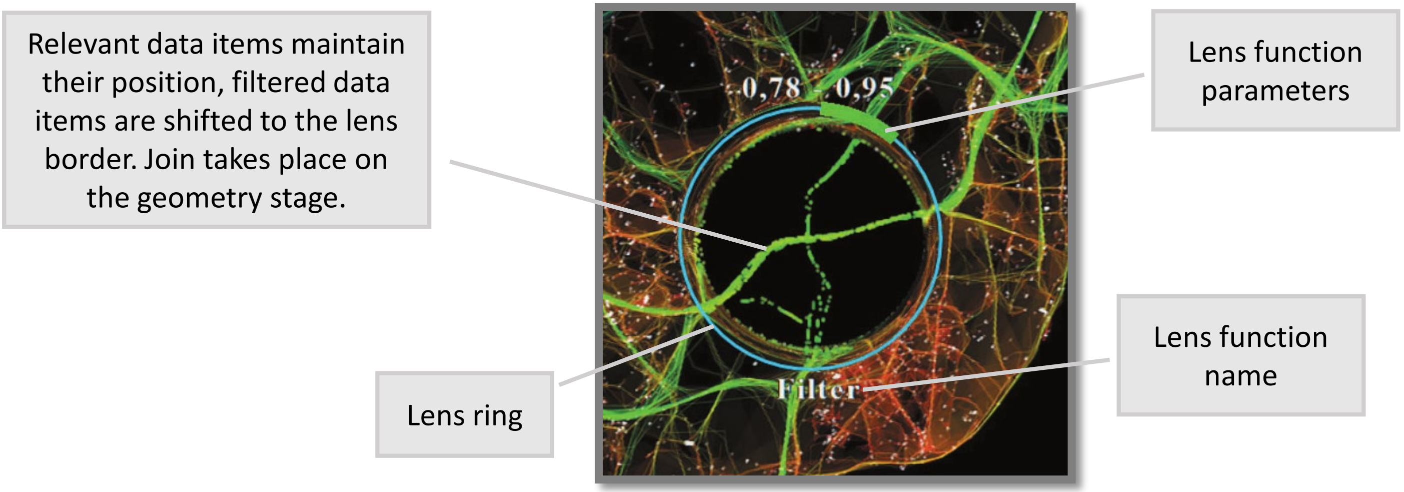


Enrich: Add labels

Interactive Lenses

Join: How to bring lens and visualization together

- Join lens effect with regular visualization
- Possible at stages of pixels, geometry, or underlying data
- Additional visuals: UI, lens circle, dim background



Hurter et al. (2011)

Interactive Lenses

Next, we look at

- Definition
- Conceptual model
- Adjustable properties

Interactive Lenses

Properties

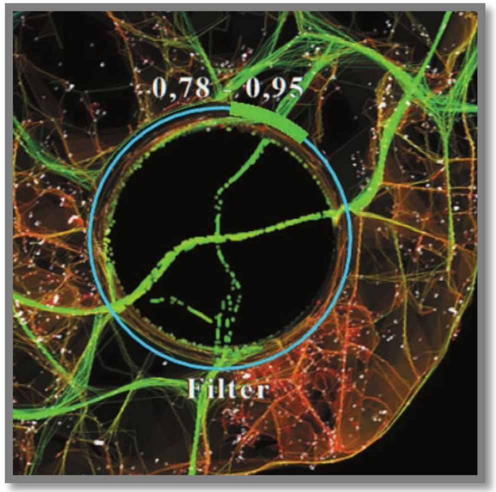
- Spatial selection → Geometry
  - Position & size, shape, orientation
- Lens function → Parameters
  - Depend on function
  - Example: Magnification factor, sampling rate, …

Interactive Lenses

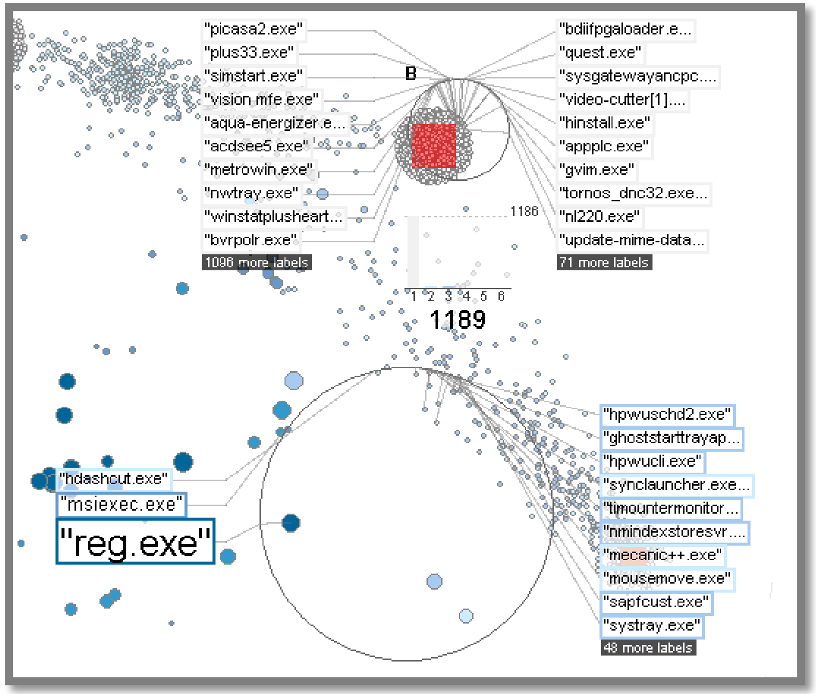
Position & Size

- Usually interactive via natural drag gestures
- May also be set automatically

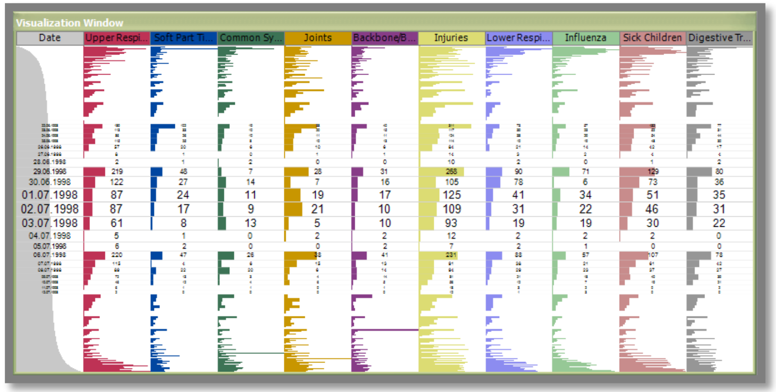
MoleView,
Hurter et al. (2011)
(interactive size)



Extended Excentric Labeling,
Bertini et al. (2009)
(automatic size)



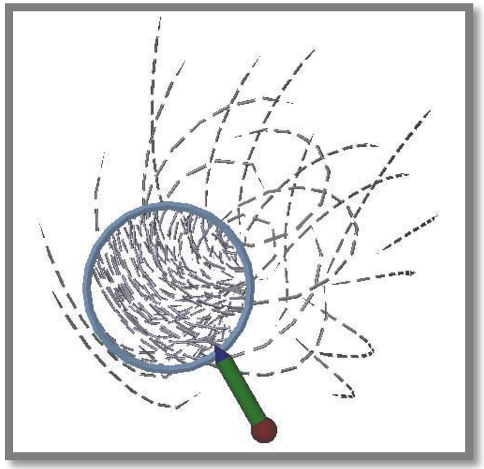
Automatic table lens,
Tominski (2011)
(automatic position)



Interactive Lenses

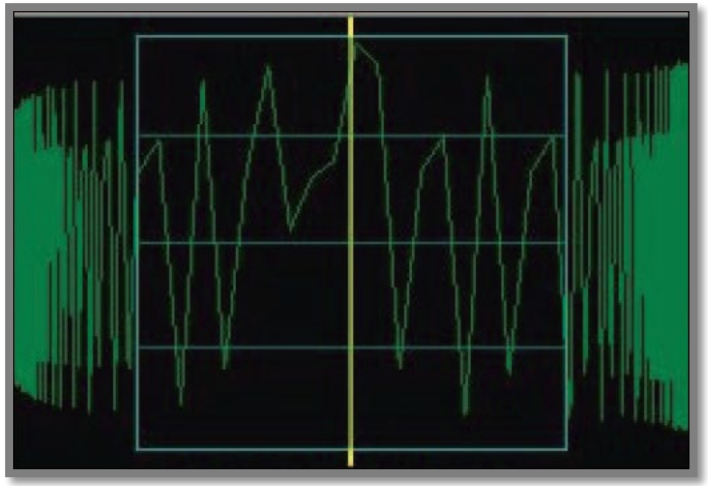
Shape

- Circular
- Rectangular
- Self-adapting

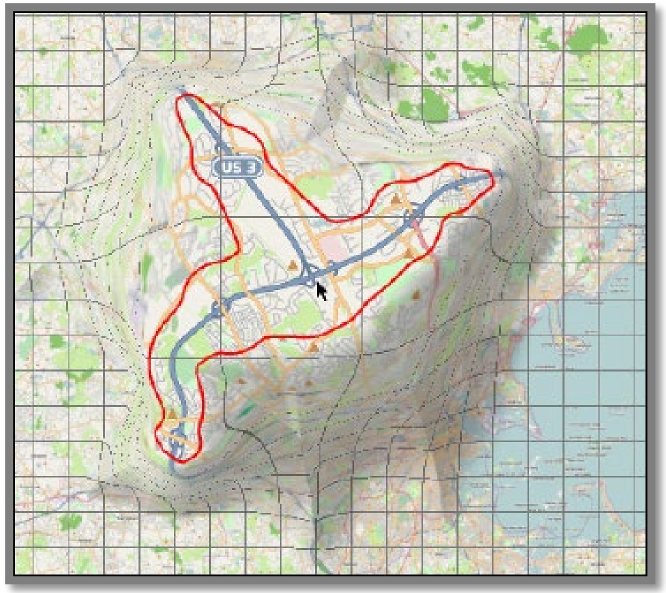


Lens for Flow Vis.,
[Fuhrmann & Gröller (1998)]
(circular)

JellyLens,
[Pindat et al. (2012)]
(self-adapting)



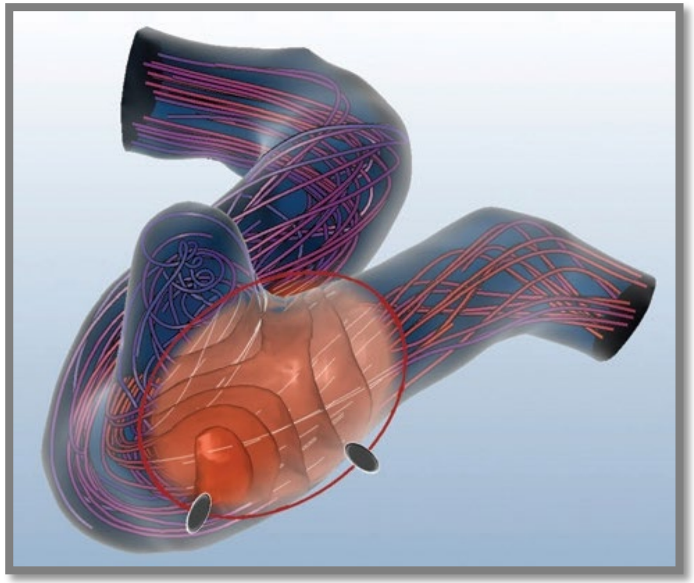
SignalLens,
[Kincaid (2010)]
(rectangular)



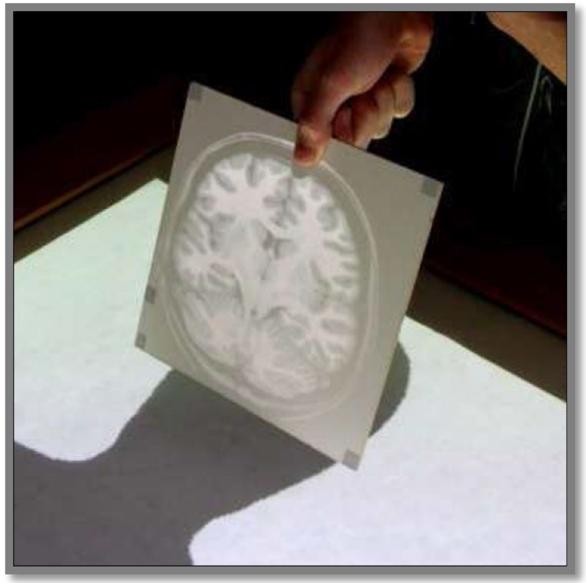
Interactive Lenses

Orientation

- Useful for fine-tuning selection
- Opens up new applications of lenses



FlowLens,
Gasteiger et al. (2011)



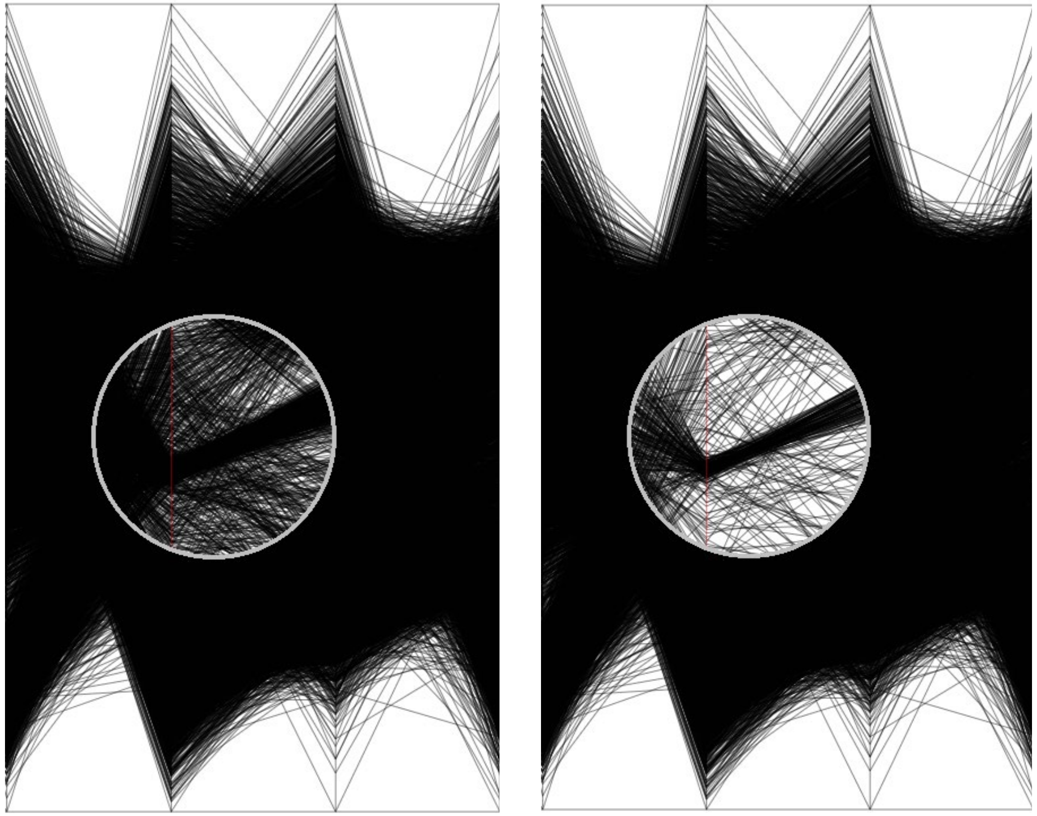
PaperLens,
Spindler et al. (2009)

Interactive Lenses

Parameters of lens function

- Magnification factor
- Sampling rate
- …

Sampling Lens,
Ellis & Dix (2006)
(sampling rate)



Interactive Lenses

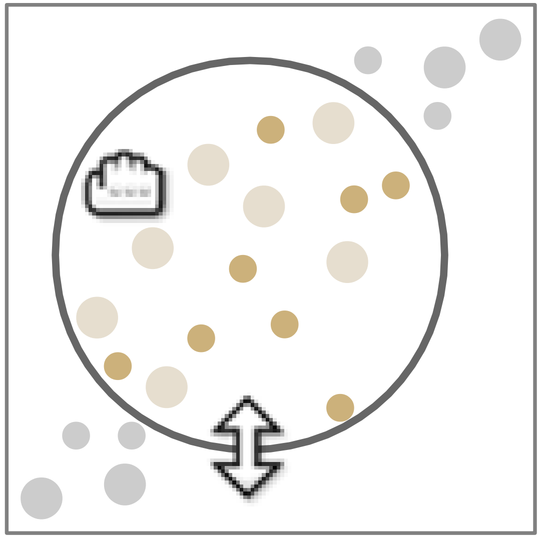
Interacting with lenses

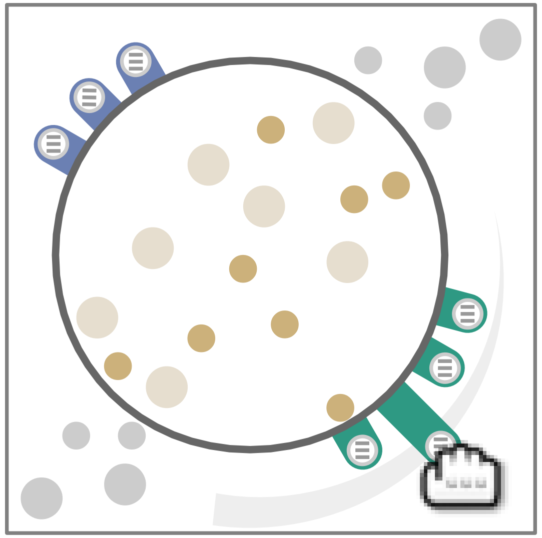
> “[…] the fundamental problem is how you provide the user a quick and easy way to: Position the lens, work through the lens, and (possibly) parameterize the lens.
>
> — Maureen Stone, 2014

Interactive Lenses

Interacting with lenses

- Positioning and resizing via natural drag gestures
- Parameters adjustments via dedicated on-lens GUI elements





- Think about: How would you interact with data items under a lens?

Interactive Lenses

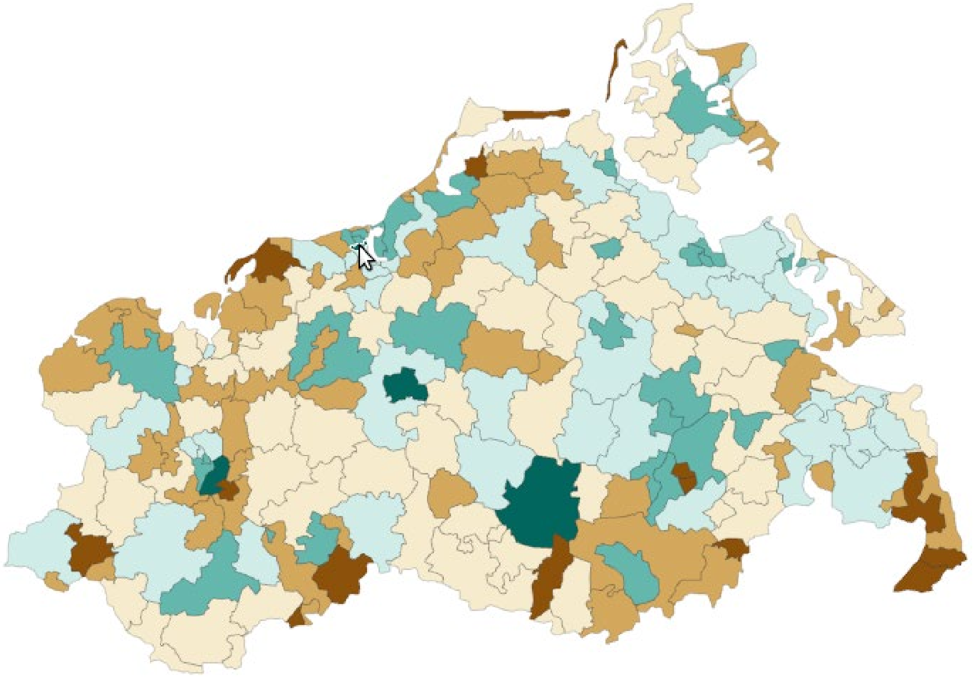
Lenses in action

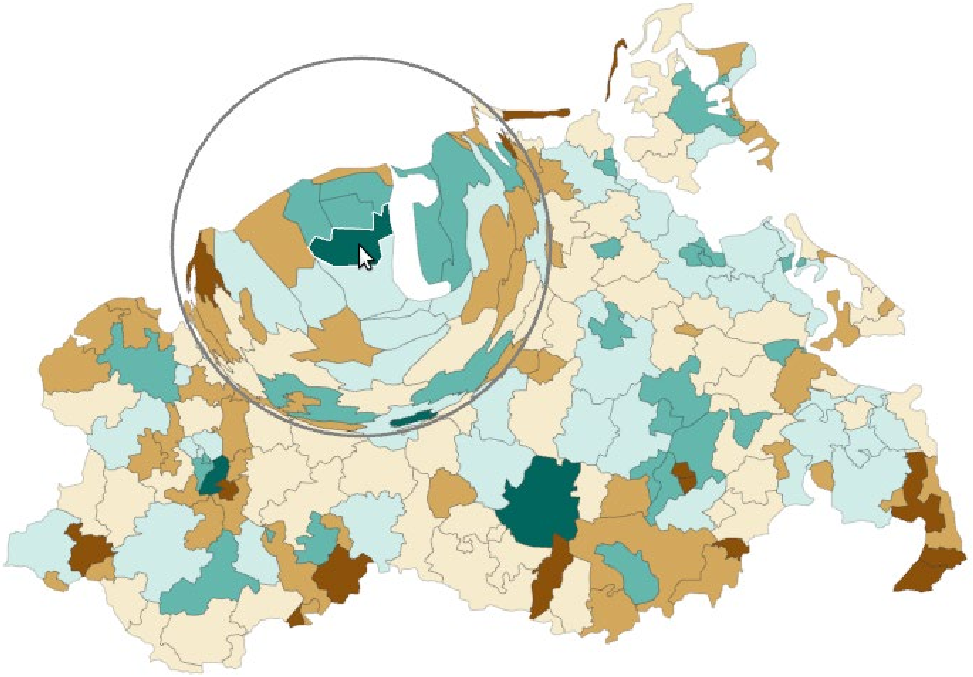
- Exploring details: Fish-eye lens
- Exploring structural relationships: Graph lenses
- Exploring temporal aspects of movements: Time lens
- Semi-automatic graph editing: Edit lens

Interactive Lenses

Exploring details: Fish-eye lens

We already talked about **fisheye distortion** in Lecture 4.





Problem: Identify color under cursor w/o zooming

Fish-eye lens: Local magnification around cursor

Interactive Lenses

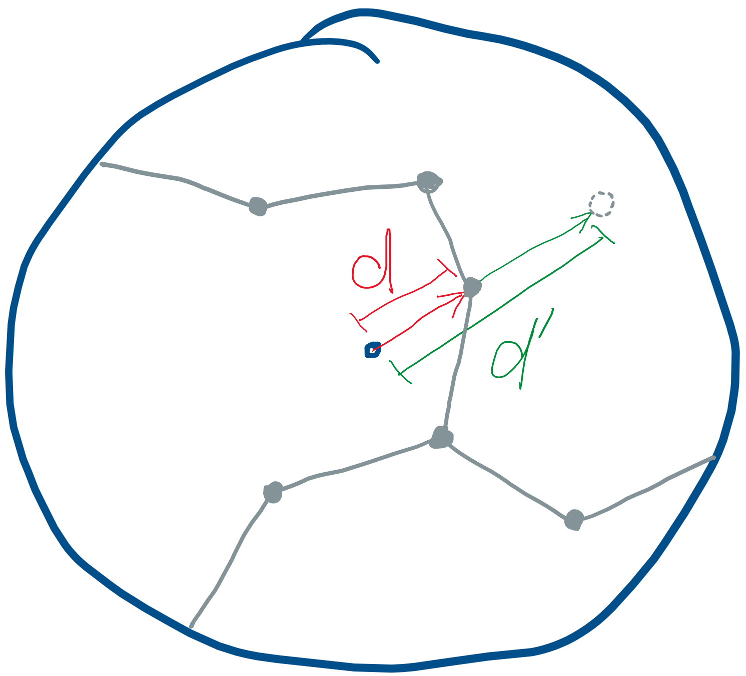
Exploring details: Fish-eye lens

- Fisheye distortion (Sarkar and Brown, 1994)
  - Selection: Points of the geometric description of the geographic regions
  - Lens function and effect:
    - Processing: Translate points to new position (details on next slide)
    - Effect: Magnified geographic regions around the lens center
  - Join:
    - Override original point positions with new ones
    - Superimpose lens ring

Interactive Lenses

Exploring details: Fish-eye lens

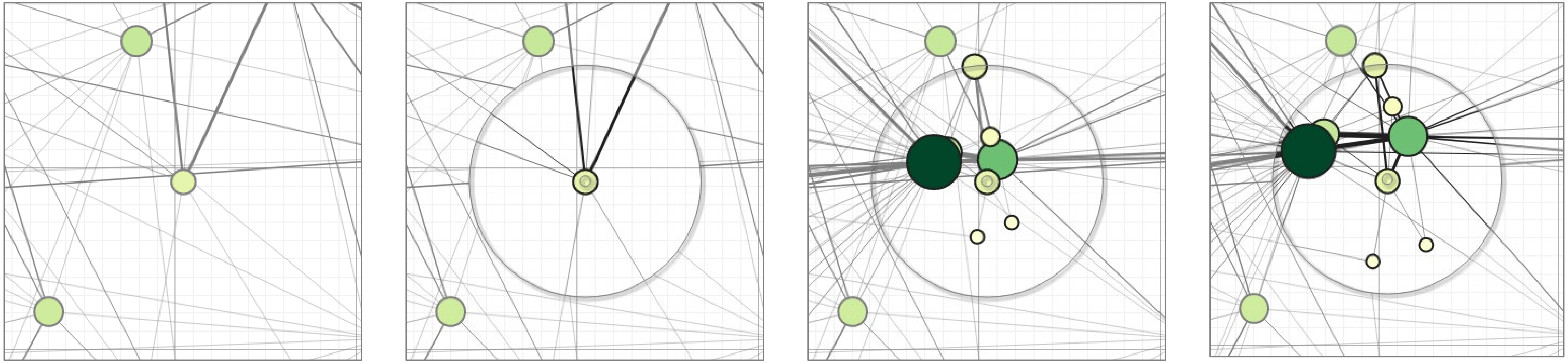
- Fisheye distortion (Sarkar and Brown, 1994)
  1. Select all geometric points with distance to lens center $d < d_{max}$ (the lens radius)
  2. For each selected point, push it toward the lens boundary via parametric mapping of original distance d to new distance d' (magnification factor m defines how far points are pushed)
- Think about: Is this simple approach really enough to create a correct magnification?



$$d' = d_{max} \cdot \frac{(m+1) \cdot \frac{d}{d_{max}}}{m \cdot \frac{d}{d_{max}} + 1}$$

Interactive Lenses

Exploring structural relationships: Graph lenses (Demo)



Problem: view cluttered with edges, neighbors not visible

Local edge lens: Clear view of irrelevant edges

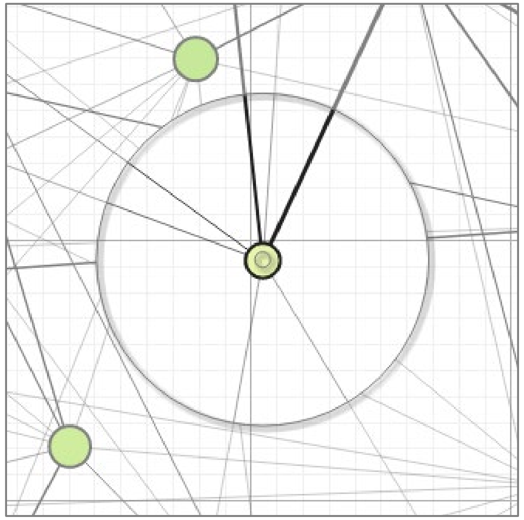
Bring-neighbors lens: Make neighbors visible

Composite lens: Combine lens effects

Interactive Lenses

Exploring structural relationships

- Local edge lens:
  - Selection: Nodes within the lens
  - Lens function and effect:
    - Processing: Determine edges incident to selected nodes
    - Effect: Suppress all other edges (i.e., edges with no endpoint in lens)
  - Join:
    - Define clip area (corresponding to lens)
    - Clear lens interior
    - Draw background grid and selected edges
    - Superimpose lens ring

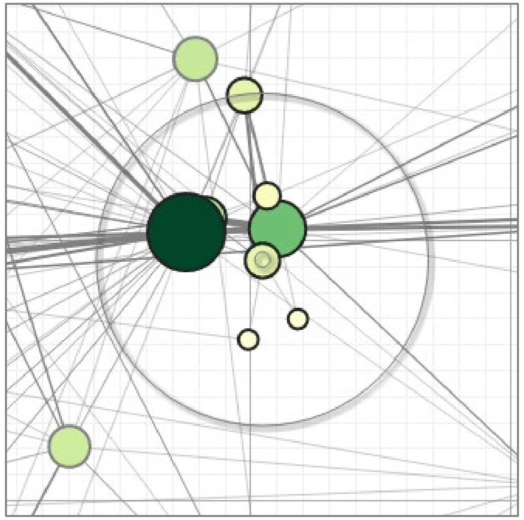


Interactive Lenses

Exploring structural relationships

The **bring-neighbors lens** implements the idea of "Bring & Go" as discussed in **Lecture 10**.

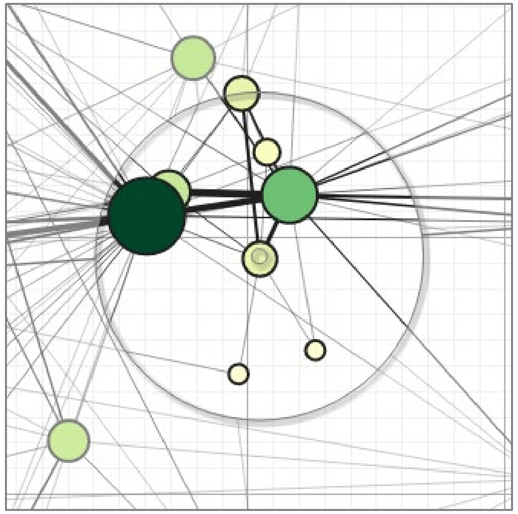
- Bring-neighbors lens:
  - Selection: Nodes within the lens
  - Lens function and effect:
    - Processing: Determine and relocate neighbors of selected nodes
    - Effect: All neighbors are within the lens
    - Think about: What are reasonable relocation strategies?
  - Join:
    - Override original node positions of neighbors
    - Superimpose lens ring



Interactive Lenses

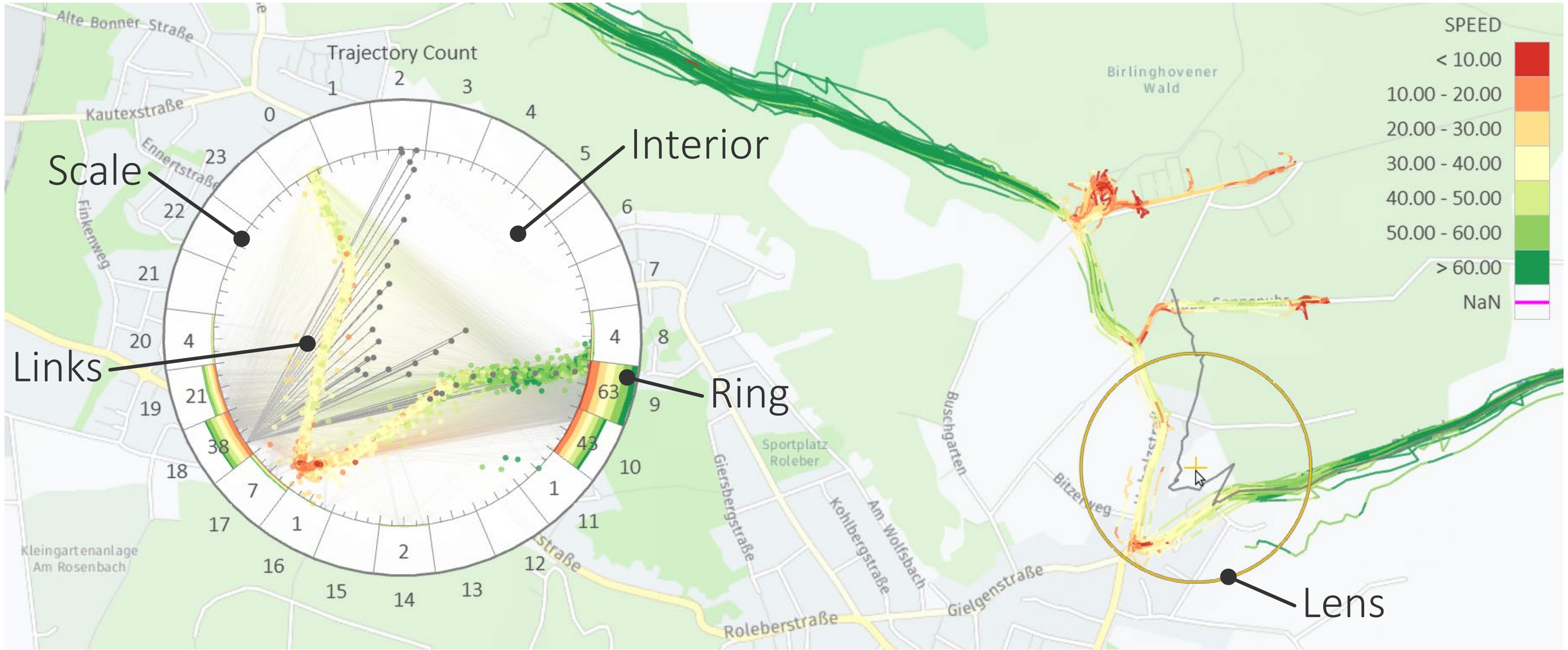
Exploring structural relationships

- Composite lens:
  - Combine three different lens effects
    - Local edge lens
    - Bring-neighbors lens
    - Fisheye lens



Interactive Lenses

Exploring temporal aspects of movements: Time lens (Demo)



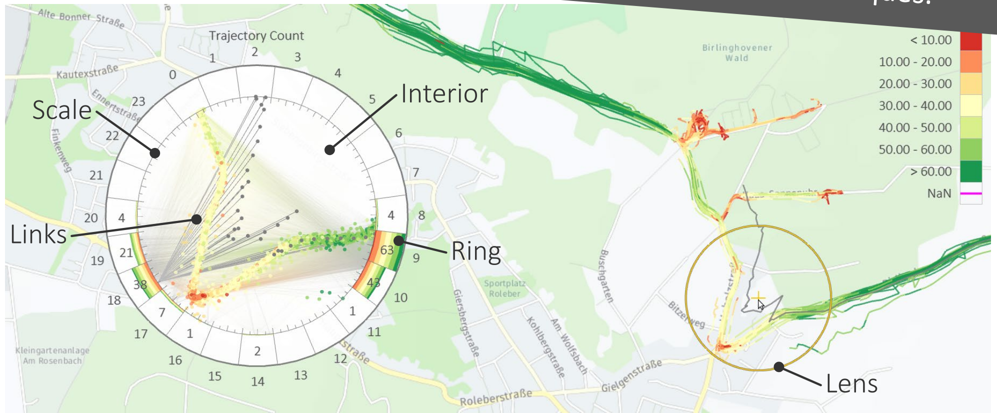
Time lens: Show aggregated temporal information about selected spatial region

Interactive Lenses

Exploring temporal aspects of movements

- Time lens
  - Selection: Trajectory points in lens
  - Lens function and effect:
    - Processing: Compute aggregated temporal information for selected trajectory points
    - Effect: Auxiliary display
  - Join:
    - Embed auxiliary display
      - Radial time histogram in ring
      - Selected nodes + time links in interior
    - Superimpose lens

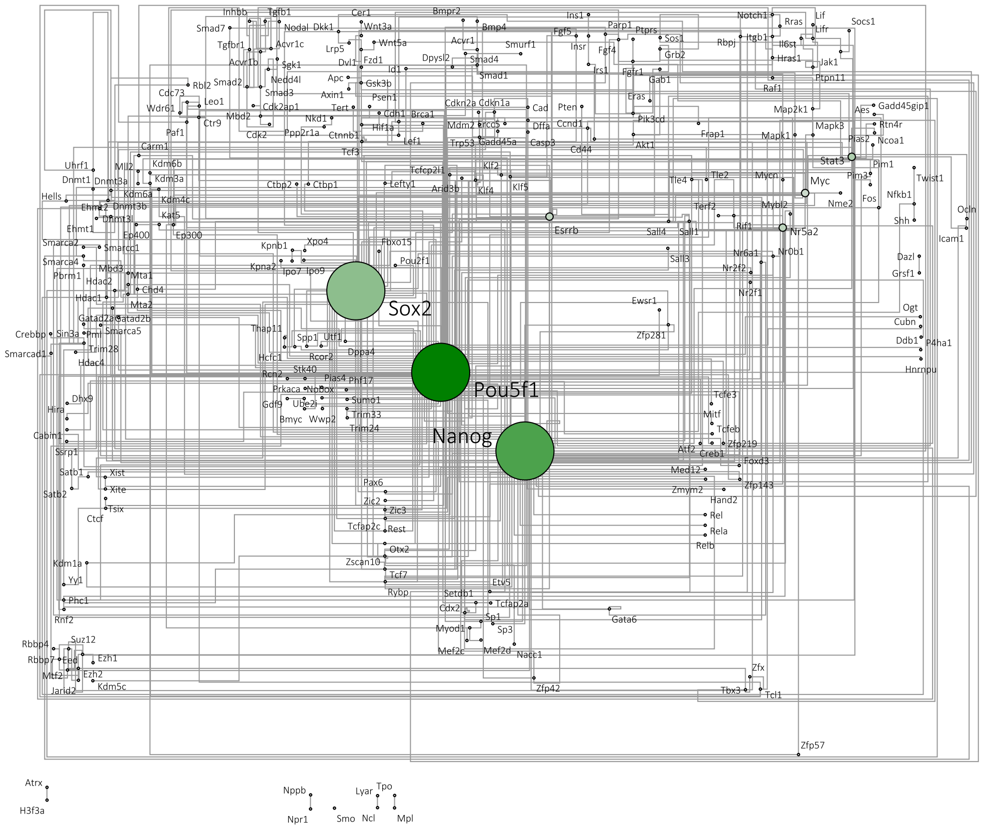
With the dedicated display, the time lens is an example of an edge case of lens techniques.



Interactive Lenses

Semi-automatic graph editing: Edit lens

- Background:
  - Large biological network manually curated based on scientific publications
  - New publications → Network changes
- Challenge: How to locally adapt an already complex network while maintaining the overall layout to preserve the users' mental map?



- Answer: Local semi-automatic adaptations with the EditLens

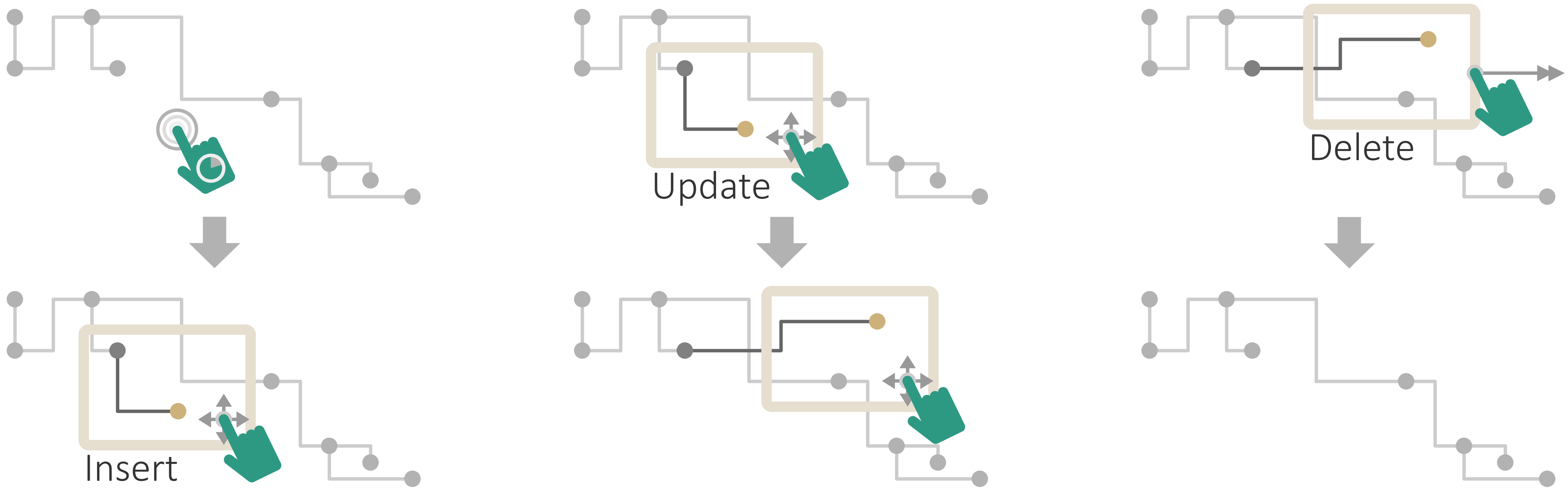
Interactive Lenses

Semi-automatic graph editing: Edit lens

- Two step procedure
  - Interactive: Coarsely position the edit lens to define region where edit should take effect
  - Automatic: Compute locally-optimal layout wrt. lens region
- Advantages
  - Human expertise can be used to maintain overall layout and existing layout conventions
  - Computer takes over optimization, which would be too costly to carry out for human users

Interactive Lenses

Semi-automatic graph editing: Edit lens



Interactive Lenses

Summary

- **Lightweight**, **local**, and **transient** adjustments of visualization
- Conceptual components: **selection**, **lens function** and **lens effect**, **join**
- Adjustable properties: Position, size, shape, orientation, parameters
- Interaction: Direct + dedicated GUI
- Lenses in action: Fish-eye lens, graph lenses, time lens, and edit lens

Assignments

- Read "Interactive Lenses for Visualization: An Extended Survey" by Tominski et al. for more conceptual background and more examples of interactive lenses!

Questions

1. Explain the mapping of data to zoomable axes!
2. What is multi-scale input and how does it help with data navigation?
3. How are interactive lens techniques defined, what are their key characteristics in terms of interaction and visual feedback?
4. Sketch the conceptual model of interactive lenses and explain how selection, lens function, and join operate?
5. What three fundamental lens effects are possible?
6. What properties do lenses have and how can they be adjusted?
7. What is a *fish-eye lens*? Explain its lens function and effect!
8. What is the key idea behind the *bring neighbors lens*?