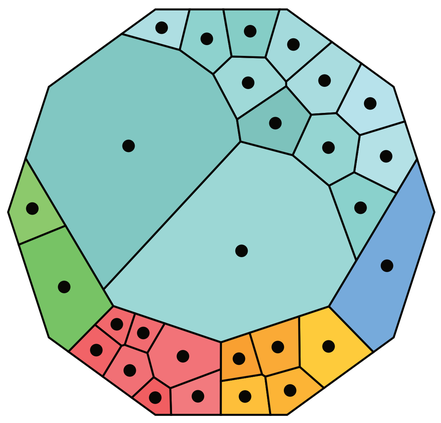
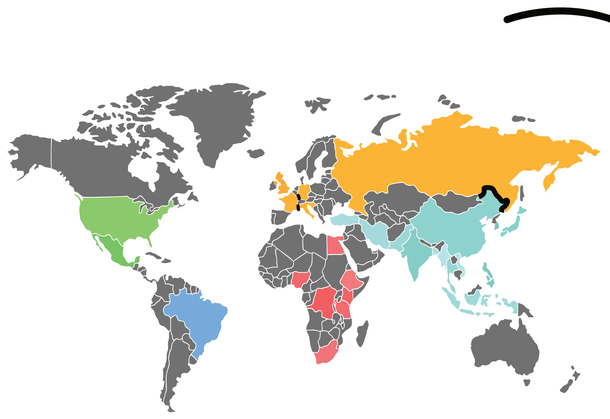
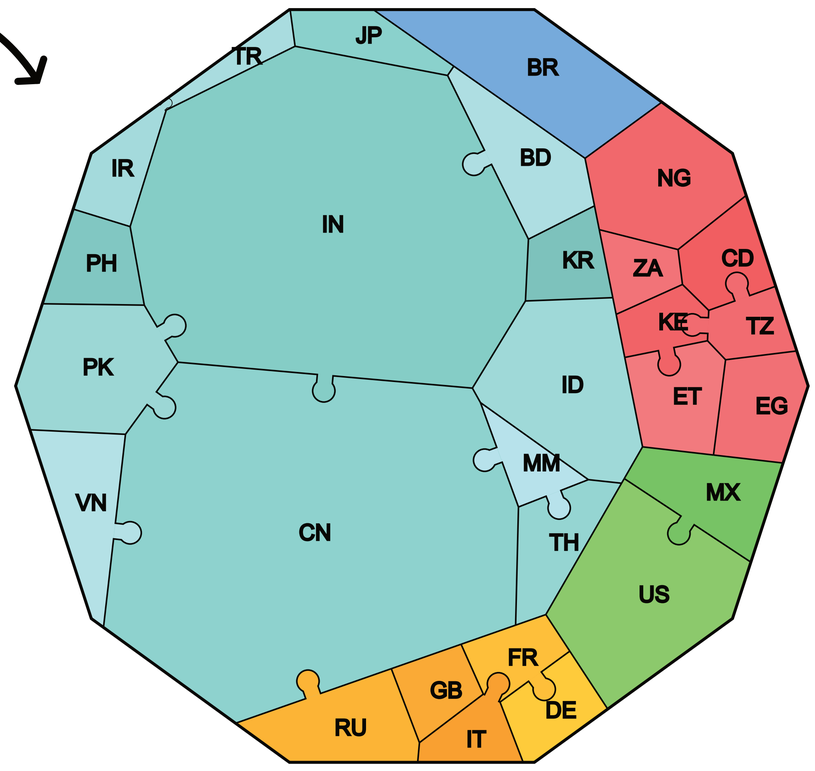
Neighborhood-Preserving Voronoi Treemaps

Rebecca Kehlbeck[1], Patrick Paetzold[1], Yumeng Xue[1], Bin Chen[1], Yunhai Wang[2], Oliver Deussen[1]

[1]University of Konstanz [2]Renmin University of China







1 A **Voronoi treemap visualization** showing countries with a population > 50 million grouped by continent. The position of countries within a continent is arbitrary. Cells are weighted by the countries' GDP.

2 Additional data features such as country borders show interesting relations between countries.

3 Using our proposed method, we create Voronoi treemaps with all their benefits, but in addition, **retain neighborhood relations** and **visually indicate the similarity** between neighboring cells.

DATASET	LEVELS	#TOTAL NODES	#LEAF NODES	#NEIGBORHOOD RELATIONS (LOWEST LEVEL)	AVG. AREA ERROR	MAX GRAPH DISTANCE	AVG. ASPECT RATIO	# CONSTRAINTS PRESERVED (LOWEST LEVEL) RANDOM (BASELINE)	AFTER INITIALIZATION (OUR METHOD)	AFTER OPTIMIZATION (OUR METHOD)
Country Population	3	34	28	18	0.01	2	1.12	10(52.6%)	16(88.8%)	15(83.33%)

4 Compared to the baseline Voronoi treemap visualization in 1, our method **maintains 35,2% more** neighborhood relations.

Motivation

Weighted Voronoi Diagrams
- Voronoi diagrams are a space-filling visualization technique showing a collection of data elements. Weighted Voronoi diagrams encode an attribute (e.g., GPD of each country) via the size of each Voronoi cell
- Cells move towards their centroids in an iterative optimization process

Voronoi Treemaps
- Voronoi treemaps are a popular technique to visualize hierarchical data using Voronoi diagrams for each hierarchy level

Neighborhood Relations
- Oftentimes, data elements have attributes that describe relations between them, e.g., the neighborhood of countries, which makes some elements more similar to others
- These relations may cross diagram edges on the same hierarchy level

Goals

Preserving Neighborhoods during Optimization
- Utilize the similarity or relations of elements to influence the position of their Voronoi cell during optimization
- Data elements with a high similarity or common attributes should share a Voronoi edge in the diagram

Visual Encoding of Neighborhood Status
- If two Voronoi cells share a border and are highly similar, we want to communicate this in the visualization
- We utilize a 'Jigsaw' metaphor to show that neighboring nodes are similar by interlocking them with a puzzle tab

Choice of Different Initialization Strategies
- We compare and allow the user to utilize different initialization strategies for the Voronoi diagram initialization step, such as random, swapping, and force-based methods

Method

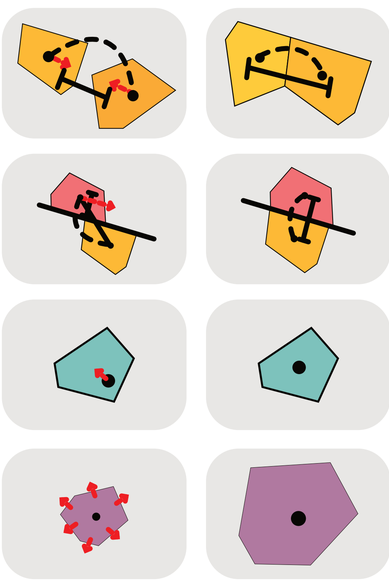
Input Data
- Hierarchical data with an additional neighborhood feature or other co-occurring attributes

Data Preprocessing
- The neighborhood feature is propagated through the hierarchy from the bottom to the top by aggregation of each node's children

Voronoi Treemap Algorithm
- The Voronoi treemap algorithm is a top-down approach
- Each level of the hierarchy is used to create one or multiple Voronoi diagrams on that level
- As relations can cross parent Voronoi cell borders, all Voronoi diagrams on a hierarchy level are computed in a round-robin queue

Voronoi Initialization & Swapping
- The initial position influences how fast the optimization converges
- In our experience, swapping improves neighborhoods the fastest, as far away neighbors otherwise need many optimization steps

Attraction Forces during Voronoi Optimization
- During each Voronoi optimization process, Voronoi cells
  - Move towards their preferred neighbor
  - Align along parent edges
  - Move towards their cell centroid
  - Grow in size according to their weight attribute



Limitations

Preservation Ratio depends on Feature Distribution
- Datasets where all nodes on a level have similar features mean that we can only preserve some neighborhoods
- Even if two nodes are not similar, they might still share a Voronoi edge in the diagram, as the visualization is space-filling

Only Locally Optimal Solution
- Because we use a greedy, optimization-based method, we may not find a globally optimal solution

Readability Issues with Voronoi Diagrams
- Some hierarchies are very deep and wide. As the size of the Voronoi treemap is fixed, this means that some cells can get very small

SFB-TRR 161 Quantitative Methods for Visual Computing
Universität Konstanz
DFG