UNIVERSITÄT BONN igg

Advanced Algorithms for GIS

Organizatorial Matters & Topic Presentation

Geoinformation Group
Institute of Geodesy and Geoinformation
University of Bonn

Aim of the seminar

- You will work on different current research topics
- Learn to read, understand, and present research papers
- Learn to develop, implement, adapt, and evaluate algorithms

Organizatorial Matters

Topic Selection (09.10.2024 – 13.10.2024)

- Today different seminar topics will be presented
- Groups of 1 – 2 students
- Each group is assigned to one topic
- You can express your preferences on eCampus until 13.10.

Organizatorial Matters

Topic Selection (09.10.2024 – 13.10.2024)

- Today different seminar topics will be presented
- Groups of 1 – 2 students
- Each group is assigned to one topic
- You can express your preferences on eCampus until 13.10.

Working Phase (16.10.2024 – 20.01.2025)

- Working on literature (handed out) and implementation tasks
- Each Wednesday, 10:15 – 11:45 free working in the GIS-Labor with the possibility to talk to your supervisor

Organizatorial Matters

Topic Selection (09.10.2024 – 13.10.2024)

- Today different seminar topics will be presented
- Groups of 1 – 2 students
- Each group is assigned to one topic
- You can express your preferences on eCampus until 13.10.

Working Phase (16.10.2024 – 20.01.2025)

- Working on literature (handed out) and implementation tasks
- Each Wednesday, 10:15 – 11:45 free working in the GIS-Labor with the possibility to talk to your supervisor

Final Presentations (21.01.2025 + 22.01.2025)

- 15 minutes of presentation + 5 minutes of discussion per student
- Approx. 1/3 of the time for problem motivation and related work, 1/3 details on algorithm engineering and implementation, 1/3 experiments
- Attendance is mandatory

Organizatorial Matters

Topic Selection (09.10.2024 – 13.10.2024)

- Today different seminar topics will be presented
- Groups of 1 – 2 students
- Each group is assigned to one topic
- You can express your preferences on eCampus until 13.10.

Working Phase (16.10.2024 – 20.01.2025)

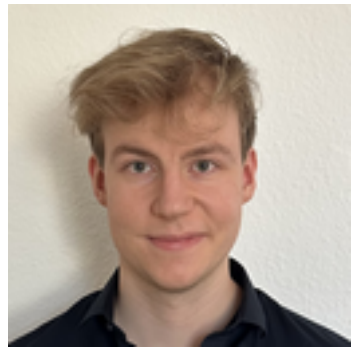
- Working on literature (handed out) and implementation tasks
- Each Wednesday, 10:15 – 11:45 free working in the GIS-Labor with the possibility to talk to your supervisor

Final Presentations (21.01.2025 + 22.01.2025)

- 15 minutes of presentation + 5 minutes of discussion per student
- Approx. 1/3 of the time for problem motivation and related work, 1/3 details on algorithm engineering and implementation, 1/3 experiments
- Attendance is mandatory

Successful presentation is decisive for the exam admission!

Contact



Alexander Naumann
naumann@igg.uni-bonn.de
Room 2.014

Alina Nitzke
nitzke@igg.uni-bonn.de
Room 2.017

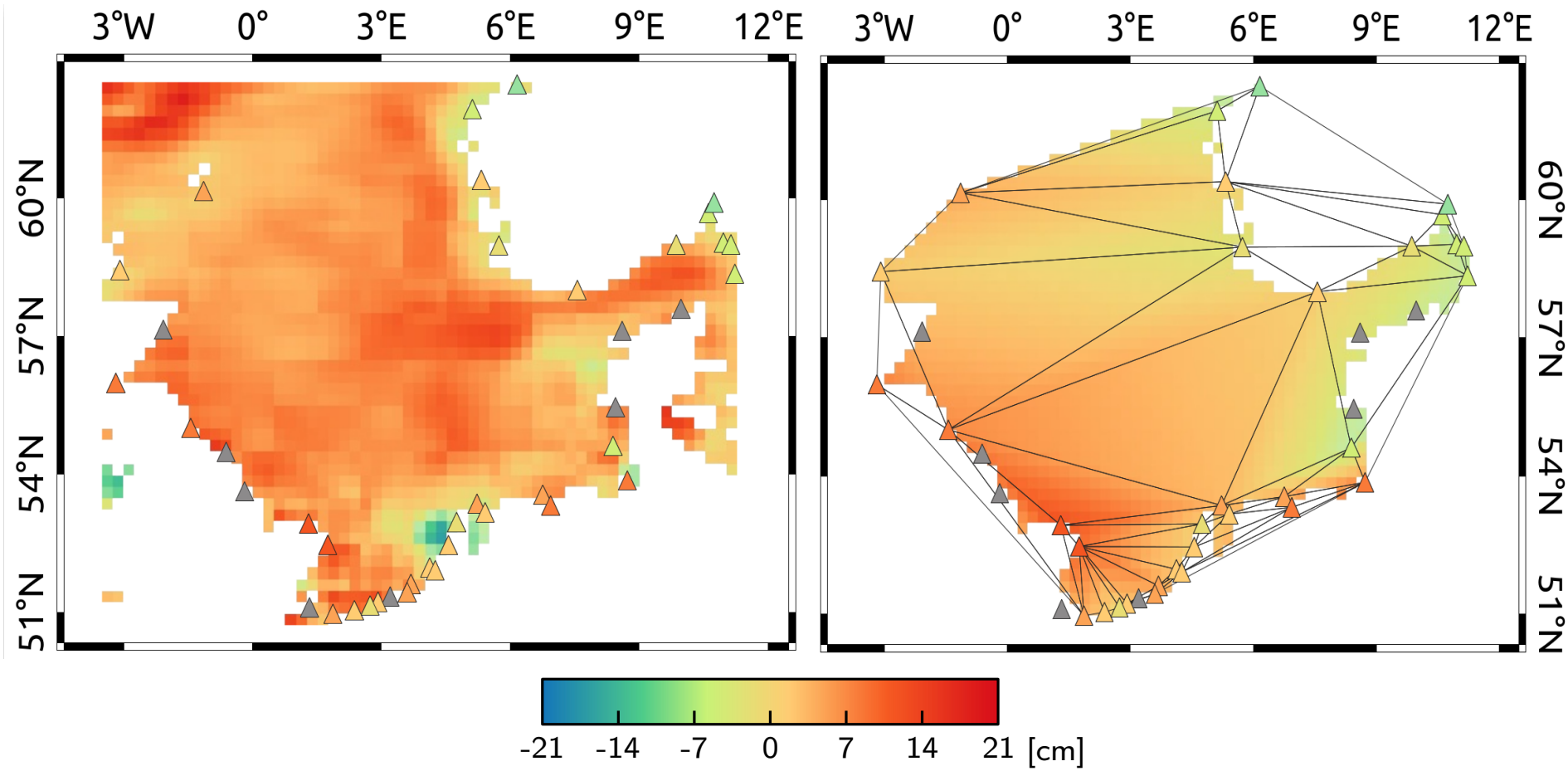
Daniel Bauer
bauer@igg.uni-bonn.de
Room 2.013

Julius Knechtel
knechtel@igg.uni-bonn.de
Room 2.007

Peter Rottmann
rottmann@igg.uni-bonn.de
Room 2.007

Sven Gedicke
gedicke@igg.uni-bonn.de
Room 2.017

Topic 1: Local Search Approach Using Edge Flips for Data-dependent Triangulation



- Supervisor: Alina Nitzke
- Programming Language: Java or Python
- Number of Students: 2

Topic 1: Local Search Approach Using Edge Flips for Data-dependent Triangulation

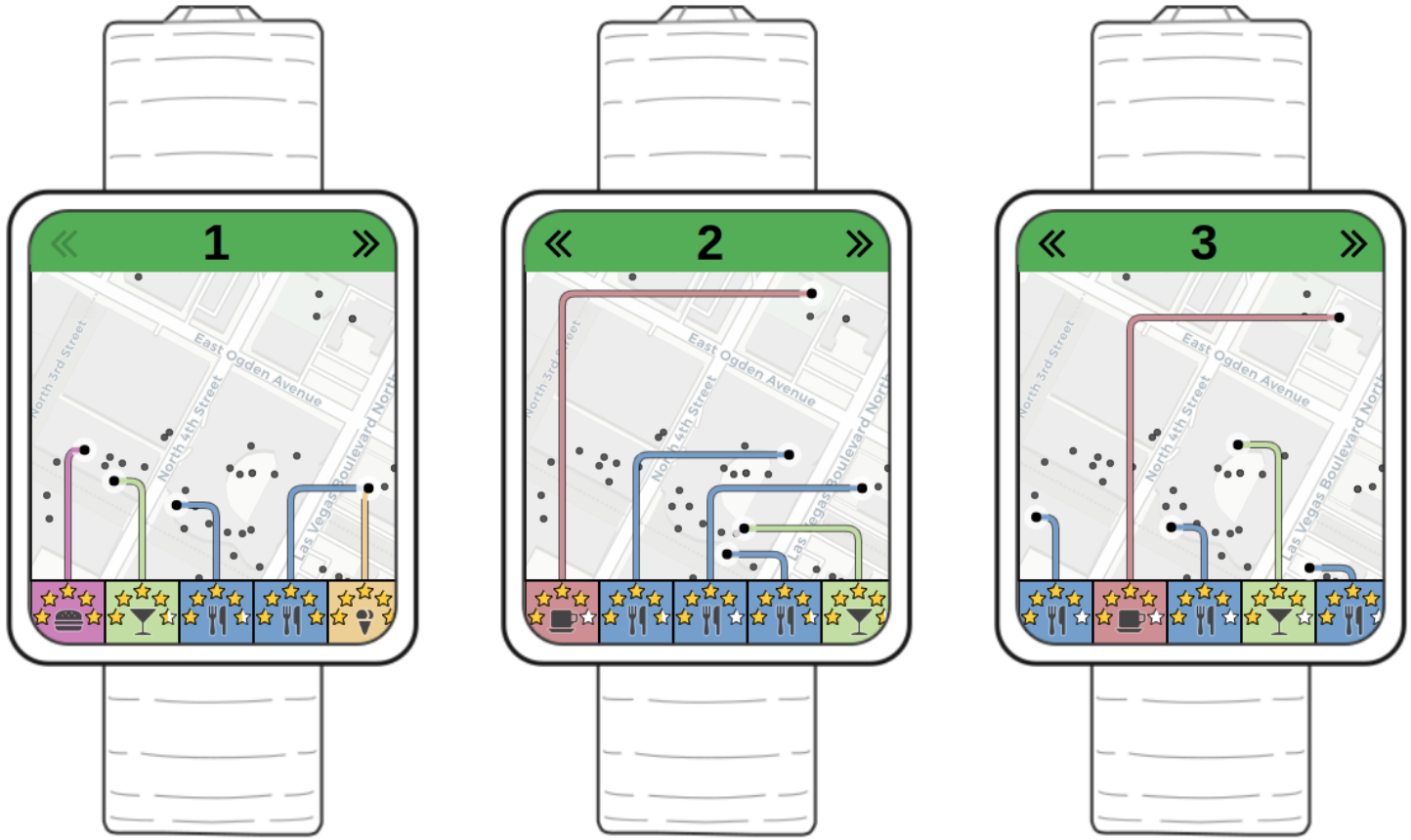
In the field of computational geometry, triangulation techniques are used to interpolate data and approximate surfaces. The objective of *data-dependent triangulations* is to optimize the triangulation for a better representation of the underlying data (Dyn et al., 1990), such as sea surface heights. This task focuses on optimizing a triangulation of sea surface heights from tide gauges along the North Sea coast. You will improve an initial Delaunay Triangulation (DT) by using a heuristic *edge-flip-based local search algorithm* to approximate an optimal, data-dependent triangulation that minimizes interpolation error.
Key aspects:

1. Edge Flip Implementation: Develop a local search strategy to iteratively improve the initial triangulation by flipping edges.
2. Criteria Selection: Choose a useful search criterion.
3. Improvement and Comparison: Visualize the improvement and compare the results with an optimal solution.

[1] Dyn, N., Levin, D., Rippa, S., 1990. Data dependent triangulations for piecewise linear interpolation. IMA Journal of Numerical Analysis 10, 137-154. doi:10.1093/imanum/10. 1.137.

[2] Hurtado, F., Noy, M., Urrutia, J. (1996, May). Flipping edges in triangulations. In Proceedings of the twelfth annual symposium on Computational geometry (pp. 214-223).

Topic 2: One-Sided Multi-Page Boundary Labeling Using Local Search



- Supervisor: Sven Gedicke
- Programming Language: Java
- Number of Students: 2

Topic 2: One-Sided Multi-Page Boundary Labeling Using Local Search

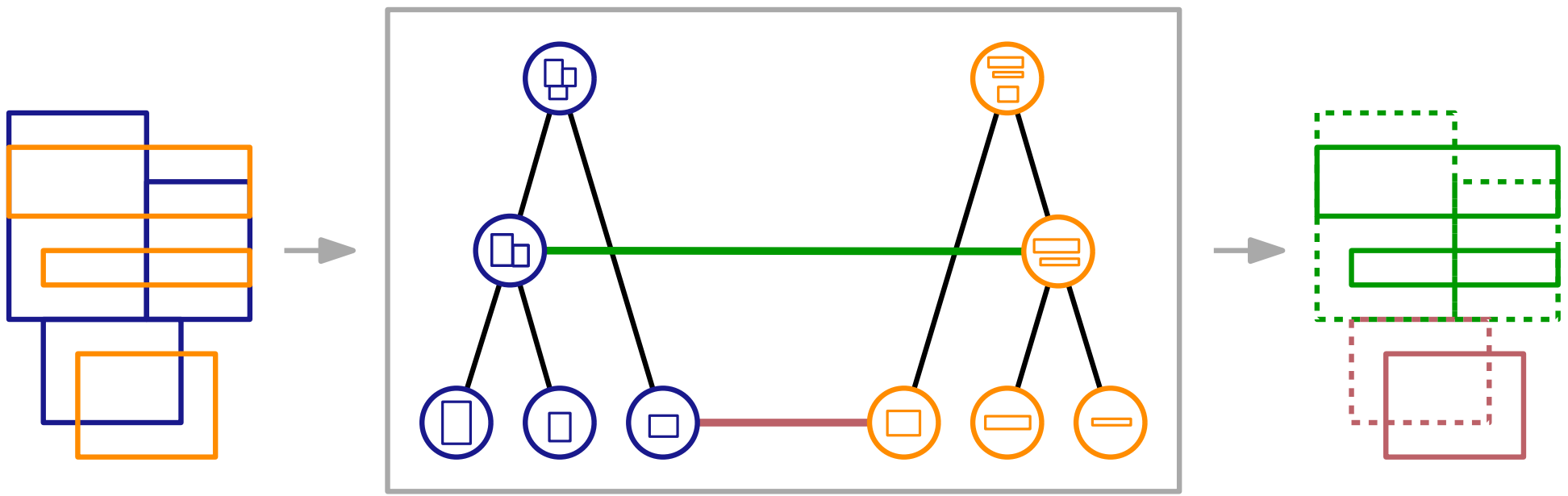
In the field of cartography, labels in the form of text or descriptive symbols are used for conveying additional information. In the approach of *one-sided multi-page boundary labeling*, the labels are placed on one side of the map and visually connected to their corresponding features via so-called *leaders* [1]. To be able to explore all features within a map section, the labels are distributed on multiple pages (see figure) through which a user can navigate.

An exact optimization approach, which distributes the labels on pages considering both the relevance of the features and the length of the leaders, has been proposed by Gedicke et al. (2021) [2]. You will be provided with the corresponding code in Java. Your task is to develop an alternative heuristic approach that uses a local search strategy. In addition to the previously mentioned criteria, this approach should also focus on avoiding leaders running closely in parallel.

[1] Bekos, M. A., Niedermann, B., & Nöllenburg, M. (2019). External labeling techniques: A taxonomy and survey. In *Computer Graphics Forum* (Vol. 38, No. 3, pp. 833-860). https://doi.org/10.1111/cgf.13729

[2] S. Gedicke, A. Bonerath, B. Niedermann, & J.-H. Haunert (2021). Zoomless maps: external labeling methods for the interactive exploration of dense point sets at a fixed map scale. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1247–1256. https://doi.org/10.1109/TVCG.2020.3030399

Topic 3: m:n Matching of Hierarchically Grouped Polygons



- Supervisor: Alexander Naumann
- Programming Language: Python
- Number of Students: 1–2

Topic 3: m:n Matching of Hierarchically Grouped Polygons

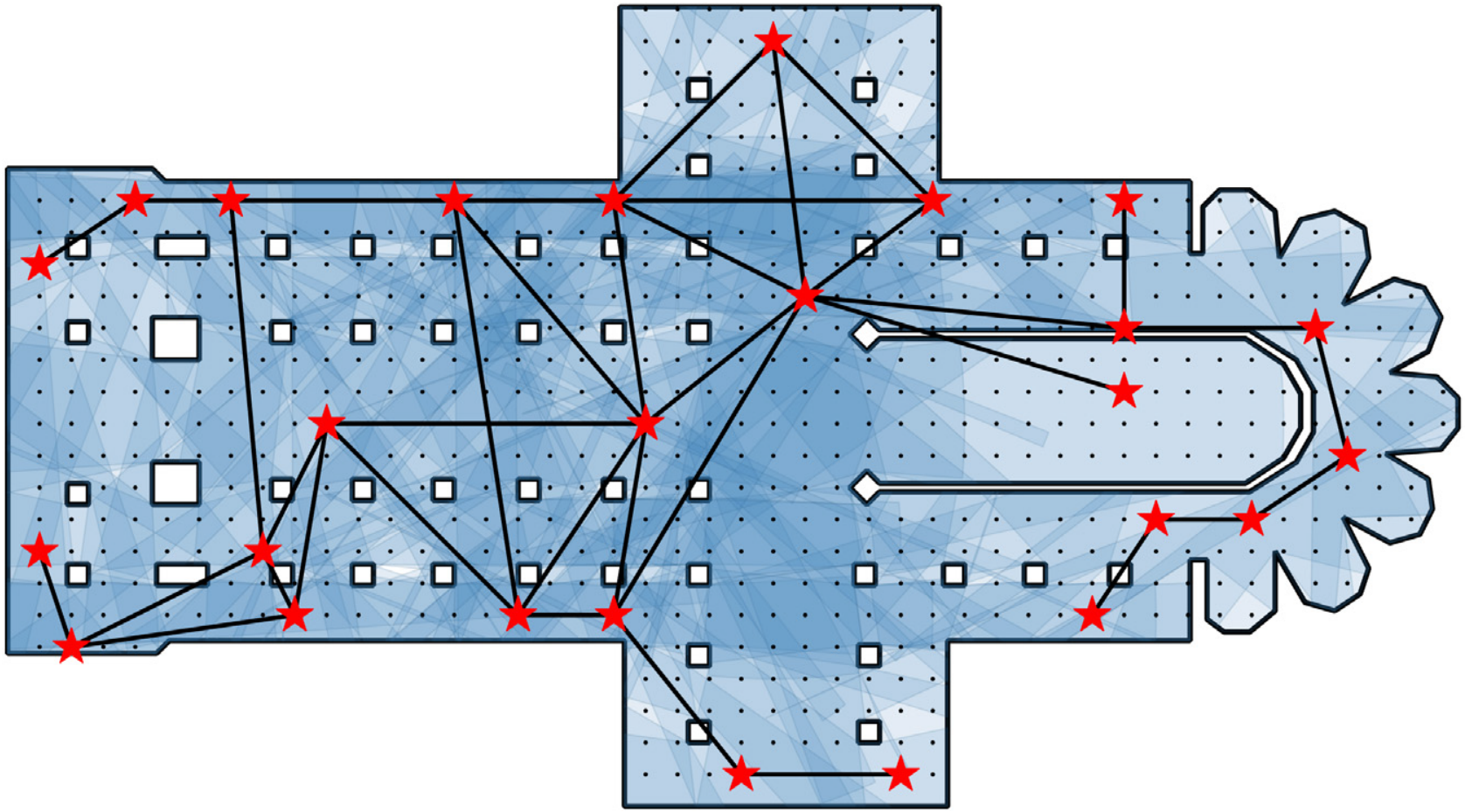
This task involves implementing and evaluating an algorithm to hierarchically group polygons in the context of many-to-many matching.

Sets of polygons will be provided. The new code should then compute and output a hierarchical grouping per set. This will be fed into existing code to compute a matching. The strategy of building the hierarchy will be similar to Kruskal's algorithm for minimum spanning trees in that it will combine nodes iteratively based on a distance measure. The method should finally be evaluated based on the resulting matching regarding advantages, disadvantages and plausibility.

[1] Fan, H., Zipf, A., Fu, Q., & Neis, P. (2014). Quality assessment for building footprints data on OpenStreetMap. International Journal of Geographical Information Science, 28(4), 700–719. https://doi.org/10.1080/13658816.2013.867495

[2] Rottmann, Peter & Driemel, Anne & Haverkort, Herman & Röglin, Heiko & Haunert, Jan-Henrik. (2022). Bicriteria Shapes: Hierarchical Grouping and Aggregation of Polygons with an Efficient Graph-Cut Approach. 10.21203/rs.3.rs-2116893/v1.

Topic 4: Employing Local Search for Static Laserscan Planning



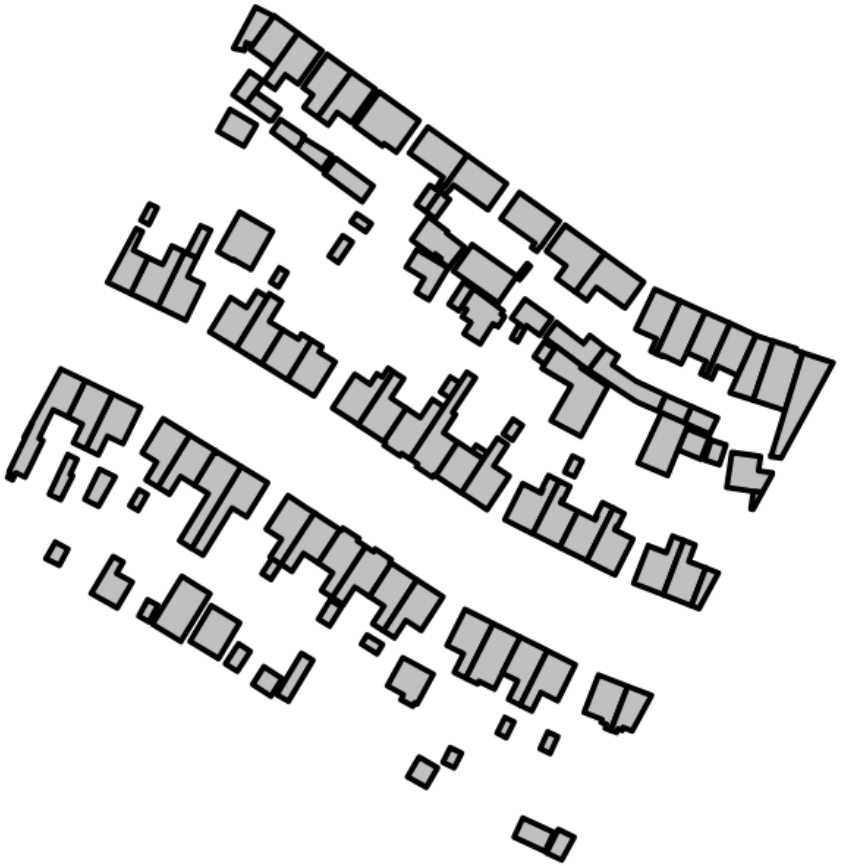
- Supervisor: Julius Knechtel
- Programming Language: Java
- Number of Students: 2

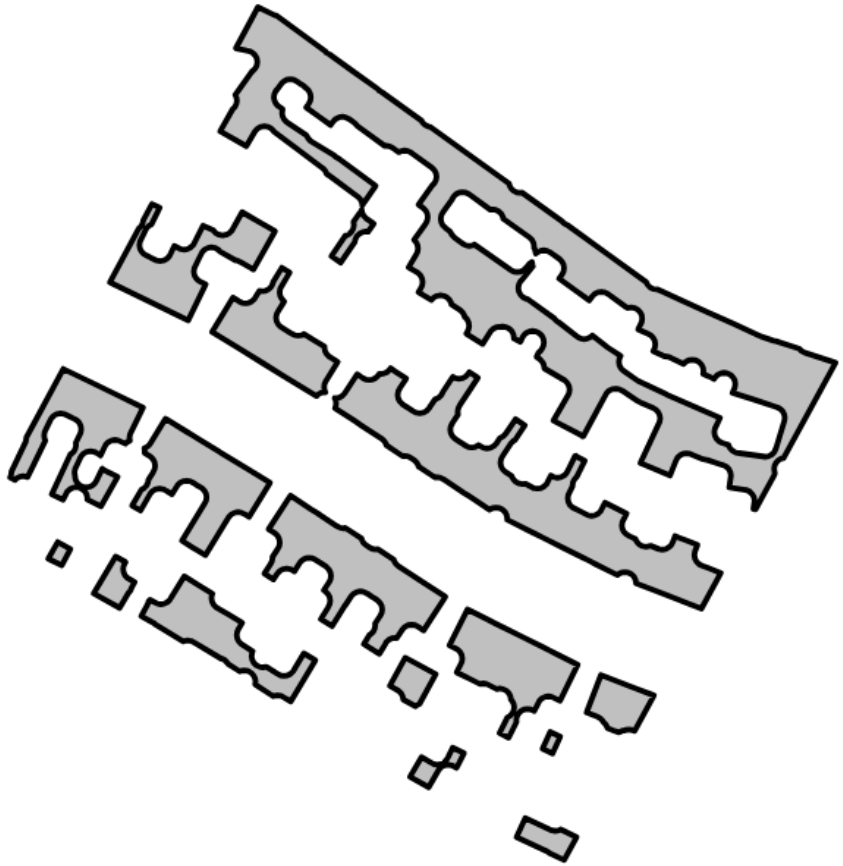
Topic 4: Employing Local Search for Static Laserscan Planning

In the context of laserscanning, the number of standpoints heavily influences the duration of a survey and, hence, also the economic efficiency. However, using less standpoints poses the risk of not observing some parts of the scene or not being able to sucessfully register the respective scans together. Dehbi et al. proposed an ILP based approach to calculate the optimal number of standpoints and their respective positions incorporating the visibility of the scan object and registrability of the standpoints.
However, solving an ILP can be time-consuming. Hence, Soudarissanane and Lindenbergh proposed a greedy algorithm selecting the next best viewpoint. However, registrability constraints are not considered. The task in this project is to build on given code (i.e. visibility and registrability analysis) to employ a local search algorithm. This algorithm should be evaluated, e.g. with respect to the number of standpoints.

[1] Dehbi, Y. and Leonhardt, J. and Oehrlein, J. and Haunert, J.-H.: Optimal scan planning with enforced network connectivity for the acquisition of three-dimensional indoor models, ISPRS Journal of Photogrammetry and Remote Sensing, Volume 180, 2021, 103-116, https://doi.org/10.1016/j.isprsjprs.2021.07.013
[2] Soudarissanane, S. and Lindenbergh, R.: Optimizing Terrestrial Laser Scanning Measurement Set-up, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XXXVIII-5/W12, 2011, 127–132, https://doi.org/10.5194/isprsarchives-XXXVIII-5-W12-127-2011,

Topic 5: Building Aggregation with Morphological Operators





- Supervisor: Peter Rottmann
- Programming Language: QGIS/Python
- Number of Students: 1–2

Topic 5: Building Aggregation with Morphological Operators

In the context of map generalization, creating a single large area from a given set of smaller, detailed polygons is a common challenge. The main difficulty is to construct polygons that best represent the given polygons at the desired map scale, i.e., polygons that are similar to the input while simplifying the overall shape. Simplification of the output shape should result in a reduction of the total boundary length compared to the total boundary length of the input polygons.

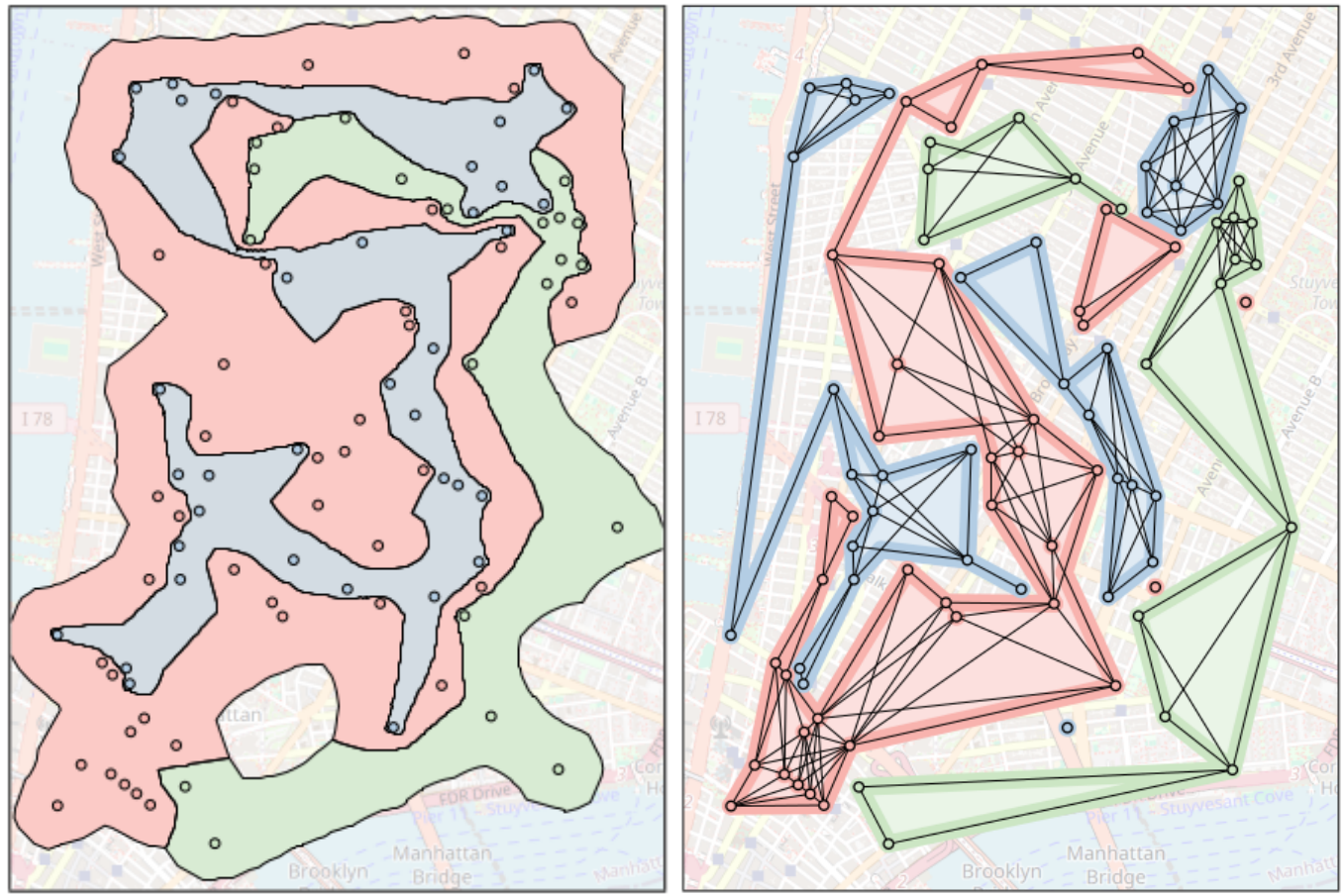
In recent years, the use of morphological operators has been investigated [1,2]. By setting an appropriate distance threshold, these methods can merge multiple buildings into multiple settlement areas.

The task of this project is to investigate the influence of different buffering thresholds and techniques on the boundary length of the resulting polygons. Additionally, the number of separate polygons in the results has to be evaluated.

[1] Damen, Jonathan, Marc van Kreveld, and Bert Spaan. "High quality building generalization by extending the morphological operators." 11th ICA Workshop on Generalization and Multiple Representation, Montpellier, France. 2008.

[2] Peng, Dongliang, and Guillaume Touya. "Continuously generalizing buildings to built-up areas by aggregating and growing." Proceedings of the 3rd ACM SIGSPATIAL Workshop on Smart Cities and Urban Analytics. 2017.

Topic 6: Connecting Coloured Points with Steiner Trees



- Supervisor: Daniel Bauer
- Programming Language: Python (or something else if you can convince me)
- Number of Students: 2

Topic 6: Connecting Coloured Points with Steiner Trees

Maps visualise categorised locations with different colours. This approach enhances readability as a label is replaced with a colour. However, finding all objects of the same category can be troublesome. Current research investigates ways of finding mathematical models to connect coloured points in order to easily find objects of the same category. The state of art considers (among others) variations of the Steiner tree problem to model and solve this problem. This group will follow this approach by

- understanding the Steiner tree problem, the given approximation algorithm [1] and how it is used to solve our problem [2]
- designing their own algorithm for our use case (with obstacles) based on the given algorithm [1]
- implementing above mentioned algorithms for categorised points in the plane
- critically discussing pros and cons of their approach

[1] Wu, Y.F., Widmayer, P. & Wong, C.K. A faster approximation algorithm for the Steiner problem in graphs. Acta Informatica 23, 223–229 (1986). https://doi.org/10.1007/BF00289500
[2] Efrat, A., Hu, Y., Kobourov, S.G., Pupyrev, S. (2014). MapSets: Visualizing Embedded and Clustered Graphs. In: Duncan, C., Symvonis, A. (eds) Graph Drawing. GD 2014. Lecture Notes in Computer Science, vol 8871. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-45803-7_38