ChinaVis 2021

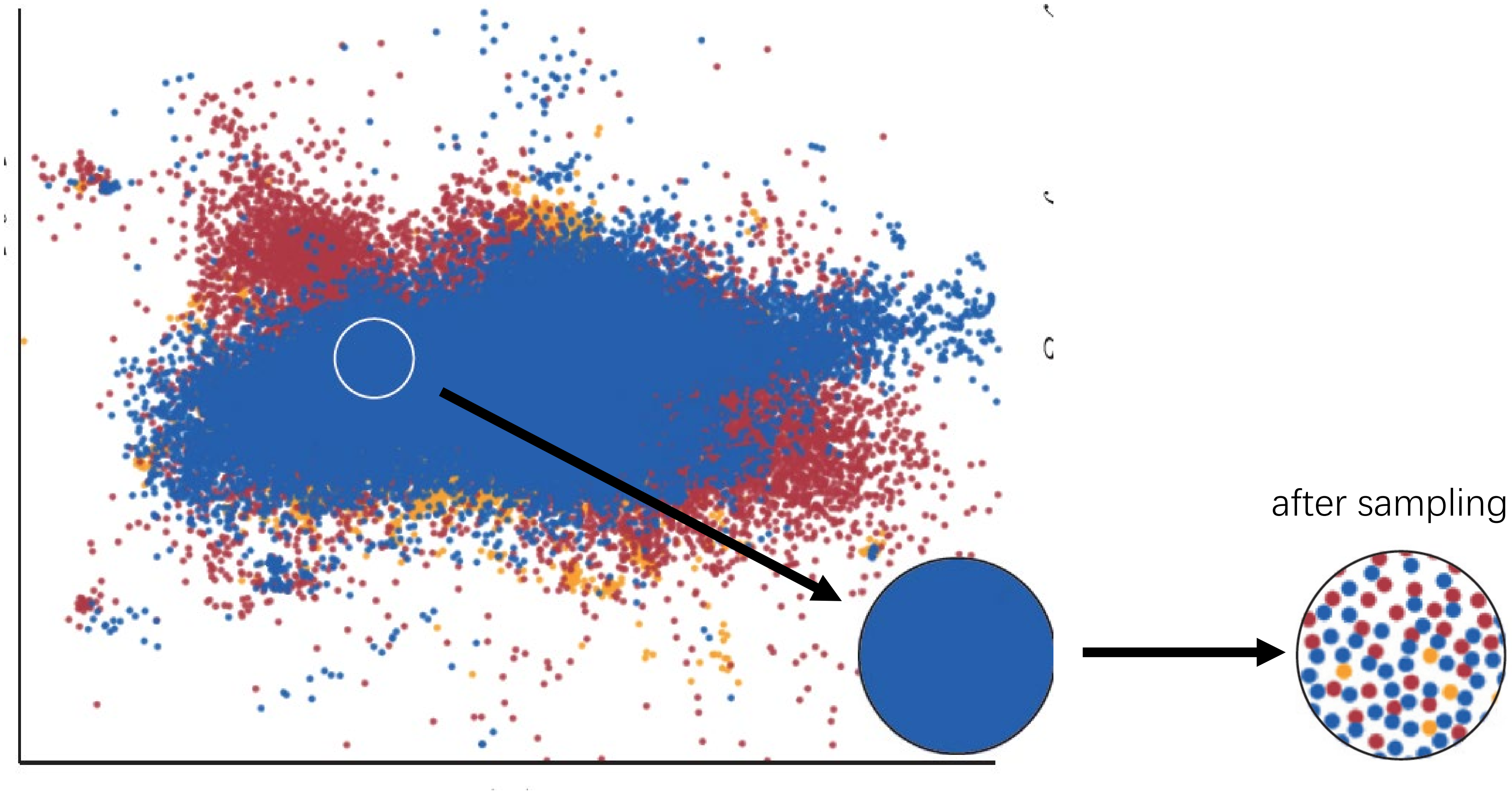
ChinaVis 2021
第八届中国可视化与可视分析大会
The 8th China Visualization and Visual Analytics Conference

Construct Boundaries and Place Labels for Multi-class Scatterplots

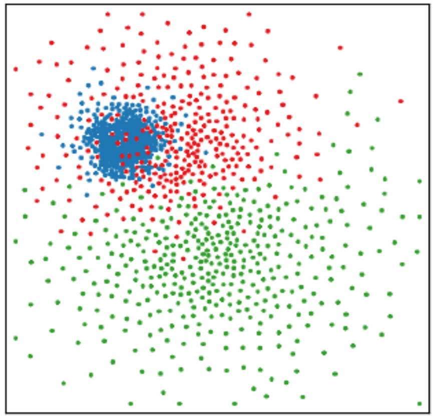
Zeyu Li, Teng Wang, Meng Wang, Jiawan Zhang

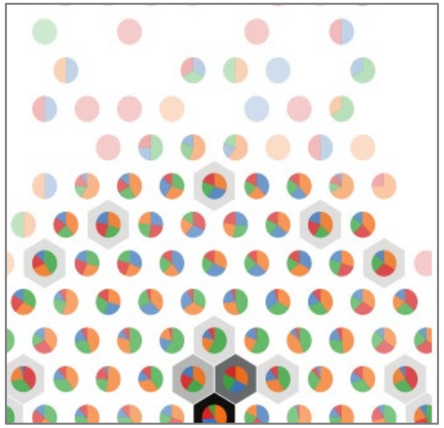
Tianjin University

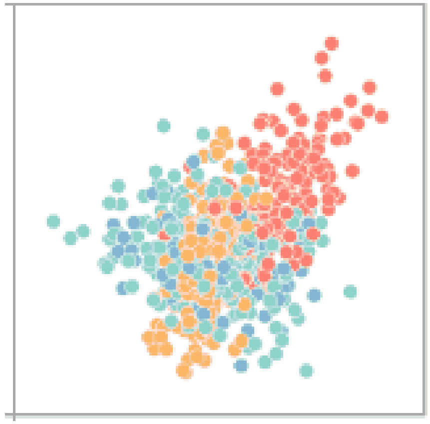
Overdraw problem in Multi-class Scatterplots

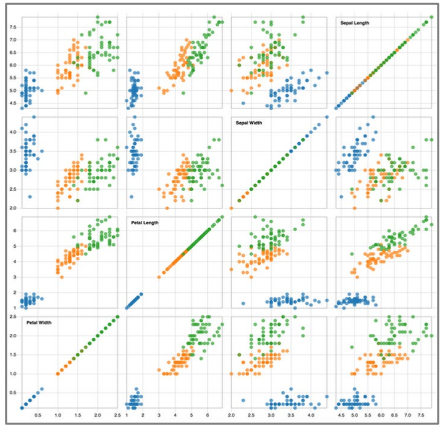


Relieve Overdraw problem by visual abstraction


Sampling


Bin aggregation


Animation

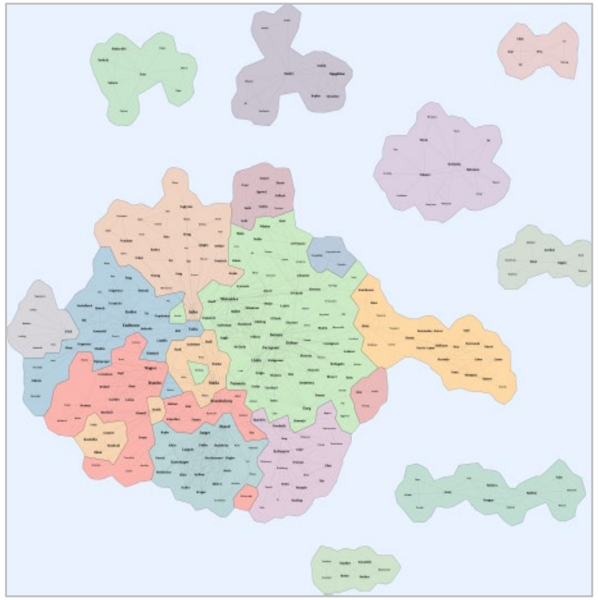

Scatterplot Matrix


Boundary Construction

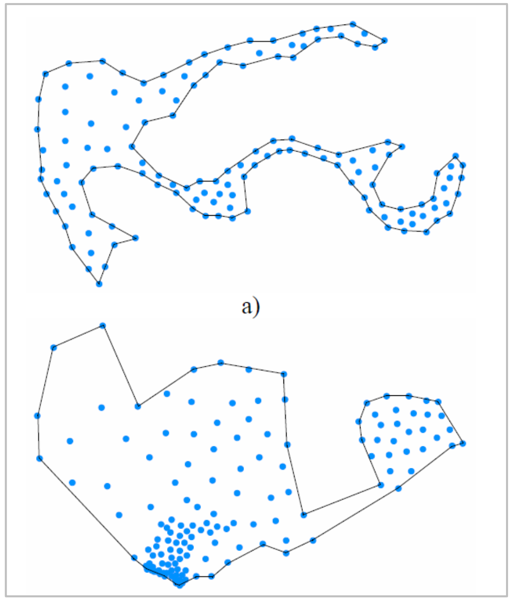
The advantages of **Boundary Construction** over other methods:

- highlights the scope of class explicitly
- has better scalability on the number of class
- adapts to static media
- needs less space, and can reveal relationships between more than two classes simultaneously

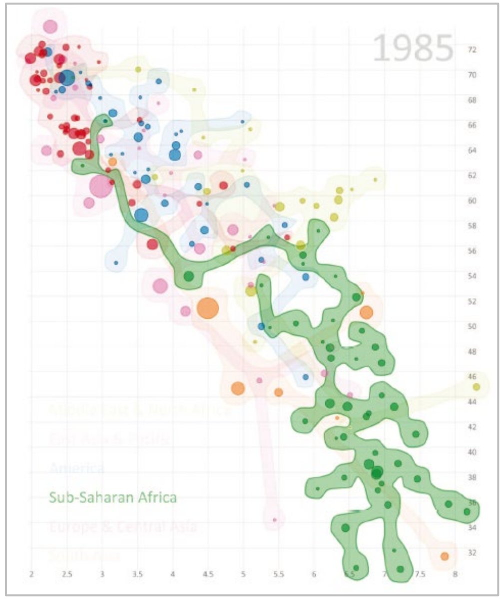
Four Techniques for Boundary Construction



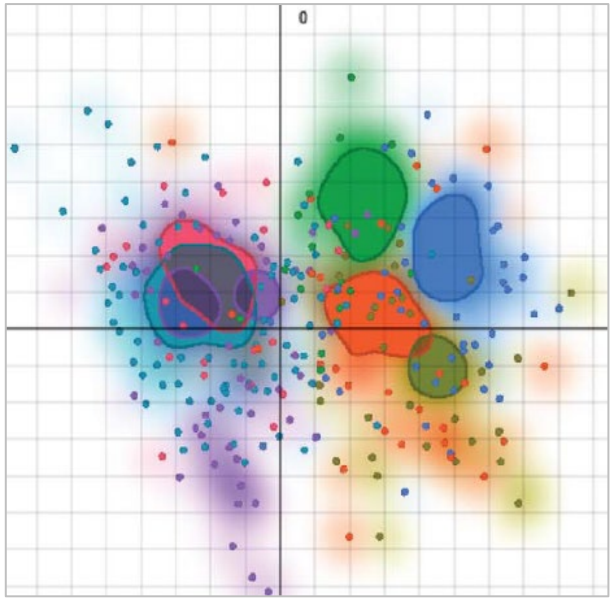
Tessellation-based technique
Gmap[1]



Hull-based technique
Concave hull[2]



Set visualization based technique: Bubble Sets[3]

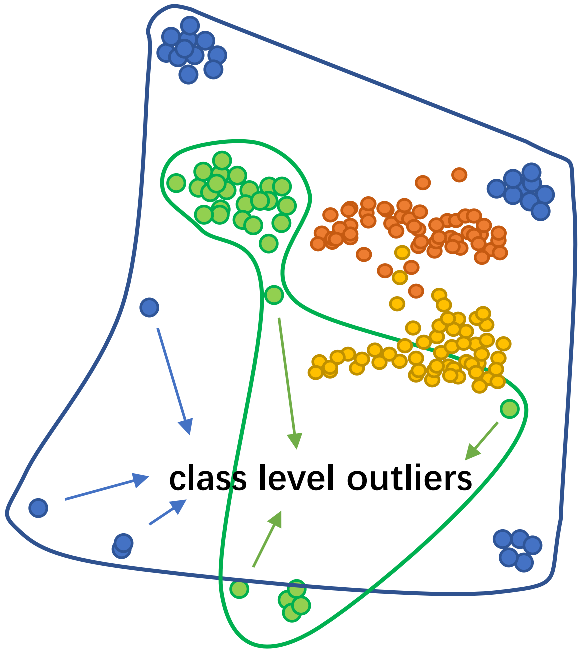


KDE-based technique
Splatterplots[4]

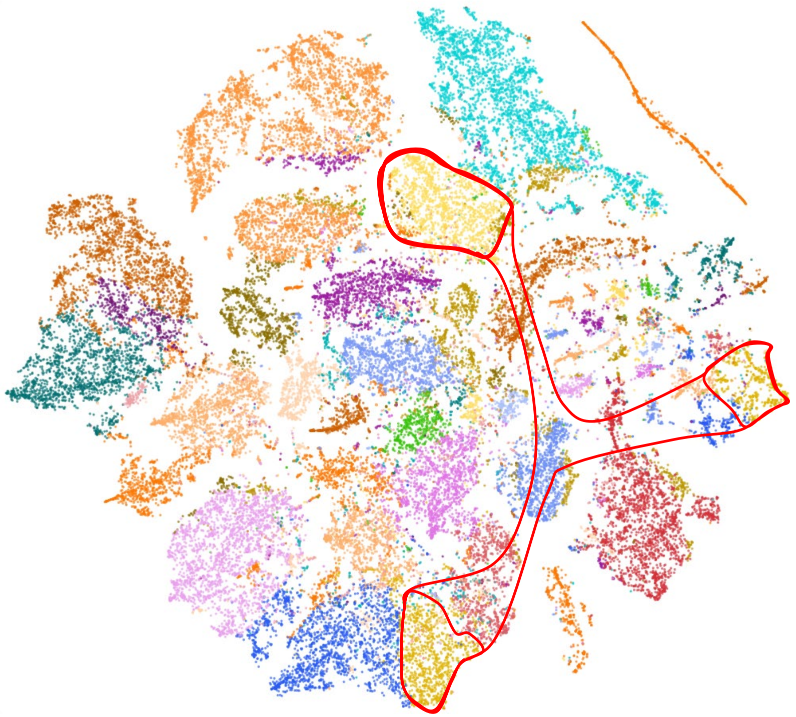
[1] Gansner E R, Hu Y, Kobourov S. GMap: Visualizing graphs and clusters as maps[C]//2010 IEEE Pacific Visualization Symposium (PacificVis). IEEE, 2010: 201-208.
[2] Moreira A, Santos M Y. Concave hull: A k-nearest neighbours approach for the computation of the region occupied by a set of points[J]. 2007.
[3] Collins C, Penn G, Carpendale S. Bubble sets: Revealing set relations with isocontours over existing visualizations[J]. IEEE Transactions on Visualization and Computer Graphics, 2009, 15(6): 1009-1016.
[4] Mayorga A, Gleicher M. Splatterplots: Overcoming overdraw in scatter plots[J]. IEEE transactions on visualization and computer graphics, 2013, 19(9): 1526-1538.

Disadvantages of the First Three Techniques

Data misunderstanding



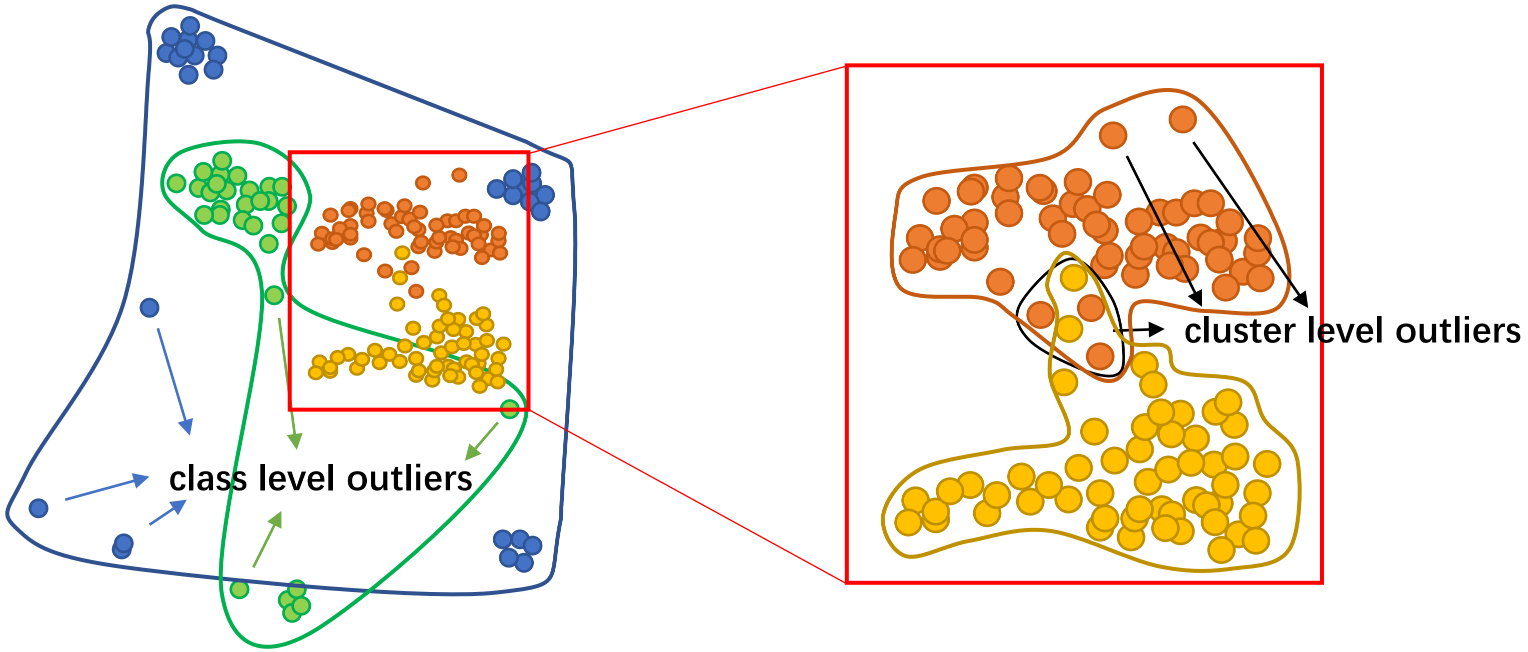
Instance



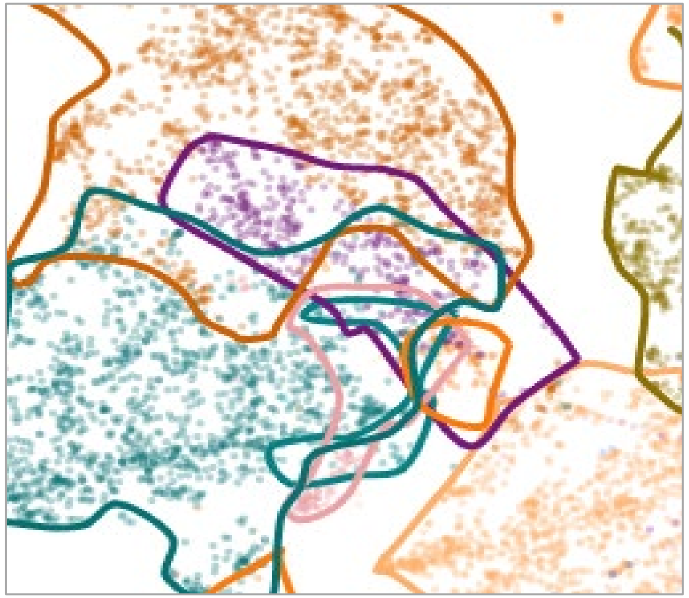
Disadvantages of the First Three Techniques

Data misunderstanding

Visual clutter



Instance

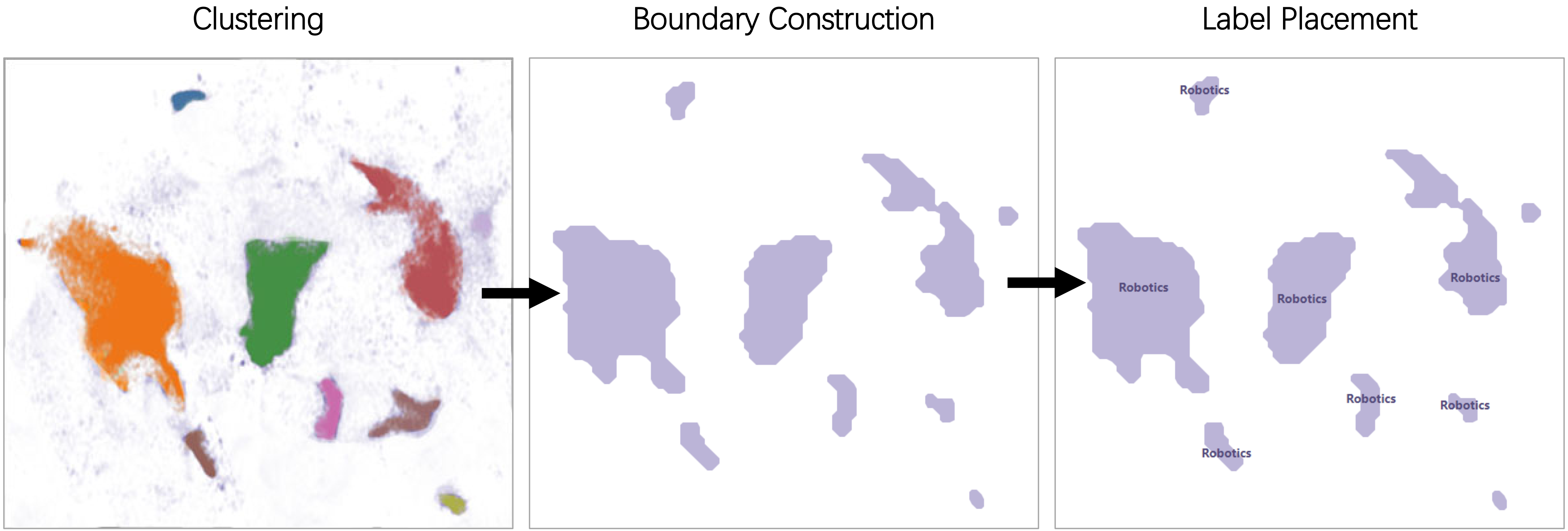


Disadvantages of KDE-based technique

The KDE-based technique can only reveal regions with high density, it can not support task which needs more flexible data filtering, for example,

– find exclusive region of classes (the proportion of the target class is higher than a threshold);
– find regions that meets the condition like "the density of the target class is twice the density of another specific class"

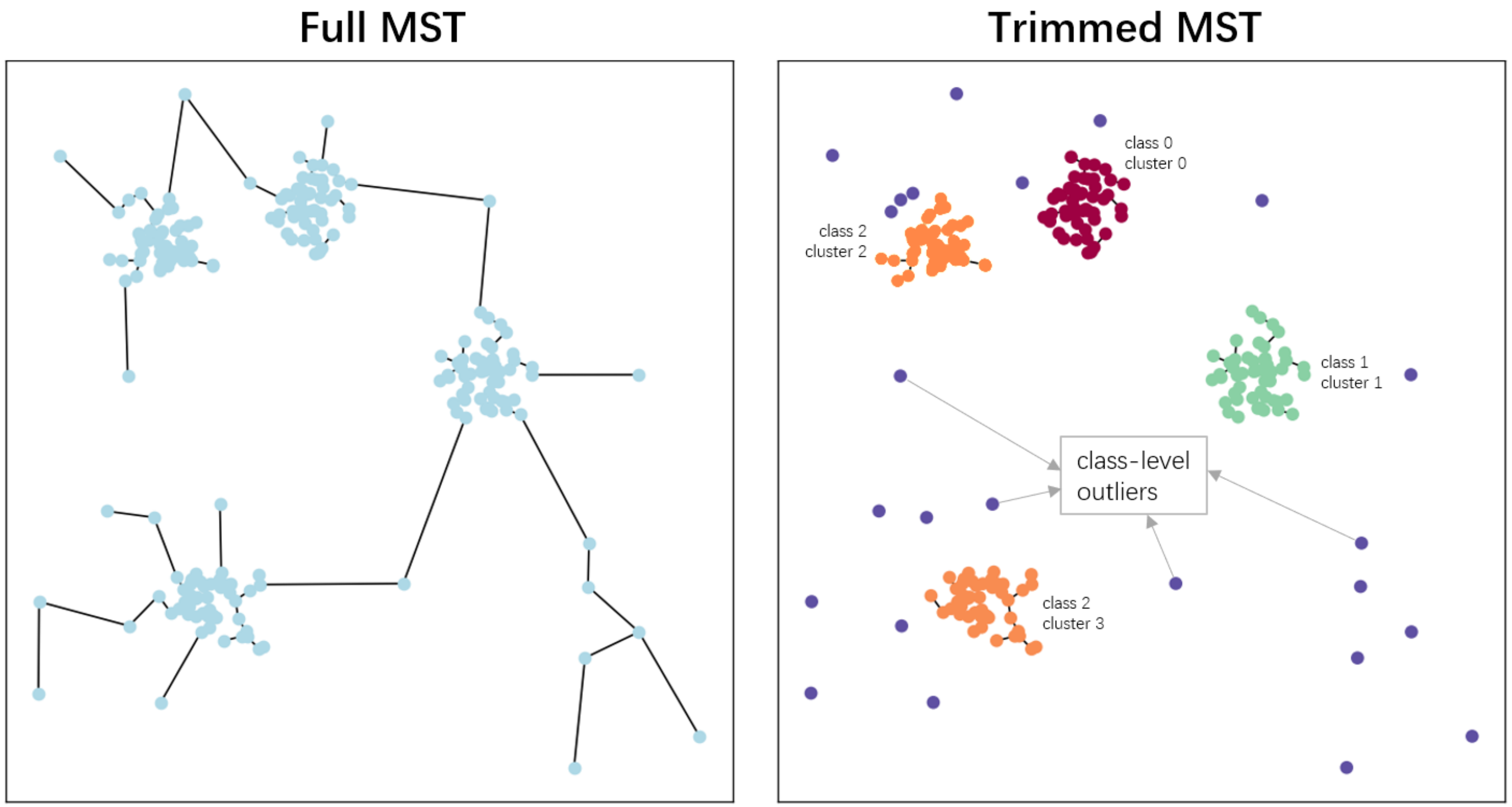
Our Method: a Three-step Framework



First Step: Clustering

Minimum Spanning Tree(MST) clustering algorithm

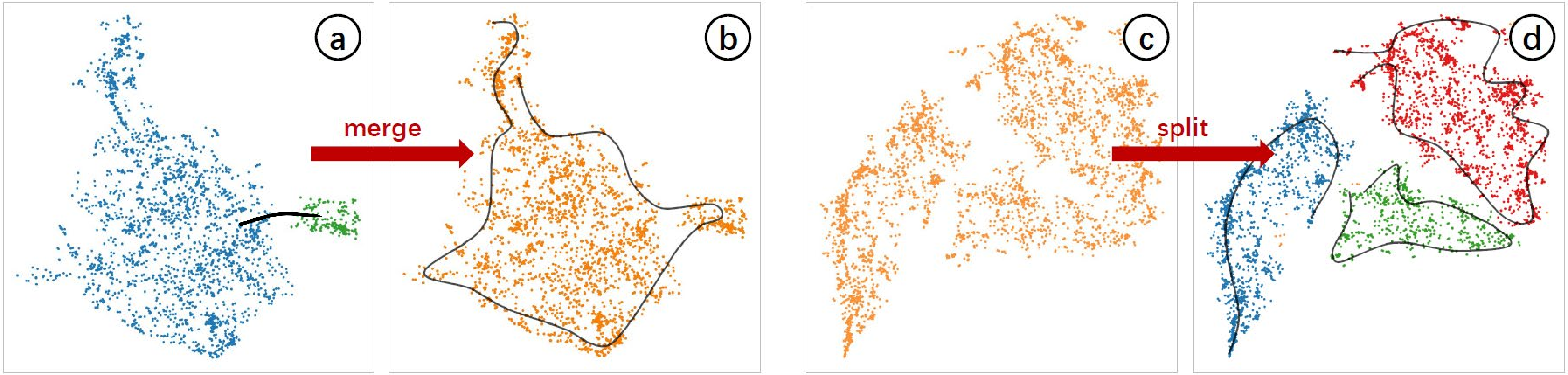
- parameter is intuitional
- time complexity is low
- can identify significant clusters
- can remove class-level outliers
- determine the number of clusters automatically
- can identify clusters with concave boundary
- supports cluster refinement with a stroke-based interaction



First Step: Stroke-based Interaction for Cluster Refinement

Merge two identified clusters

Split a identified cluster into three clusters



Second Step: Boundary Construction

Goal: make the boundaries tightly wrap most of the target data points while keeping concise and readable

Second Step: Boundary Construction

Goal: make the boundaries tightly wrap most of the target data points while keeping concise and readable

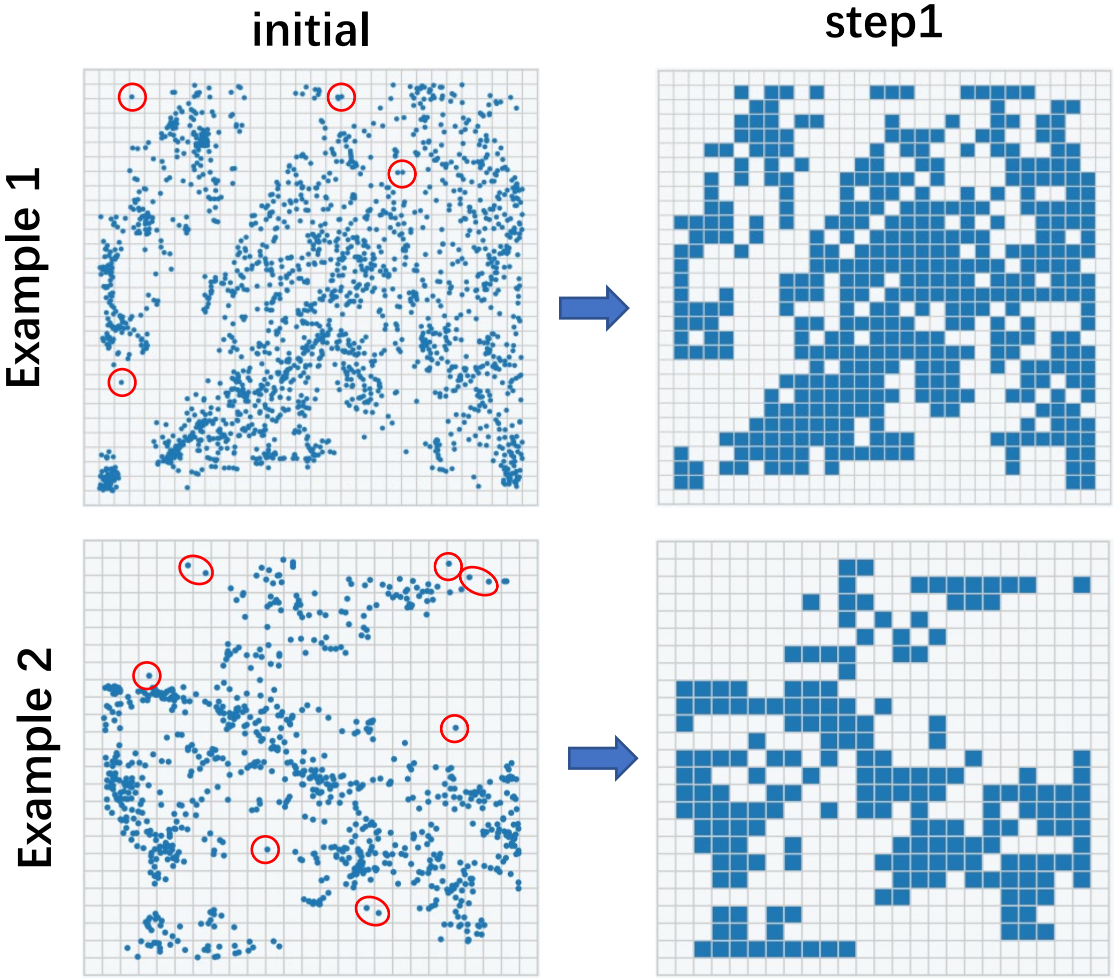
Step1: Griding and filter grids preliminarily

Filter grids by:

- the number of data points in grid
- the proportion of target class in grid
- other customized filters

Griding:

- introduces the concept of 'region', making it possible to shape different data scopes
- improves the scalability of boundary construction in terms of the number of data points



Second Step: Boundary Construction

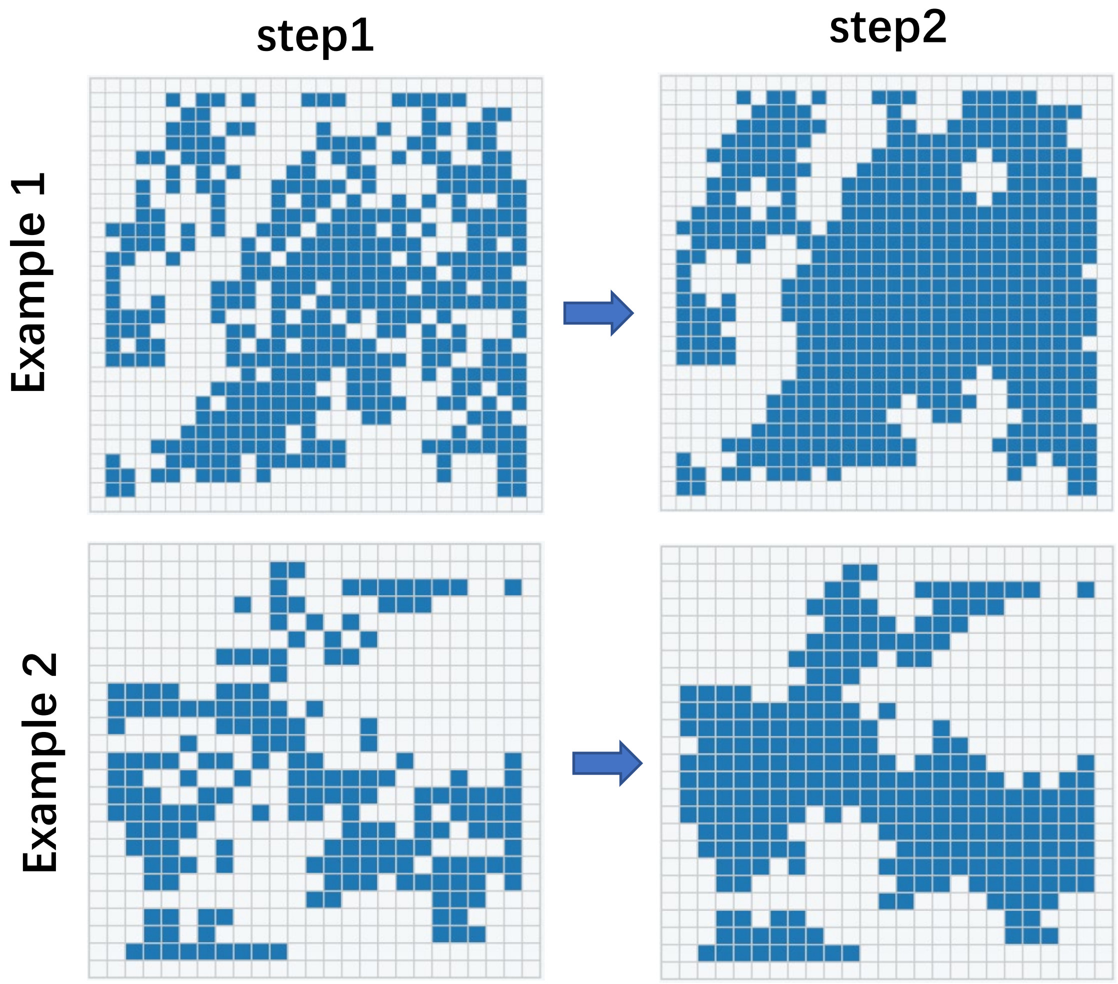
Goal: make the boundaries tightly wrap most of the target data points while keeping concise and readable

Step2: Eliminate the discrete blank grids that are interspersed with the filled grids

To rebuild contiguous distribution regions (it is necessary for boundary construction)

Morphology in image processing:

- open operation
  remove grids that form spikes and small islands
- close operation
  fill the interspersed blank grids

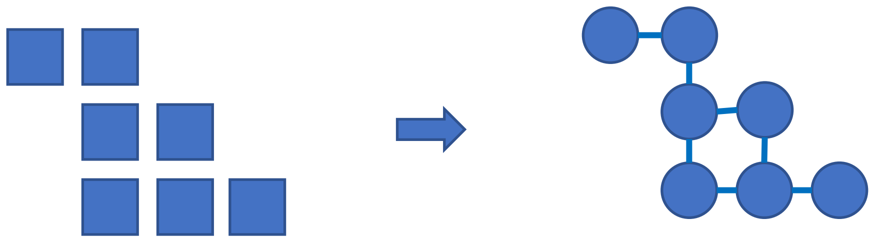


Second Step: Boundary Construction

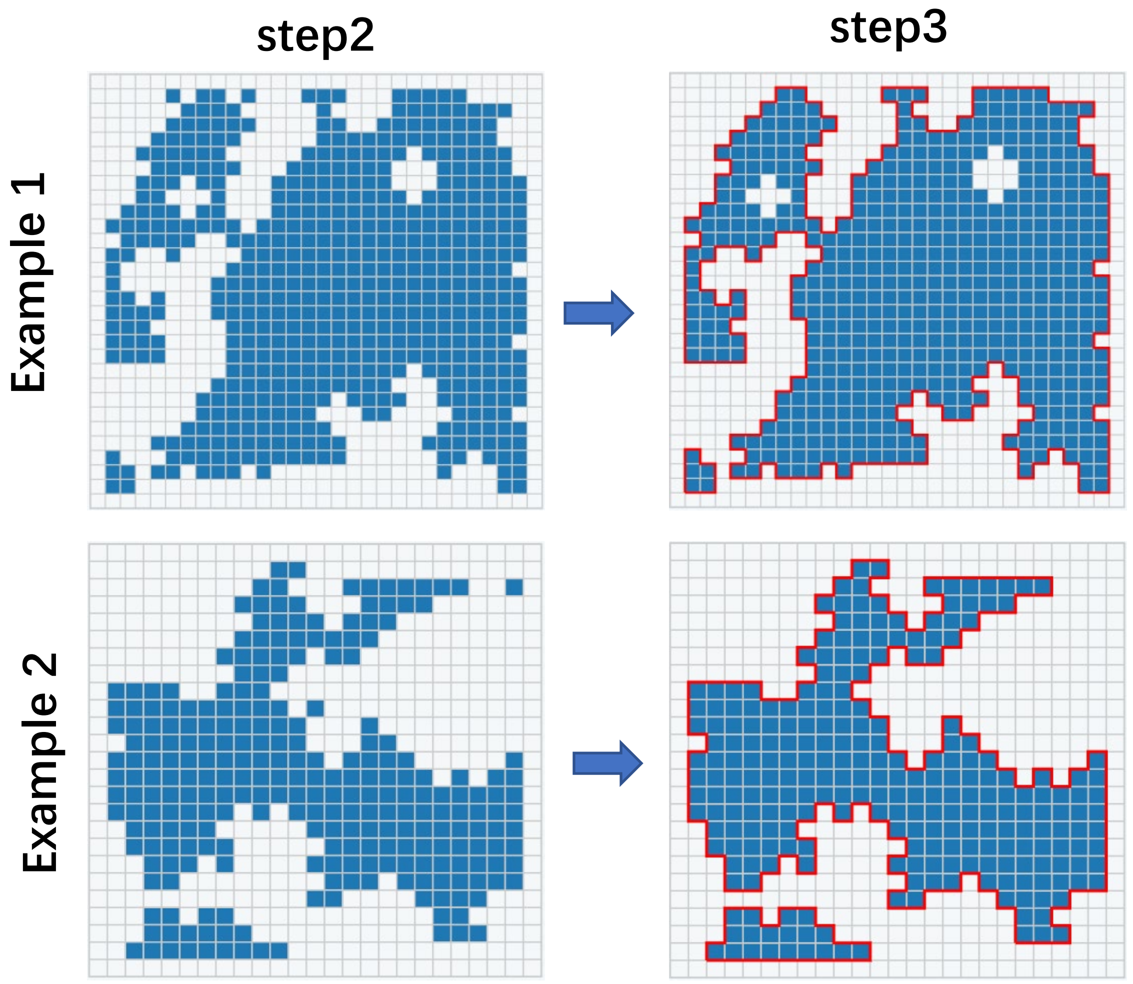
Goal: make the boundaries tightly wrap most of the target data points while keeping concise and readable

Step3: Identify continents, filter continents, and determine the boundary of continents

- continent identification ⟷ connected component identification



- filter out islands (continents whose grids less than a threshold)
- determine boundary using Moore-neighbor tracing algorithm



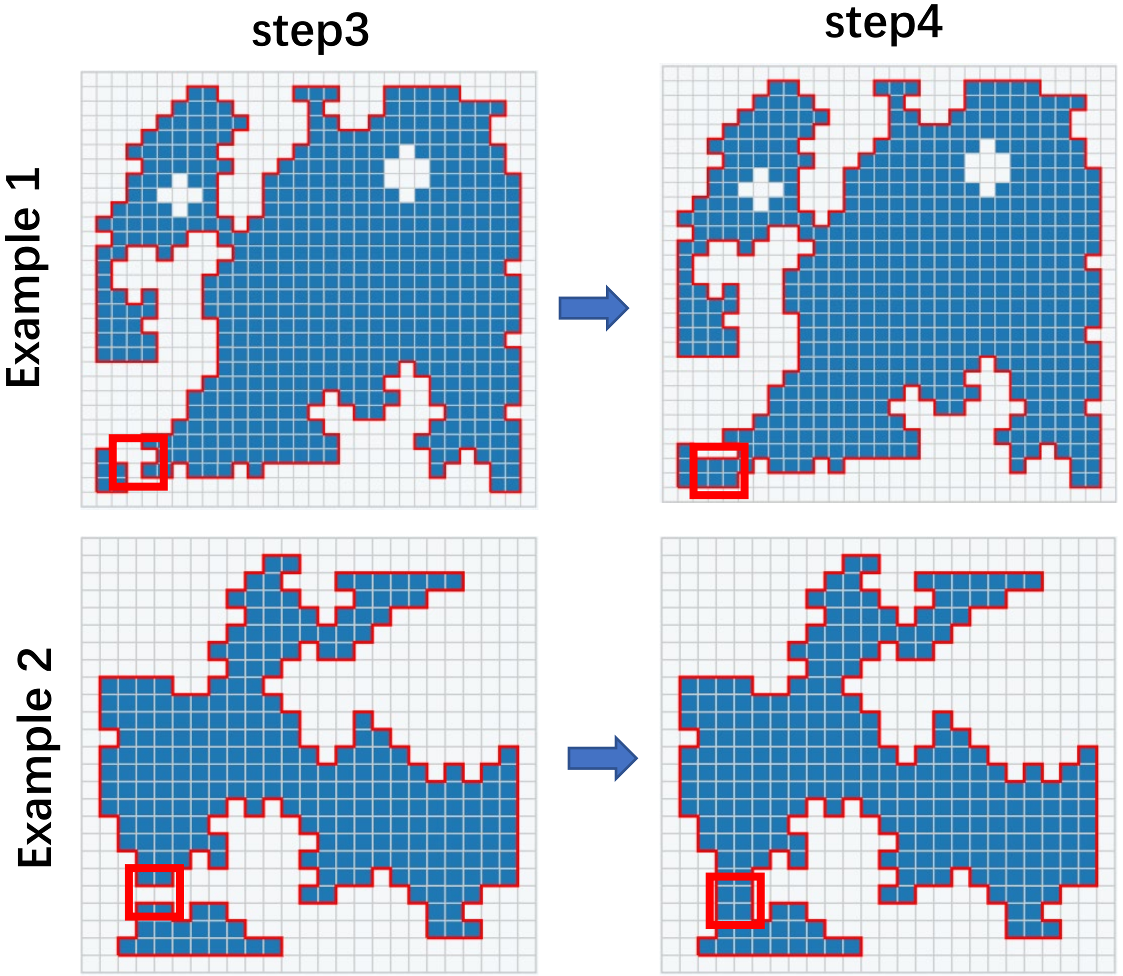
Second Step: Boundary Construction

Goal: make the boundaries tightly wrap most of the target data points while keeping concise and readable

Step4: Merge adjacent continents

To reduce the overall complexity of boundaries by eliminating unnecessary separation between continents

- Merge adjacent continents within n-jump by filling grids on n-jump paths



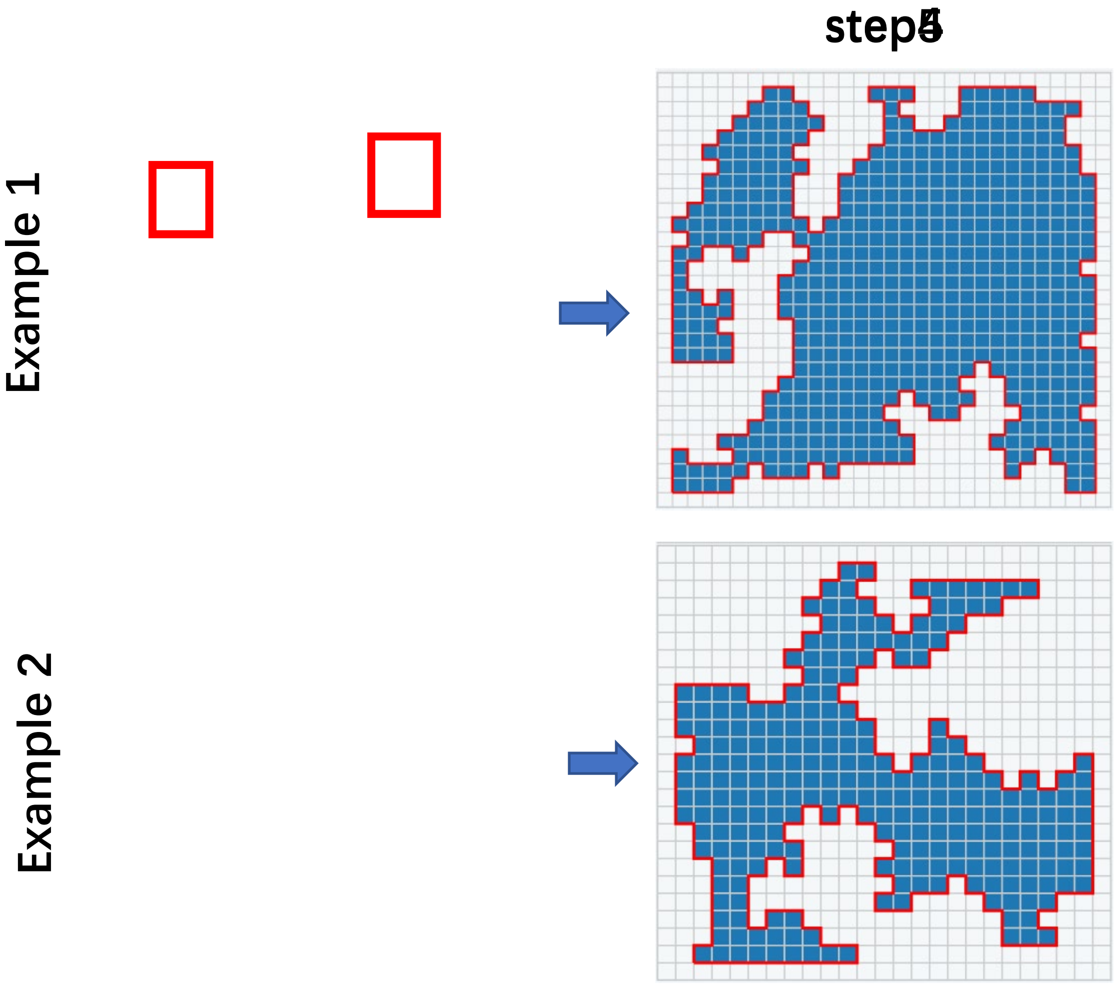
Second Step: Boundary Construction

Goal: make the boundaries tightly wrap most of the target data points while keeping concise and readable

Step5: Fill small holes

– fill small holes, because they are meaningless but increase visual complexity

– remain large holes, because they represent meaningful characteristics of data distribution



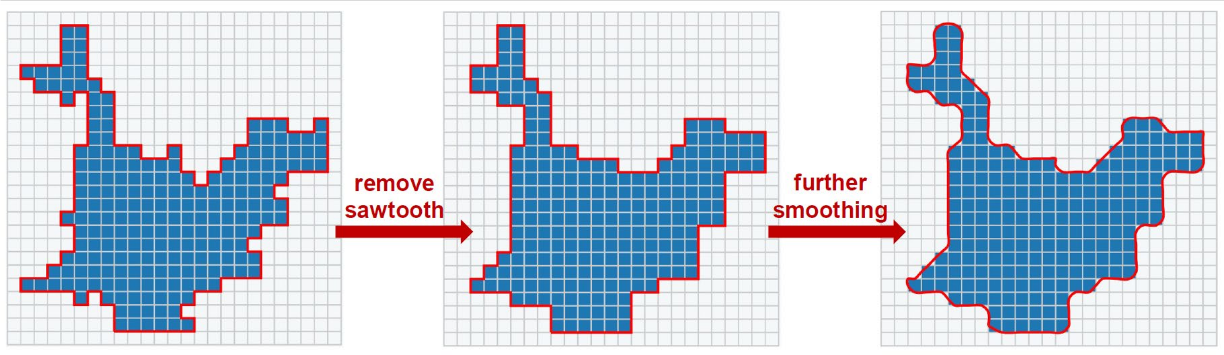
Second Step: Boundary Construction

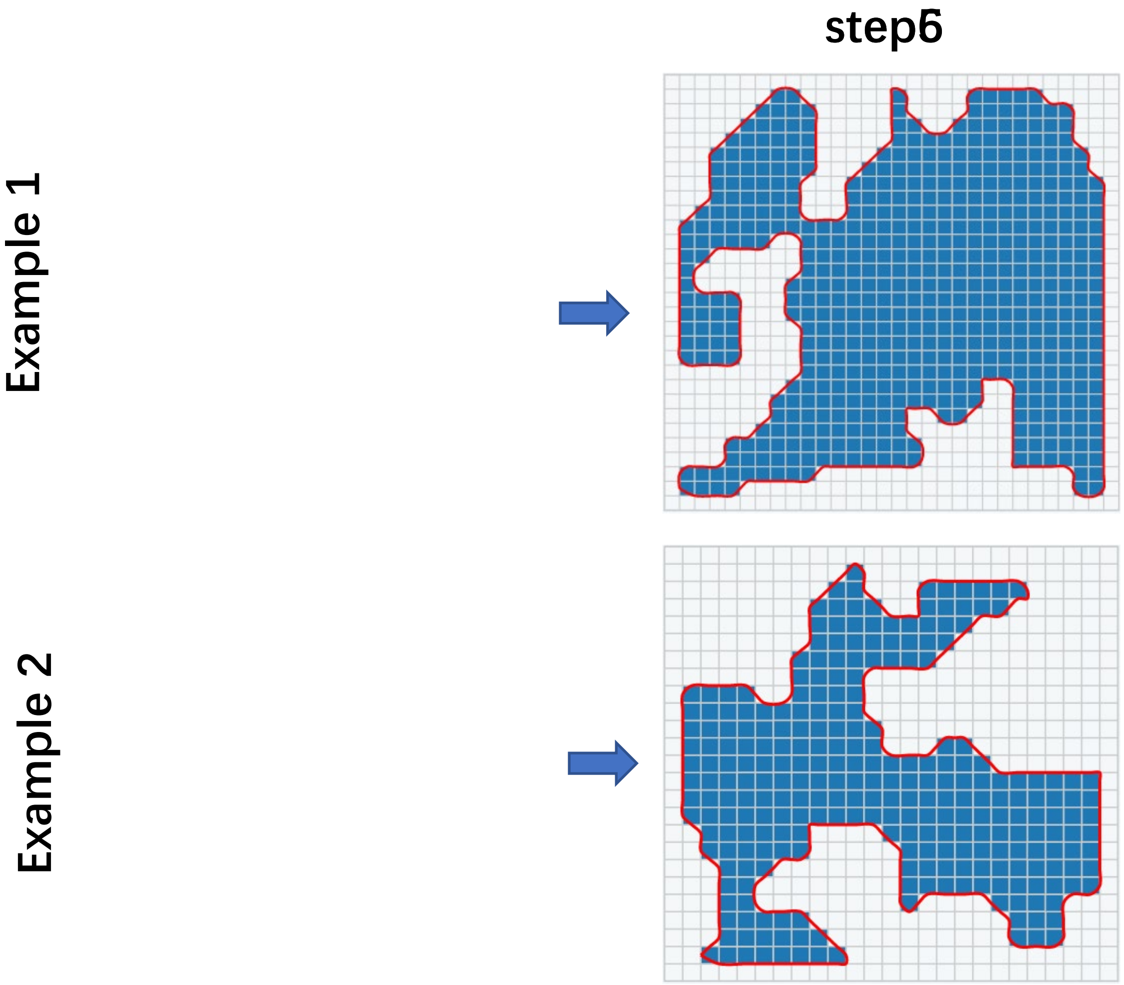
Goal: make the boundaries tightly wrap most of the target data points while keeping concise and readable

Step6: Smooth the boundary

To make boundaries more aesthetic and natural

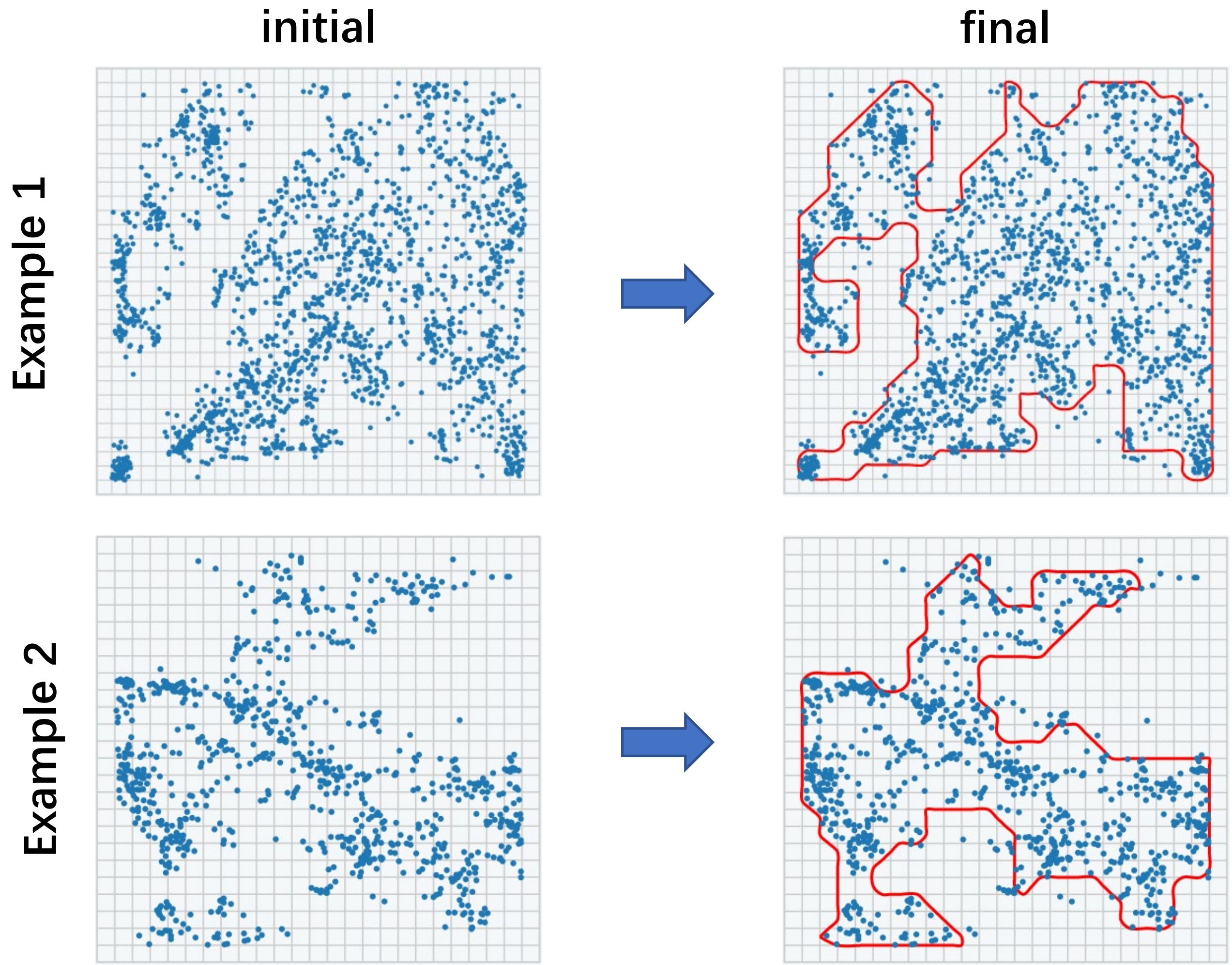
- remove sawtooth
- apply Catmullrom Curve



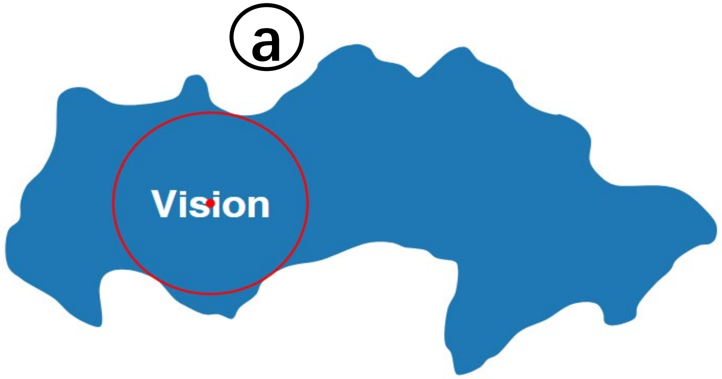


Second step: Boundary Construction

Goal: make the boundaries tightly wrap most of the target data points while keeping concise and readable



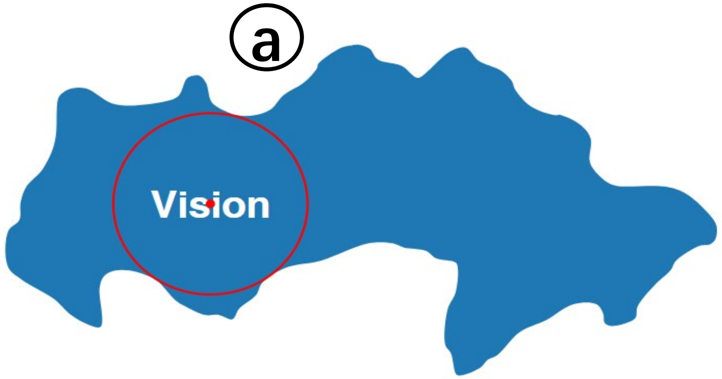
Third step: Label Placement

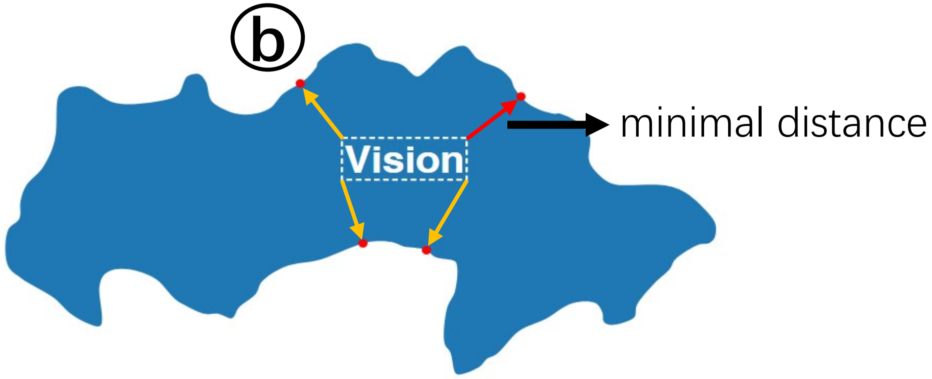


Factor:

- the boundary of cluster

Third step: Label Placement



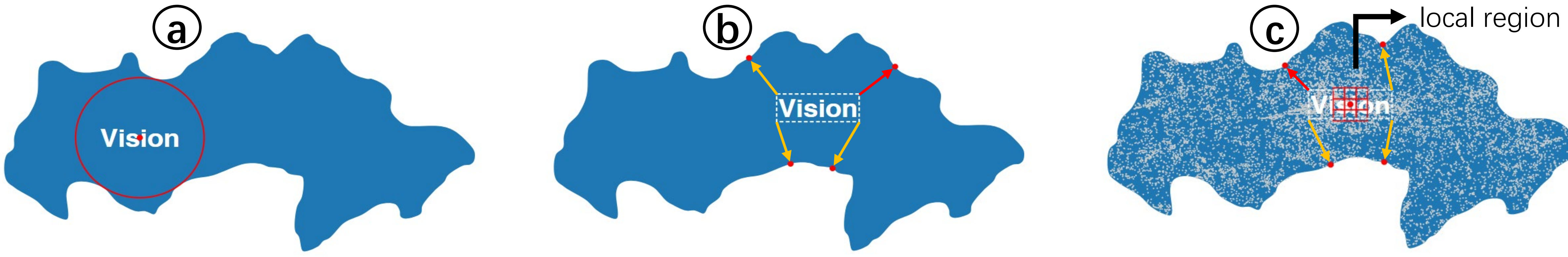


Factor:
- the boundary of cluster

Factors:
- the boundary of cluster
- the boundary of label

distance index

Third step: Label Placement



Factor:
- the boundary of cluster

Factors:
- the boundary of cluster
- the boundary of label

Factors:
- the boundary of cluster
- the boundary of label
- the density distribution of data points

density index

For each child grid of the cluster, compute a score by:
Score = α * z-score(distance index) + (1 - α) * z-score(density index)
α ∈[0, 1], can be adjusted by users

Grid with highest score is the final placement position

Qualitative Evaluation of Boundary Construction

Data
- 46 thousand papers
- 30 classes
- 98 clusters

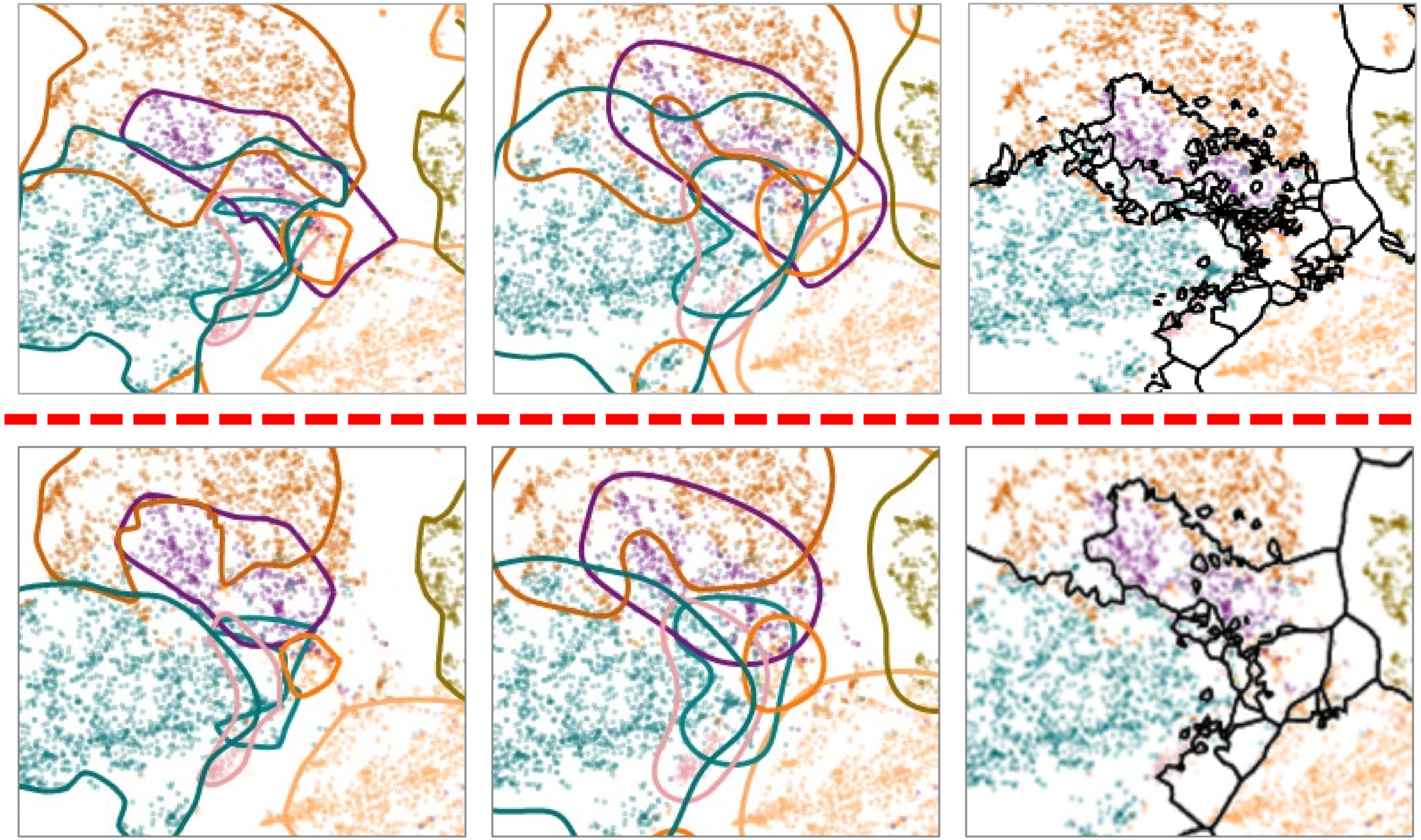
Compared methods
- Concave hull (hull-based technique)
- Bubble Sets (set visualization based technique)
- Gmap (tessellation-based technique)

Qualitative Evaluation of Boundary Construction

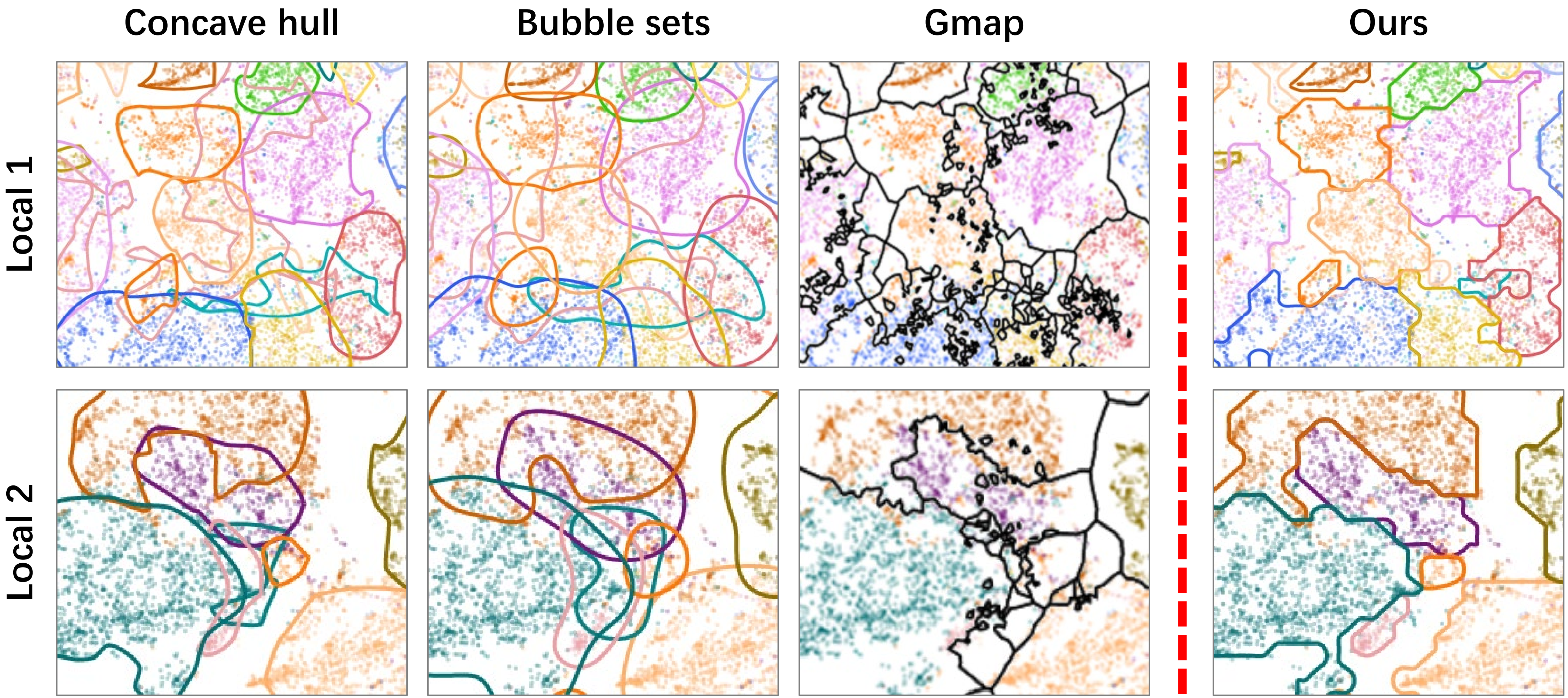
Concave hull | Bubble sets | Gmap

without outlier removal

with outlier removal



Qualitative Evaluation of Boundary Construction



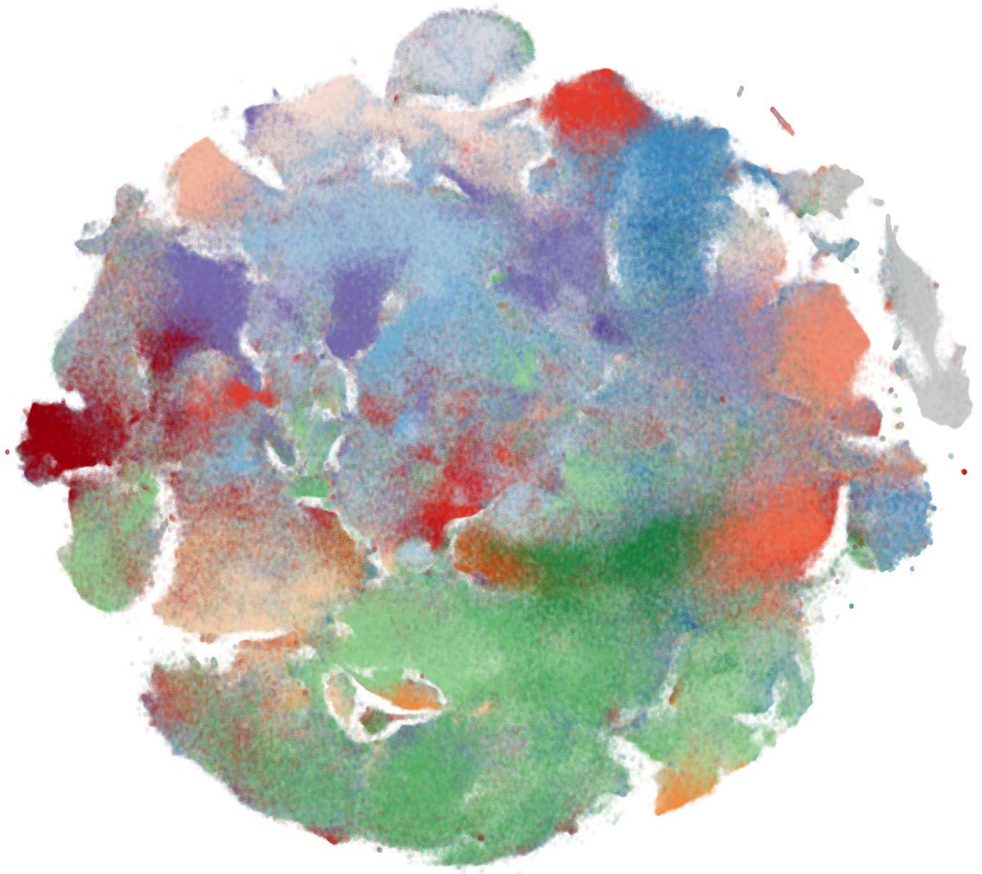
Case：Make a Turbid Science Map Readable

Data
- 4.1 million papers
- 38 research areas(classes)
- from DBLP and Microsoft Academia Graph

Overdraw
- cannot perceive the distribution of each class
- can not determine the overlaps between classes
- lacks semantics

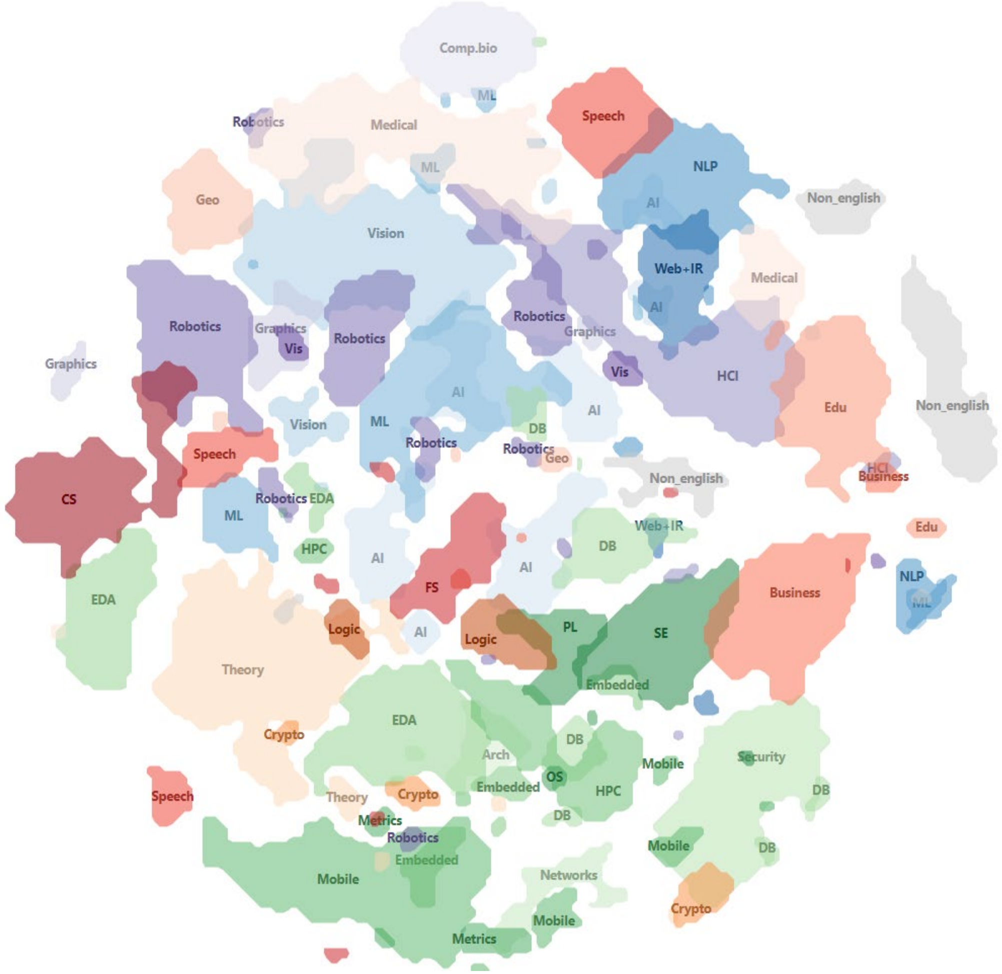
initial scatterplot

Case：Make a Turbid Science Map Readable

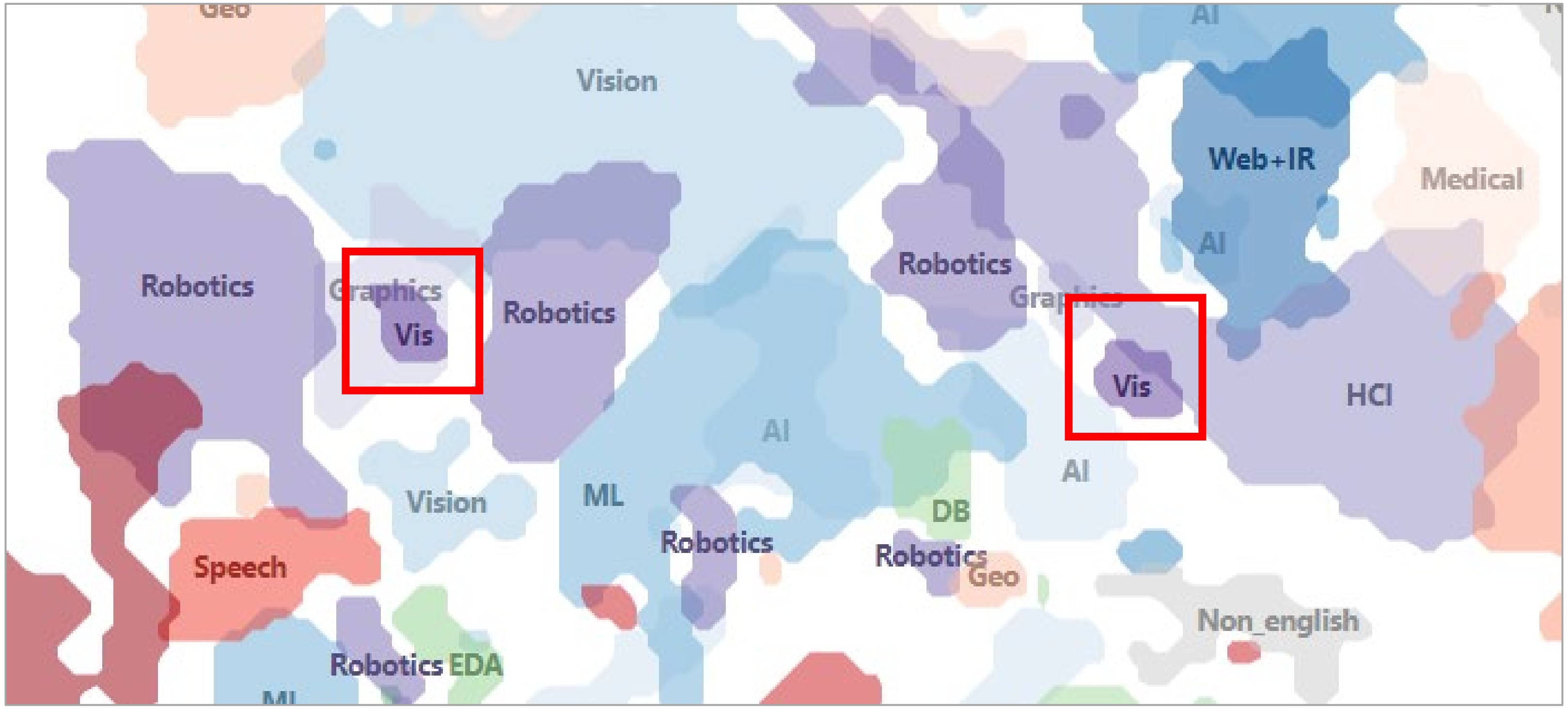


our three-step framework

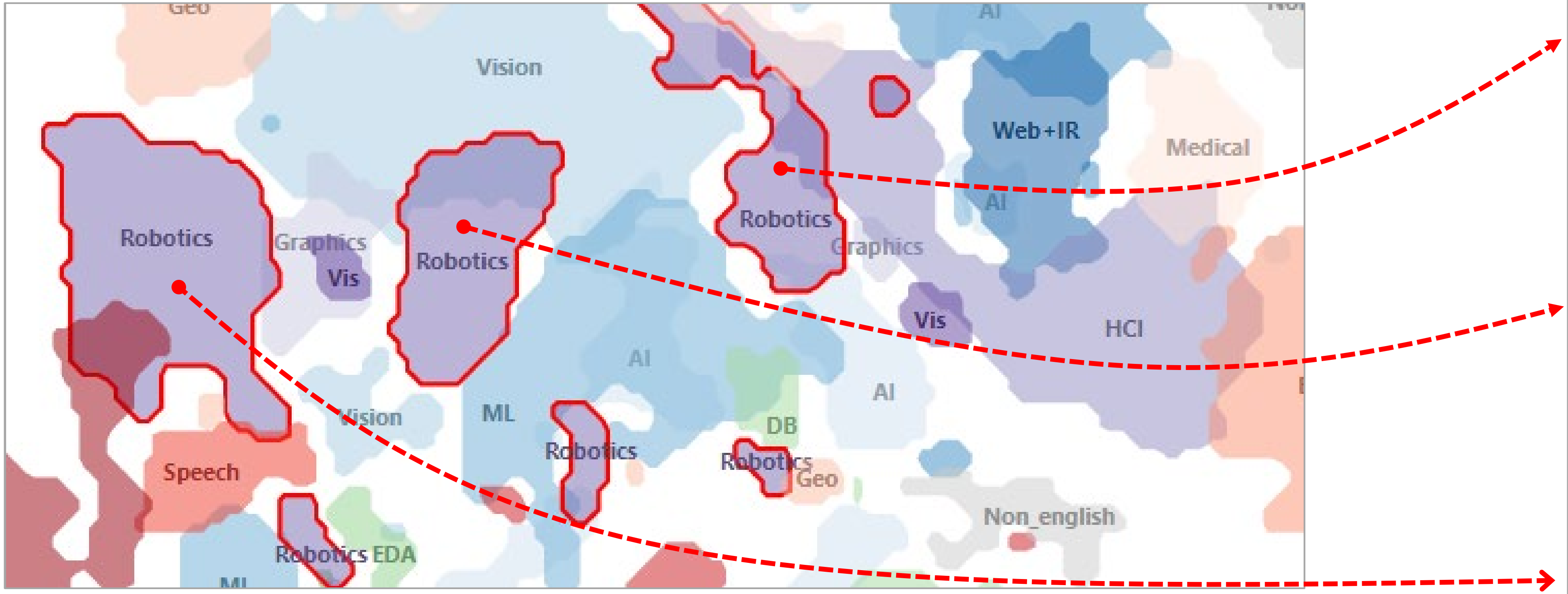
handled scatterplot
314 clusters



Case：Make a Turbid Science Map Readable



Case：Make a Turbid Science Map Readable



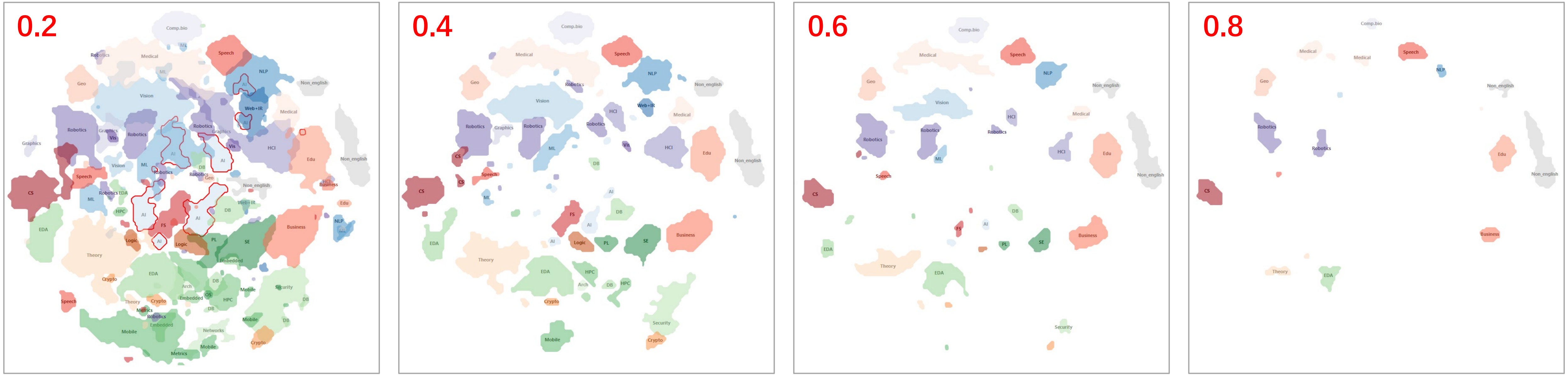
HCI
human_robot_interaction
mobile_robot
humanoid_robot
human_operator
service_robot
video_data
virtual_environment

Vision
mobile_robot
real_time
particle_filter
indoor_environment
stereo_vision
kalman_filter
obstacle_detection

Control System
neural_network
pid_controller
fault_diagnosis
fault_detection
induction_motor
nonlinear_system
fuzzy_controller

Case: Make a Turbid Science Map Readable

Build different scopes by adjusting proportion threshold (in the first step of boundary construction)



Contributions

- We propose **a three-step framework** that highlights class-level information in large-scale multi-class scatterplots by constructing boundaries for classes and then placing a text label for each boundary.

- We design **a stroke-based cluster refinement interaction** that allows the user to quickly correct clusters identified by the algorithm, or materialize the clusters in his or her mind.

- We propose **a grid-based and controllable construction pipeline** which alleviates the overdraw problem and allows the boundary to strike a balance between simplicity and accurate delineation of the distribution.

Thanks