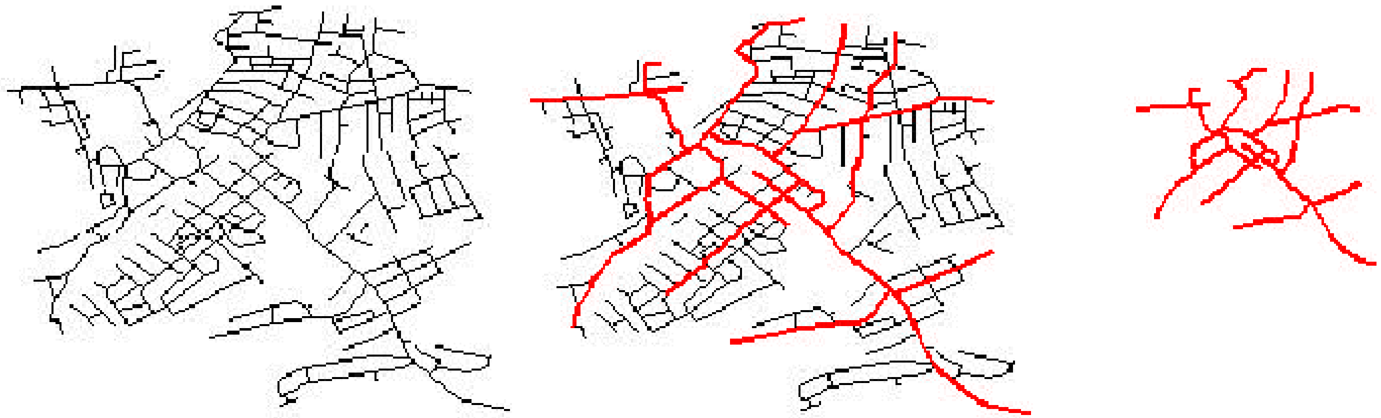
Selection of Streets from a Network Using Self-Organizing Maps

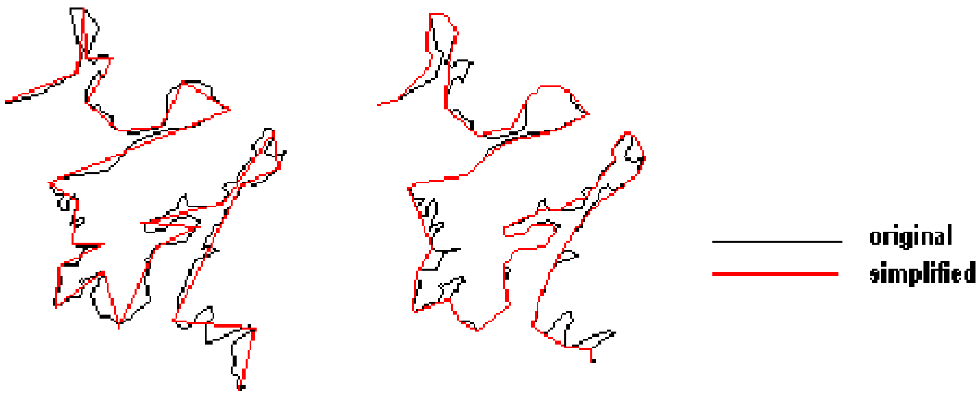
Bin Jiang, University of Gävle
Lars Harrie, Lund University

Contents

- Map generalization
- Self-organizing maps: principle and algorithm
- A structural view of street networks
- Multiple attributes of streets involving semantic, geometric and topological properties
- A case study applied to Munich dataset
- Conclusion and future work

Two kinds of map generalisation: model generalisation and graphic generalisation





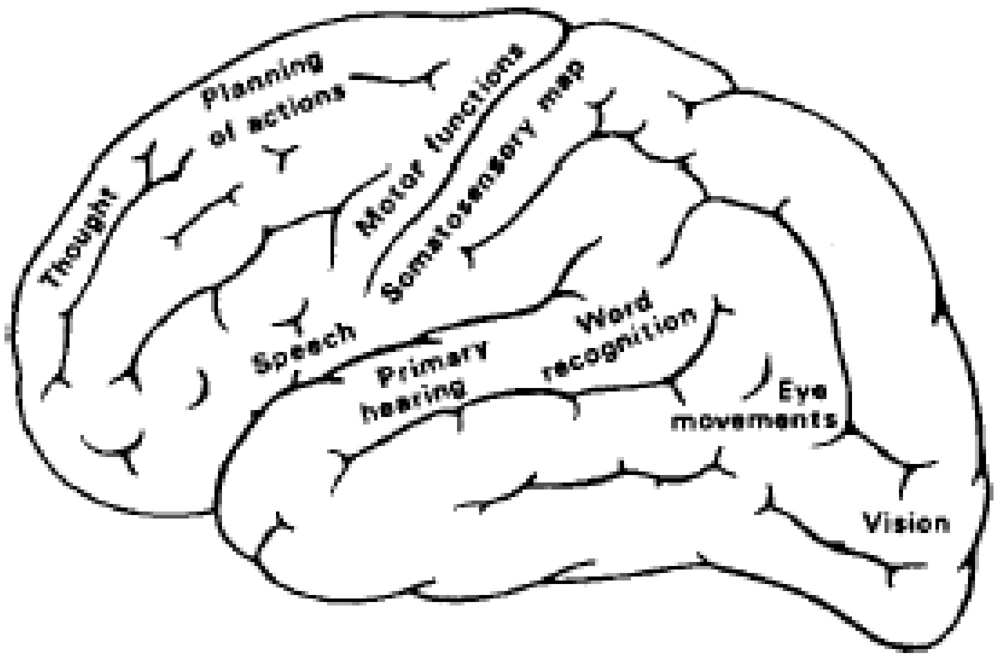
Why self-organizing maps?

- SOM is a well-developed neural netowrk technique for data clustering and visualization, developed by Teuvo Kohonen, a finish professor.
- The SOM process is done with input data alone without the presence of an external teacher - an unsupervised learning, so self-organizing.

Why self-organizing maps?

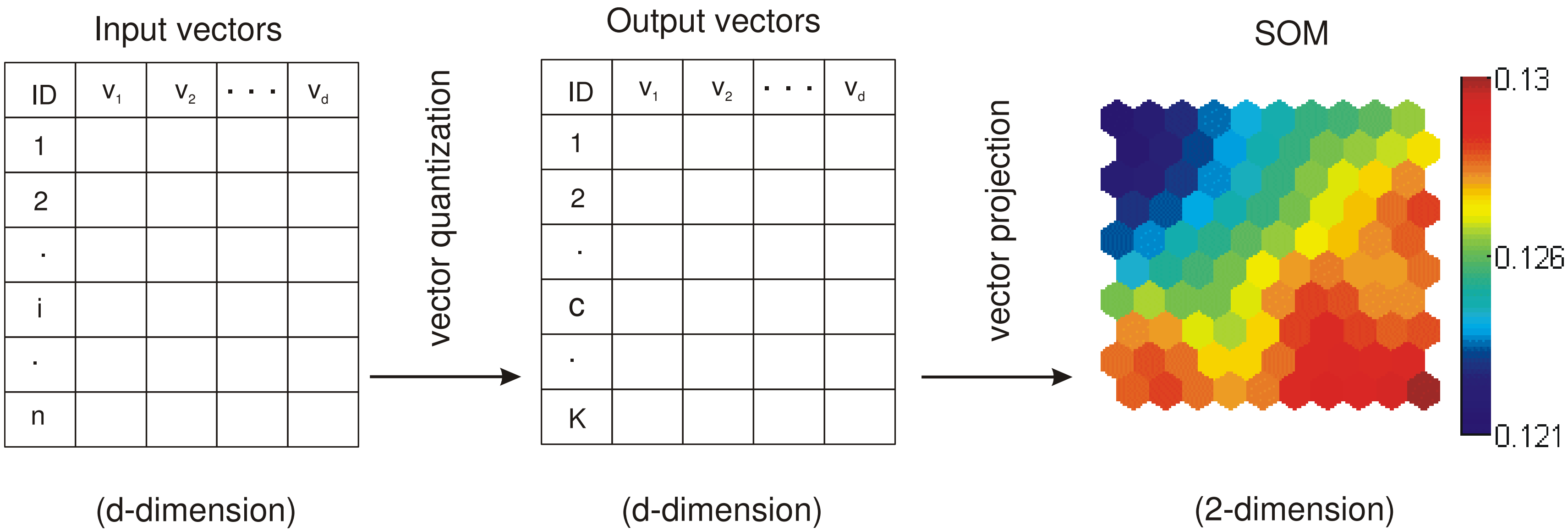
- SOM projects a large data set of a high dimension into a low dimension while retaining the initial pattern of data samples, somehow like mapping or map projection process.
- SOM resembles a geogrpahic map concerning the distribution of phenomena, in particular referring first law of geography: *everything is related to everything else, but near things are more related to each other* (Tobler 1970).

- SOM is a realistic model of the biological brain function (brain maps: different brain areas connected to different sensorial modality)



A. Brain areas (Kohonen, 1995)

Illustration of SOM principle



The best matching unit

- Let *x* be input vector, and *m* be output vectors, firstly the best matching unit is obtained by (Equation 1)

$$\| x - m_c \| = \min_i \{ \| x - m_i \| \}$$

Updating other neurons

- Secondly the matching unit or winning neuron and other output vectors in its neighbourhood are updated to be closer to *x* in the input vector space (Equation 2),

$$m_i(t+1) = m_i(t) + \alpha(t)h_{ci}(\mathrm{t})[x(t) - m_i(t)] \qquad \text{for } i \in N_c(t)$$
$$m_i(t+1) = m_i(t) \qquad \text{for } i \notin N_c(t)$$

where *x* is a sample vector randomly taken from input vectors, mi(t) is the output vector for any neuron *i* within the neighbourhood *Nc(t),* and $\alpha(t)$ and $h_{ci}(t)$ are the learning rate function and neighbourhood kernel function respectively.

Step 1: Define input vectors in particular their multiple variables that determine an attribute space.

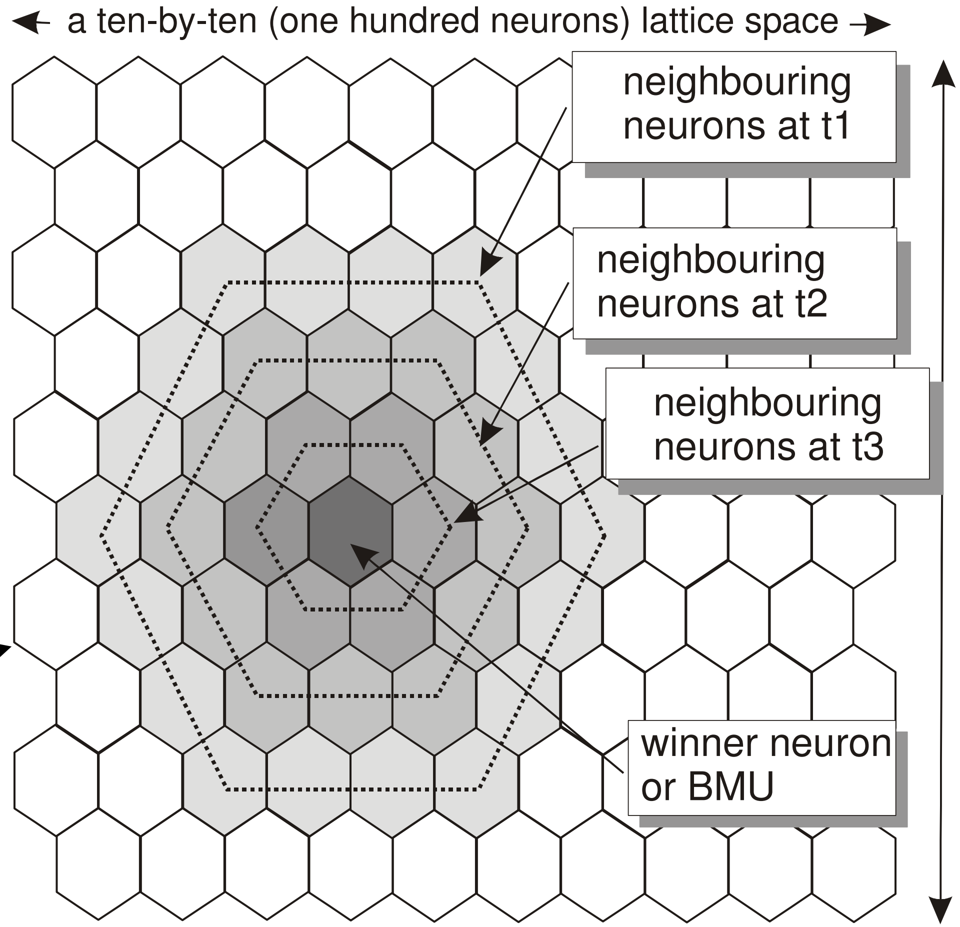
Step 2: Define the size, dimensionality, and shape of a SOM to be used.

Step 3: Initialize output vectors m randomly or linearly.

Step 4: Define the parameters that control the training process involving map lattice, neighbourhood, and training rate functions.

Neighbourhood

- the size of the neighbourhood $N_c(t)$ reduces slowly as a function of time, i.e. it starts with fairly large neighbourhoods and ends with small ones.
- t1<t2<t3

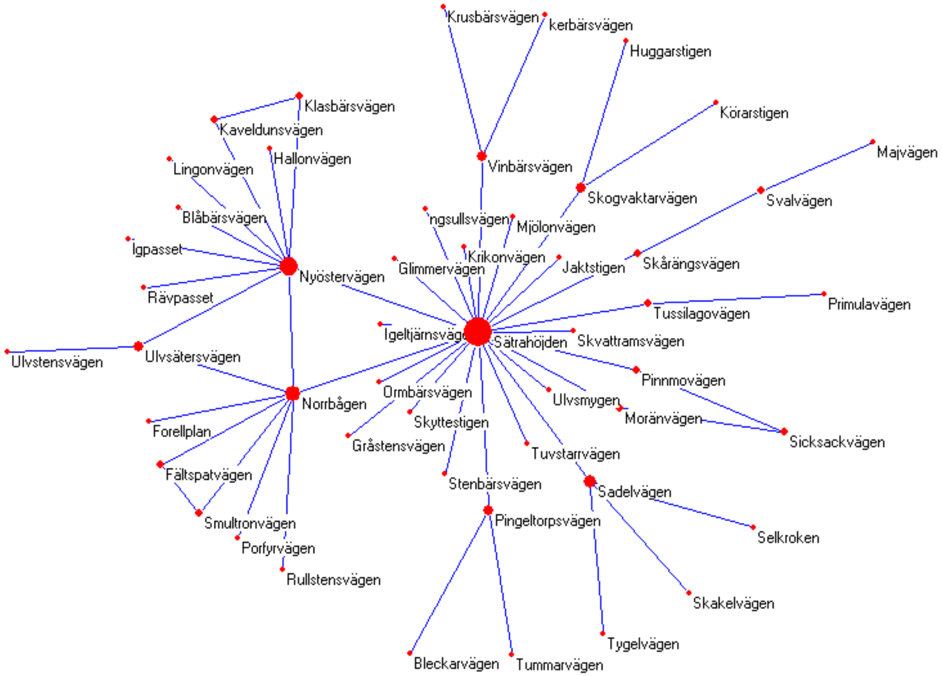


Step 5: Select one input vector x, and determine its Best-Matching Unit (BMU) or winning neuron using equation [1].

Step 6: Update the attributes of the winning neuron and all those neurons within the neighbourhood of the winning neuron, otherwise leave alone (c.f. equation [2]).

Step 7: Repeat steps 5 to 6 for a very large number of times (training length) till a convergence is reached.

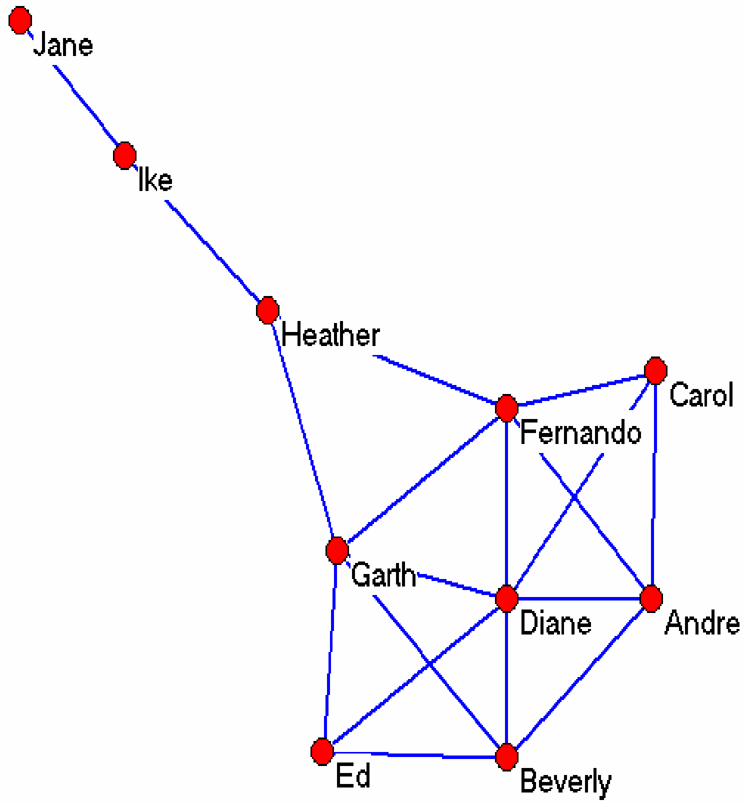
• A structural view of street networks



- Multiple properties that govern the importance of streets within a network
  - Semantic properties: functional classes (highway, motorway and normal streets), and speed limit
  - Geometric properties: Length and width
  - Topogical properties: degree centrality, closeness centrality, and betweenness centrality;

- Centrality measures:
  - Degree: Number of direct ties or links for each node within a graph;
  - Betweeness: A node with high betweenness has great influence over what flows in the network;
  - Closeness: Nodes with high closeness have the shortest paths to all other nodes– they are close to everyone else.

Centrality of nodes with kit-network (David Krackhardt)



Node	Degree	Betweenness	Closeness
Carol	3	0.00	0.50
Andre	4	0.02	0.53
Fernando	5	0.23	0.64
Diane	6	0.10	0.60
Beverly	4	0.02	0.53
Ed	3	0.00	0.50
Garth	5	0.23	0.64
Heather	3	0.39	0.60
Ike	2	0.22	0.43
Jane	1	0.00	0.31

- Principles of the SOM-based selection approach
  - Training process and creation of a SOM
  - Linkage between SOM and a street network
  - Selection or elimination of streets

SOM table

SOM-ID	v_1	v_2	. . .	v_d
1				
2				
.				
c				
.				
K				

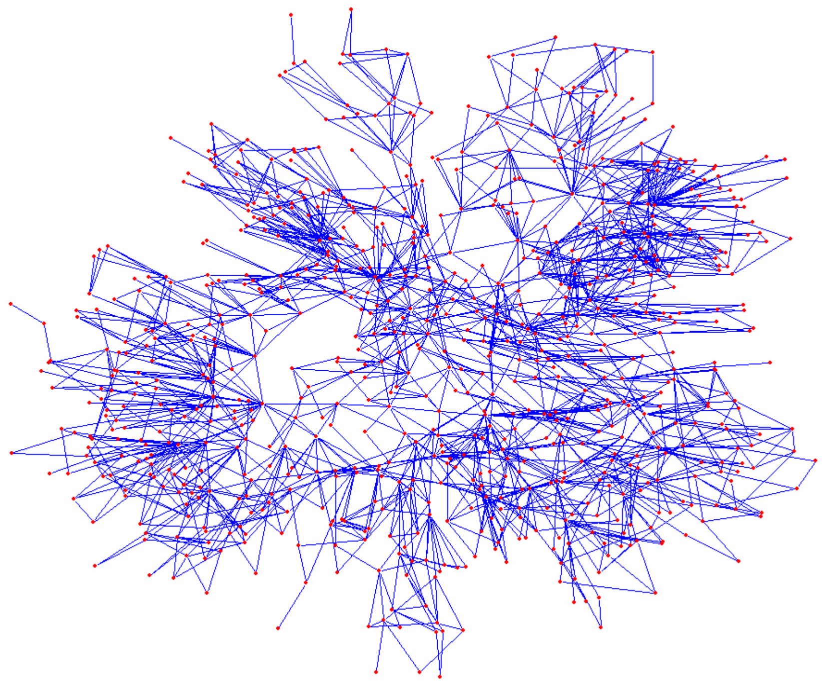
LINK table

BMU	street-ID
	1
c	2
c	i
	n

NETWORK table

street-ID	v_1	v_2	. . .	v_d
1				
2				
.				
i				
.				
n				

• A case study with Munich network (785 streets)



Training specification

Parameter	Value
Size (m)	100
Dimensionality	2
Shape	Sheet
Map lattice	Hexagonal
Neighbourhood	Gaussian
Learning rate (α)	$\alpha(t) = \alpha_0 /(1 + 100t / T)$
Initial learning rate (α_0)	0.5 for the coarse period 0.05 for the fine period
Training length in epochs (T)	0.51 epochs for the coarse period 2.04 epochs for the fine period
Initial neighbourhood radius (σ_0)	5
Final neighbourhood radius	1.25 for the coarse period 1 for the fine period

Weight vector [1, 1, 1, 2, 2, 2, 3] for the seven attributes

[degree, closeness, betweenness, length, lanes, speed, class]

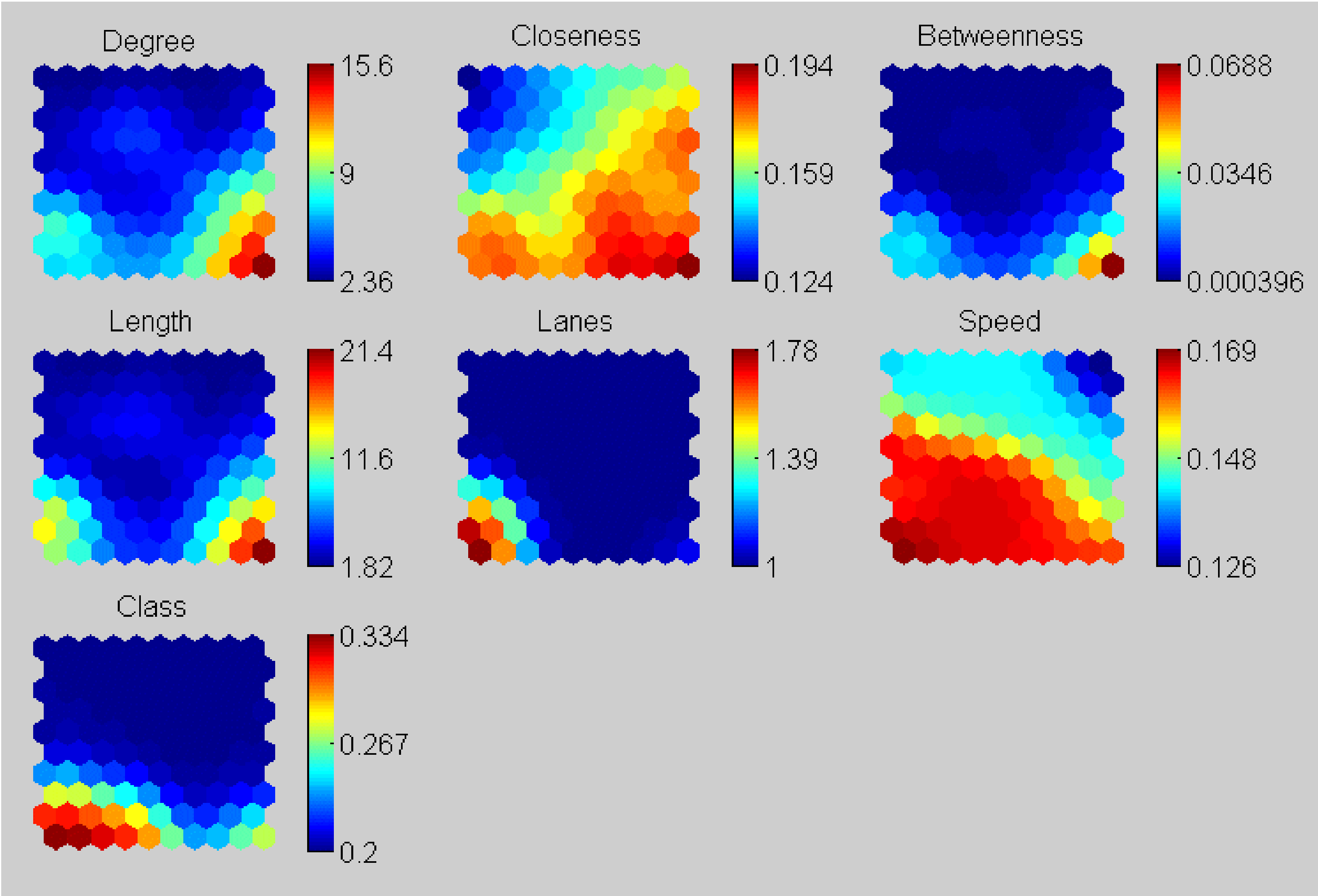
- Input streets (first 10 street of the Munich network)

Street-name	Degree	Closeness	Betweenness	Length	Lanes	Speed	Class
ACKER	2	0.149	0.000	2.42	1	0.143	0.2
ADALBERT	11	0.185	0.012	13.81	1	0.145	0.2
ADAM-ERMINGER-	2	0.117	0.000	0.55	1	0.143	0.2
ADELGUNDEN	6	0.187	0.003	3.14	1	0.152	0.2
ADELHEID	9	0.169	0.006	6.04	1	0.140	0.2
ADLZREITER	3	0.153	0.003	4.17	1	0.154	0.2
ADOLF-KOLPING-	4	0.153	0.000	4.37	1	0.160	0.2
AGILOLFINGER	4	0.128	0.001	4.71	1	0.143	0.2
AGNES	10	0.168	0.003	12.85	1	0.143	0.2
AIGNER	4	0.141	0.003	3.35	1	0.143	0.2

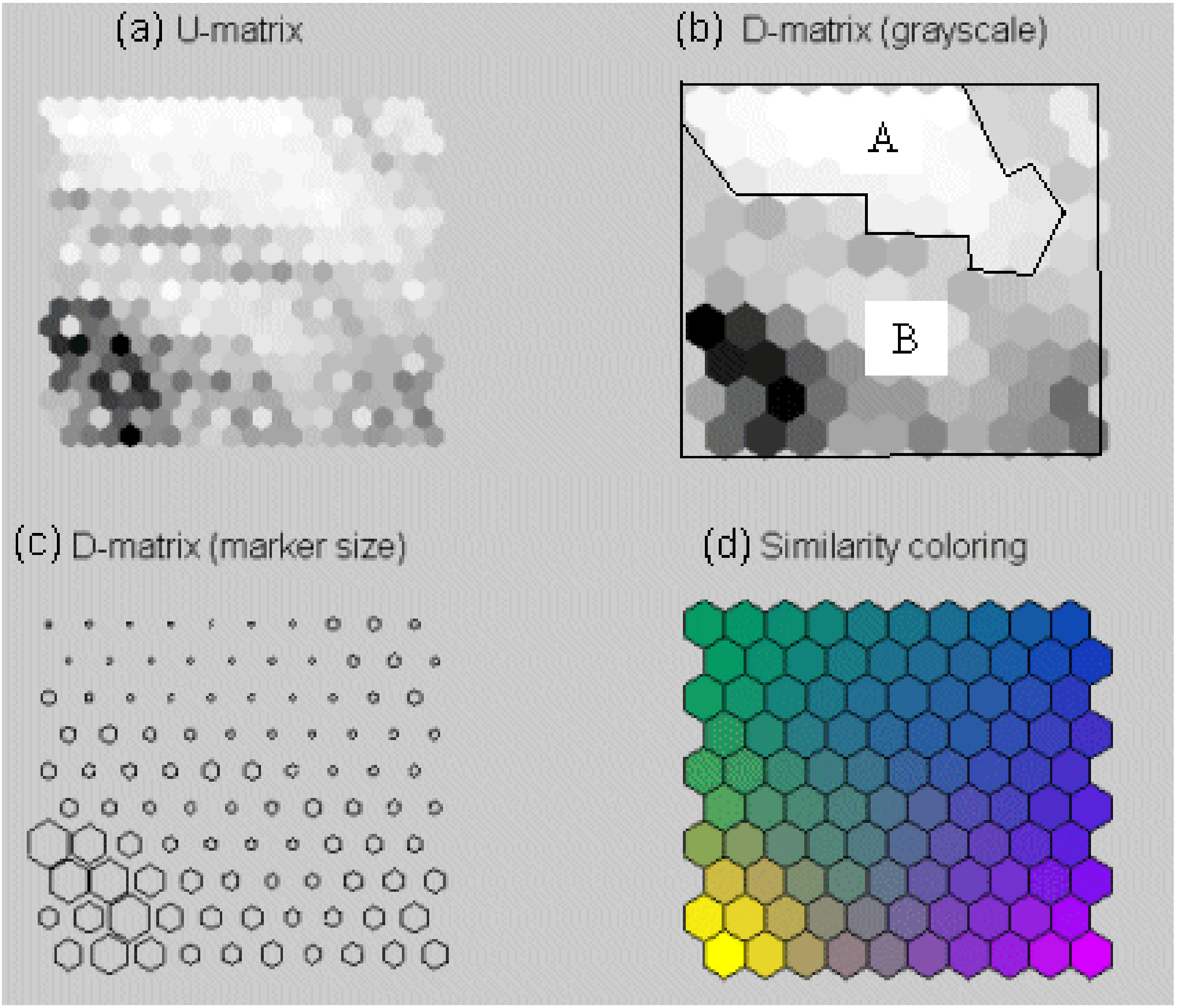
- Output weight vectors (first 10 neurons with the SOM)

Neuron-ID	Degree	Closeness	Betweenness	Length	Lanes	Speed	Class
1	2.482	0.125	0.000	2.2	1.000	0.141	0.200
2	2.625	0.129	0.000	2.4	1.000	0.143	0.201
3	2.852	0.131	0.001	2.5	1.000	0.149	0.202
4	3.034	0.138	0.001	2.7	1.003	0.158	0.203
5	3.114	0.141	0.001	2.9	1.022	0.164	0.204
6	4.095	0.148	0.004	4.5	1.103	0.164	0.212
7	6.233	0.161	0.011	8.9	1.325	0.163	0.234
8	8.071	0.175	0.021	12.6	1.544	0.164	0.279
9	7.701	0.177	0.023	13.9	1.740	0.168	0.313
10	6.858	0.178	0.023	12.1	1.781	0.169	0.334

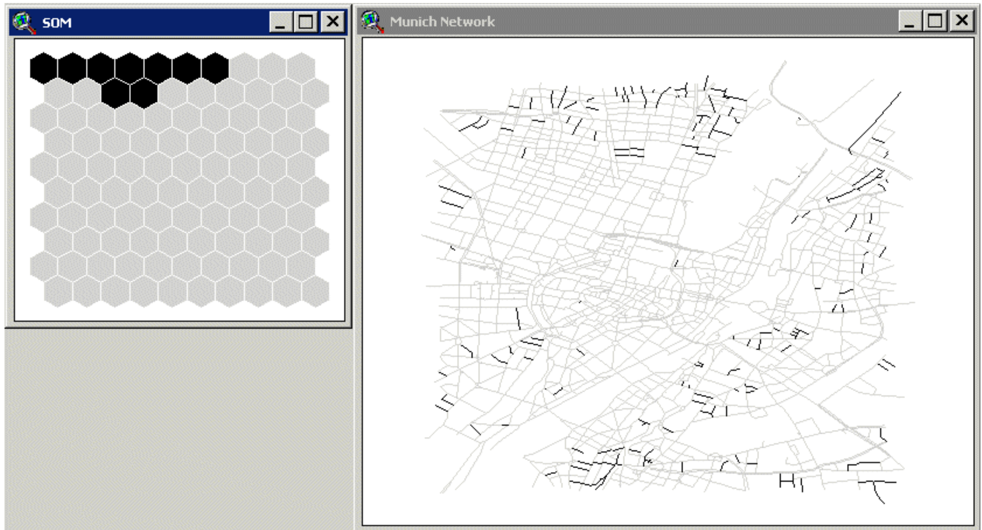
- Visualization of the SOM

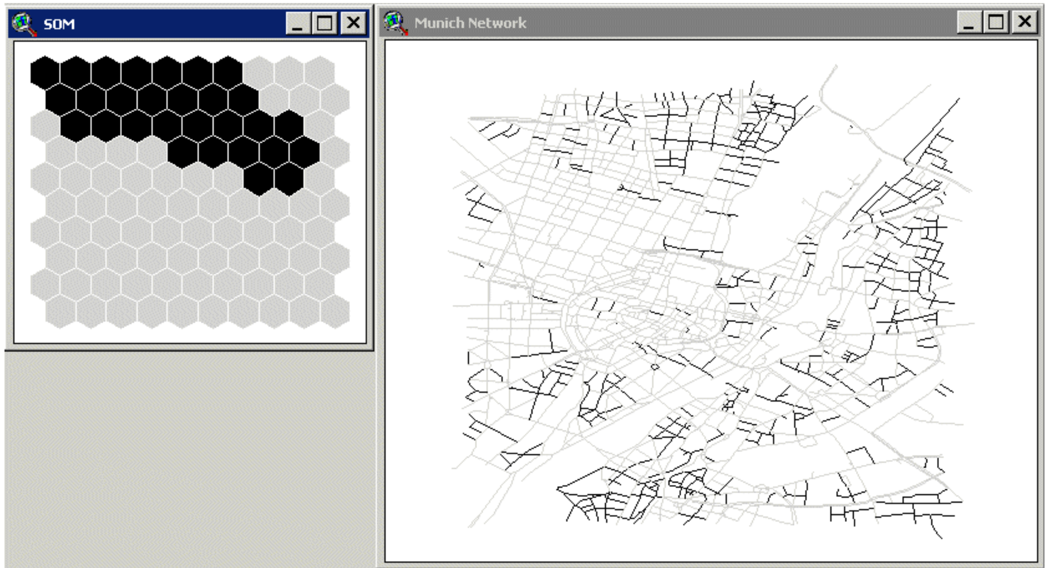


- U-matrix visualizations of the SOM

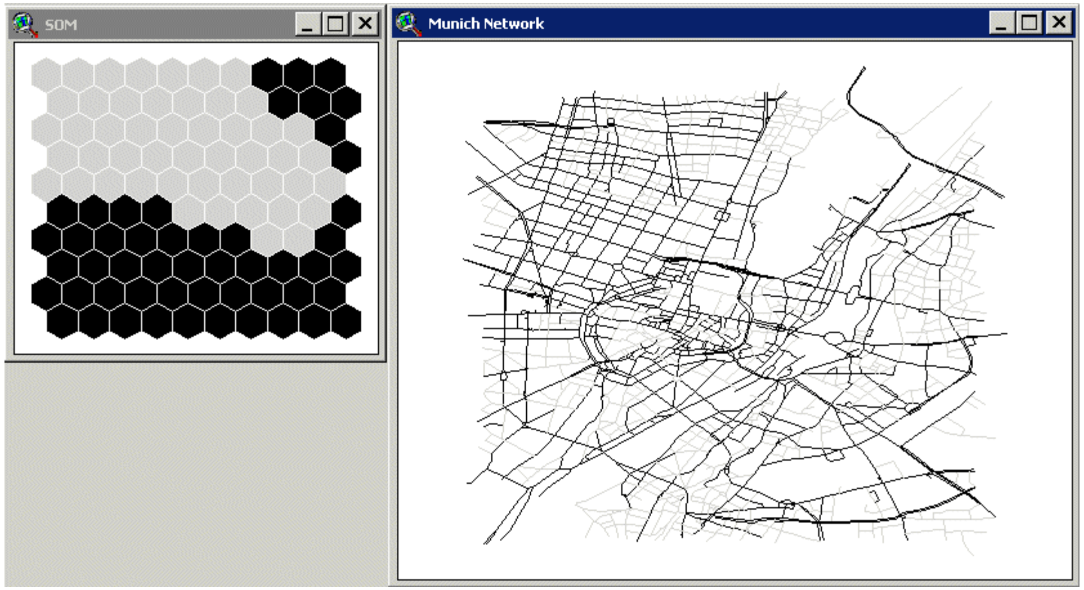


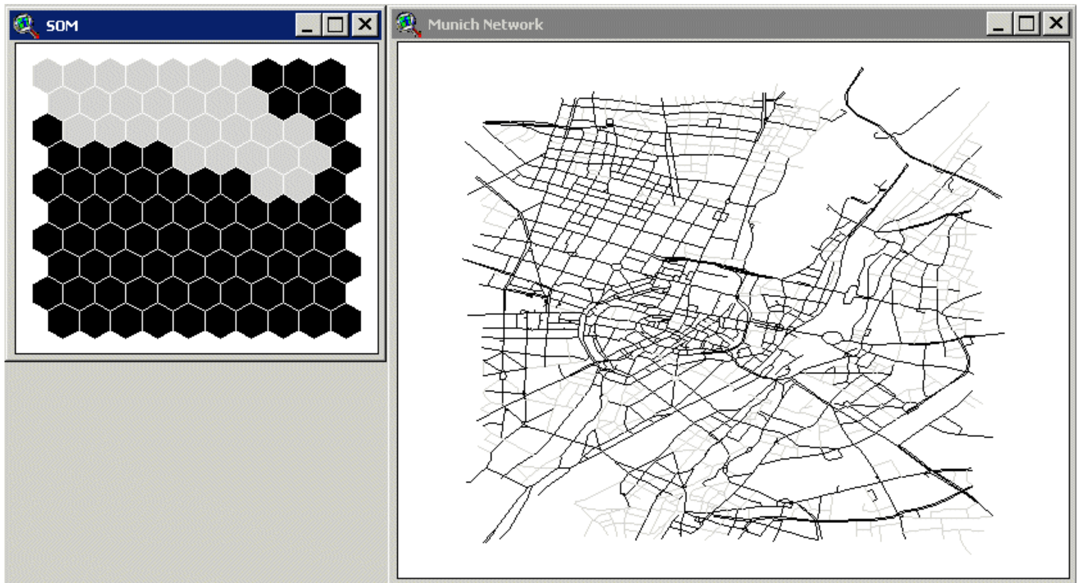
- Elimination of streets at different LOD



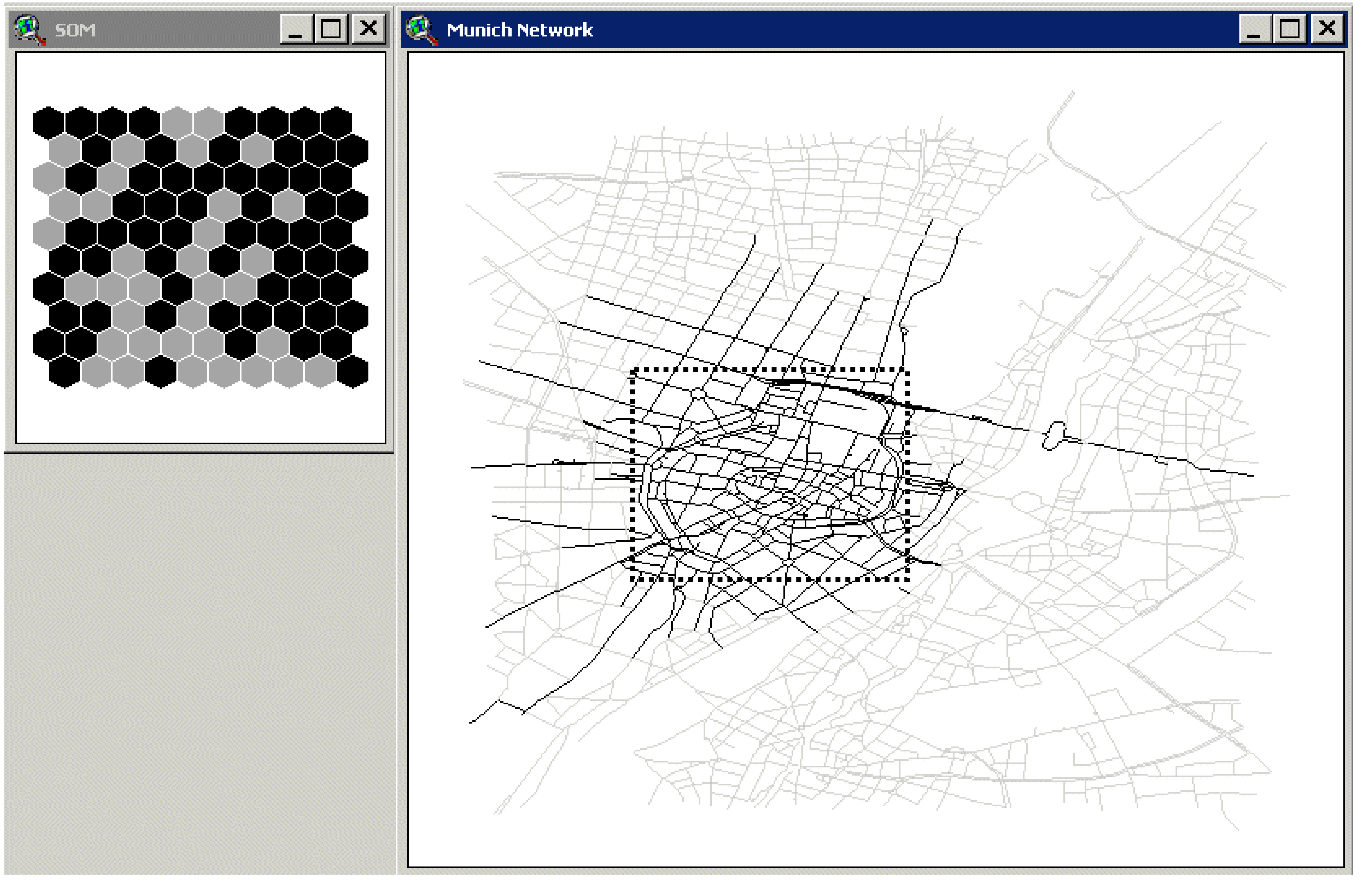


- Selection of streets at different LOD





- From network to the SOM



- Conclusion and future work
  - Our case study applied to Munich network approves that the SOM-based approach can be used as an effective way for selection of streets.
  - It also approves that it is an effective tool for data visualization and exploration for multi-dimensional data.
  - SOM training process is rather sensible process in terms of parameter settings that depend much on the properties of SOM and the training data. This deserves further research.
  - Although SOM is applied to the selection of streets, it could be applied to the selection or generalization of other spatial objects, as long as such selection is governed by multiple attributes.