*Transactions in GIS*, 2004, 8(3): 335–350

Research Article

# Selection of Streets from a Network Using Self-Organizing Maps

Bin Jiang
*Division of Geomatics*
*Department of Technology and Built Environment*
*University of Gävle*

Lars Harrie
*GIS Centre*
*Lund University*

**Abstract**
We propose a novel approach to selection of important streets from a network, based on the technique of a self-organizing map (SOM), an artificial neural network algorithm for data clustering and visualization. Using the SOM training process, the approach derives a set of neurons by considering multiple attributes including topological, geometric and semantic properties of streets. The set of neurons constitutes a SOM, with which each neuron corresponds to a set of streets with similar properties. Our approach creates an exploratory linkage between the SOM and a street network, thus providing a visual tool to cluster streets interactively. The approach is validated with a case study applied to the street network in Munich, Germany.

## 1 Introduction

Selection of streets from a street network is an important generalization operation, and may be a prerequisite to other generalization operations such as line simplification. A common technique in map production is to select the streets based on semantic attributes (mainly street types or function classes). Several researchers have pointed out that this approach is insufficient and have claimed that topological and geometrical properties of the streets cannot be neglected. Of note is the work of Mackaness and his colleague (Mackaness and Beard 1993, Mackaness 1995) who recognized the potential of graph theory in the application of street network generalization. In their work the streets were selected by their connectivity properties. Thomson and Richardson (1995) used a graph-theoretic approach based on the concept of minimum spanning trees to select important street segments within a street network. From a functional point of

**Address for correspondence:** Bin Jiang, Division of Geomatics, Department of Technology and Built Environment, University of Gävle, SE-801 76 Gävle, Sweden. E-mail: bin.jiang@hig.se

view, Morisset and Ruas (1997) model the importance of streets by the amount of 'street use'. Thus they proposed an approach of selecting characteristic streets based on high frequency of street usage through an agent-based simulation. Thomson and Richardson (1999) used a 'good continuation' principle of perceptual organization to the generalization of networks, and of street networks in particular. The principle serves as the basis for partitioning a street network into a set of linear elements (or strokes), which are chains of network arcs. In terms of structural measures such as length, class and connectivity, relative important 'strokes' can be derived for generalization purpose. All the above investigations are actually oriented towards the selection of important street segments rather than an entire named street, because street networks are modelled at a geometric level. Recently, Jiang and Claramunt (2002) proposed a set of algorithms based on a structural analysis. The structural approach is performed at a topological level with a representation, which takes named streets as nodes and street intersections as links of a connectivity graph. Based on the graph-theoretic representation, each street is assigned by some structural measures on which the selection of important streets is based. Eventually the selection of streets is applied to an entire named street rather than a street segment using these structural measures.

In this paper, we propose an approach that considers street attributes involving topological, geometric and semantic properties as input to a self-organizing map (SOM) (Kohonen 2001). SOM is an artificial neural network algorithm that, in our study, is used to categorise the streets in a network. More specifically, our approach adopts the SOM training algorithm to group all streets into different categories according to various attributes, and then selects streets at reduced map scales based on these categories. As the process is based on multiple attributes from topological, geometric and semantic perspectives, this approach can be considered to be comprehensive in terms of the process. Through the trained SOM – a two dimensional grid of neurons, the similarity of streets can be interactively explored and visualized. It should be noted that the approach we propose here is a general one for categorizing objects based on multiple attributes. Therefore, it could be used for other types of spatial object selection where multiple attributes should be considered.

SOM has been used in many fields such as data classification, pattern recognition, image analysis, and exploratory data analysis (for an overview, see Oja and Kaski 1999). In the domain of GIS and cartography, relatively few applications have been made. However, Openshaw and his colleagues have used the SOM approach in spatial data analysis to classify census data (Openshaw 1994, Openshaw et al. 1995). Recently, some new proposals have been offered for using SOM to explore spatial data (Li 1998) and in image classification (Luo and Tseng 2000). SOM has been used for building typification in cartographic generalization (Højholt 1995, Sester 2001). In the typification process, a number of building objects were set to represent a larger set of objects. A major issue here is that the new objects should reflect the original pattern of objects. In the approach introduced by Højholt and Sester, new building objects are placed randomly on the map. Then the locations of the new building objects are changed using SOM. In this training process the original building objects are used for attracting the new building objects. In this way the location of the new building objects will give a similar pattern as the location of the original building objects (but properties such as parallelism are not maintained). The use of SOM in this paper is rather different. Here it is used for attribute clustering as a pre-process for selection of streets; the locations of the streets are not altered.

The remainder of this paper is structured as follows. Section 2 presents the basic principle and algorithm of SOM. Section 3 introduces multiple attributes respectively from topological, geometric and semantic perspectives for individual streets within a network. Based on these multiple attributes, section 4 introduces a SOM-based approach for the selection of streets. Section 5 illustrates an application of our approach using the Munich street network as an example. Finally, section 6 concludes the paper and offers several ideas for future work.

## 2 Self-organizing Map

SOM is a well-developed neural network technique for data clustering and visualization. It can be used for projecting a large data set of a high dimension into a low dimension (usually one or two dimensions) while retaining the initial pattern of data samples. That is, data samples that are close to each other in the input space are also close to each other in the low dimensional space. In this sense, SOM resembles a geographic map showing the distribution of phenomena, in particular referring to the first law of geography: everything is related to everything else, but near things are more related to each other (Tobler 1970). Herewith we provide a brief introduction to the SOM; readers are encouraged to refer to a more complete description (e.g. Kohonen 2001).

### *2.1 Basic principle*

Let's represent a d-dimensional dataset as a set of input vectors of $d$ dimensions, i.e. $X = \{x_1, x_2, \ldots, x_n\}$, where $n$ is the number of input vectors. The SOM training algorithm involves essentially two processes, namely vector quantization and vector projection (Vesanto 1999). Vector quantization creates a representative set of vectors, so called output vectors from the input vectors. Let's denote the output vectors as $M = \{m_1, m_2, \ldots, m_k\}$ with the same dimension as input vectors. In general, vector quantization reduces the number of vectors, and this can be considered as a clustering process. The other process, vector projection, aims at projecting the $k$ output vectors (in $d$-dimensional space) onto a regular tessellation (i.e. a SOM) of a lower dimension, where the regular tessellation consists of $k$ neurons. The projection is performed as such: "close" output vectors in $d$-dimensional space will be projected onto neighbouring neurons in the SOM. This will ensure that the initial pattern of the input data is kept.

The two tasks are illustrated in Plate 3, where both input and output vectors are represented as a table format with columns as dimension and rows as ID of vectors. Usually the number of input vectors is greater than that of the output vectors, i.e. $n > k$, and the size of SOM is the same as that of the output vectors without exception. In Plate 3, the SOM is represented by a transitional color scale, which implies that similar neurons occur together. It should be emphasized that for the purpose of explanation, we separate it into two tasks, which are actually combined together in SOM without being sequential.

### *2.2 The algorithm*

The above two steps, vector quantization and vector projection, constitute the basis of the SOM algorithm. Vector quantization is performed as follows. First the output vectors are initialized randomly or linearly by some values for their variables. Then in the following training step, one sample vector $x$ from the input vectors is randomly chosen and the distance between it and all the output vectors is calculated. The output vector that is closest to the input vector $x$ is called the Best-Matching Unit (BMU), denoted by $m_c$:

$$\| x - m_c \| = \min_i \{\| x - m_i \|\}, \qquad [1]$$

where $\|.\|$ is the distance measure. Second the BMU or winning neuron and other output vectors in its neighbourhood are updated to be closer to $x$ in the input vector space. The update rule for the output vector $i$ is:

$$m_i(t+1) = m_i(t) + \alpha(t)h_{ci}(t)[x(t) - m_i(t)] \qquad \text{for } i \in N_c(t) \qquad [2]$$
$$m_i(t+1) = m_i(t) \qquad \text{for } i \notin N_c(t)$$

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

The algorithm can be described in a step-by-step fashion as follows.

**Step 1: Define the input vectors that define an attribute space.** The input vectors are likely to be in a table format as shown in Plate 3, where $d$ variables determine a $d$-dimensional attribute space. Based on the input vector space, an initialized SOM will be imposed for the training process (cf. step 3).

**Step 2: Define the size, dimensionality, and shape of a SOM.** The size is actually the number of neurons for a SOM. It can be determined arbitrarily, but one principle is that the size should be sufficient to detect the pattern or structure of SOM (Wilppu 1997). The number of neurons can be arranged in a 1- or 2-dimensional space (dimensionality). Usually three kinds of shape are allowed, i.e. sheet, cylinder or toroid, but sheet is the default shape.

**Step 3: Initialize output vectors *M* randomly or linearly.** At the initialisation step, each neuron is assigned randomly or linearly by some values for the $d$ variables. Thus an initial SOM is imposed in the input vector space for the following training process.

**Step 4: Define the parameters that control the training process involving map lattice, neighbourhood, and training rate functions.** The number of neurons defined can be arranged in two different map lattices, namely hexagonal and rectangular lattices. However, the hexagonal lattice is usually preferred because of its better visual effect (Kohonen 2001). The Gaussian neighbourhood function is often adopted and is defined by:

$$h_{ci}(t) = e^{-d_{ci}^2/2\sigma_t^2} \qquad [3]$$

where $\sigma_t$ is the neighbourhood radius at time $t$, $d_{ci}$ is the distance between neurons $c$ and $i$ on the SOM grid. Note that 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 (see Figure 1). The training rate function can be linear, exponential or inversely proportional to time $t$ (see Vesanto et al. 2000, p. 10). For instance, $\alpha(t) = \alpha_0/(1 + 100t/T)$ is

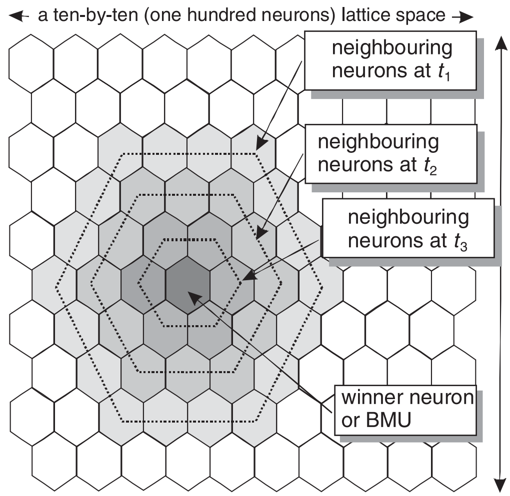


**Figure 1** The characteristics of a 10 × 10 SOM ($t_1 < t_2 < t_3$ with $h_{ci}(t)$ in equation (3))

the function we adopted in the following case study, where $T$ is the training length and $\alpha_0$ is the initial learning rate. Usually the training length is divided into two periods: $t_1$ for the initial coarse structuring period and $t_2$ for the fine structuring period.

**Step 5: Select one input vector *x*, and determine its Best-Matching Unit (BMU) or winning neuron using Equation (1).** Although Euclidian distance is usually used in Equation (1), it could be various other measures defining 'nearness' and 'similarity'. Depending on the form of data measurement, other measures are allowed as long as they represent the distance between input and output vectors.

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

**Step 7: Repeat steps 5 to 6 a very large number of times (training length) until a convergence is reached.** The convergence is set like this, $m_i(t + 1) = m_i(t)$, *for* $t \to \infty$. In practice, the training length in epochs is determined by the size of SOM ($k$) and the size of the training data ($n$), for instance for the coarse period $t_1 = \frac{4 \times k}{n}$.

Following the above steps, all output vectors are projected on to a 1- or 2-dimensional space, where each neuron corresponds to an output vector that is the representative of some input vectors. A 2-dimensional hexagonal map lattice grid is shown in Figure 1 where each hexagonal cell has a uniform neighbourhood.

## 3 Topological, geometric and semantic properties of streets within a network

It is important to note a distinction between a named street and a street segment. A named street refers to an entire street apparently identified by a unique name or ID, and usually consists of multiple street segments, while a street segment is just part of a named street (streets and named streets will be used interchangeably hereafter in the paper). In earlier studies, such as Thomson and Richardson (1995) and Morgenstern and Schurer (1999), the generalisation of the street network is based on street segments (in our terminology), but the study presented here is based on the entire named street.

For a street network, each street has different roles and there are numerous attributes that govern the roles of individual streets from the topological, geometric, and semantic perspectives. Before presenting these attributes, we shall introduce some basics on graph theory and a graph-theoretic representation of street networks.

### *3.1 Graph and a graph-theoretic representation of street networks*

A *graph* $G$ consists of a finite set of nodes (or vertices) $V$ and a finite set of links (or edges) $E$. A graph is often denoted as $G(V, E)$ where $V$ is the set of nodes, $V = \{v_1, v_2, \ldots, v_n\}$, and $E$ is the set of links, $E = \{v_i v_j\}$. For computational purposes we represent a connected, undirected and unweighted (i.e. all links with a unit distance) graph by an adjacency matrix $R(G)$: $R(G) = \lfloor r_{ij} \rfloor_{n \times n}$

where

$$r_{ij} = \begin{cases} 1 & if\ v_i v_j \in E \\ 0 & otherwise \end{cases} \quad [4]$$

It should be noted that this adjacency matrix $R(G)$ is symmetric, i.e. $\forall r_{ij} \Rightarrow r_{ij} = r_{ji}$. Also all diagonal elements of $R(G)$ are equal to zero so are not needed. Thus the lower or upper triangular matrix of $R(G)$ is sufficient for a complete description of the graph $G$. For a more complete introduction to graph theory, readers can refer for example to Gross and Yellen (1999).

A graph-theoretic representation of street works, which take named streets as nodes and street intersections as links of a connectivity graph, illustrates many nice properties for the understanding of the topology of street networks (Jiang and Claramunt 2004). For instance, Figure 2 shows a street network of a small urban area, and its corresponding

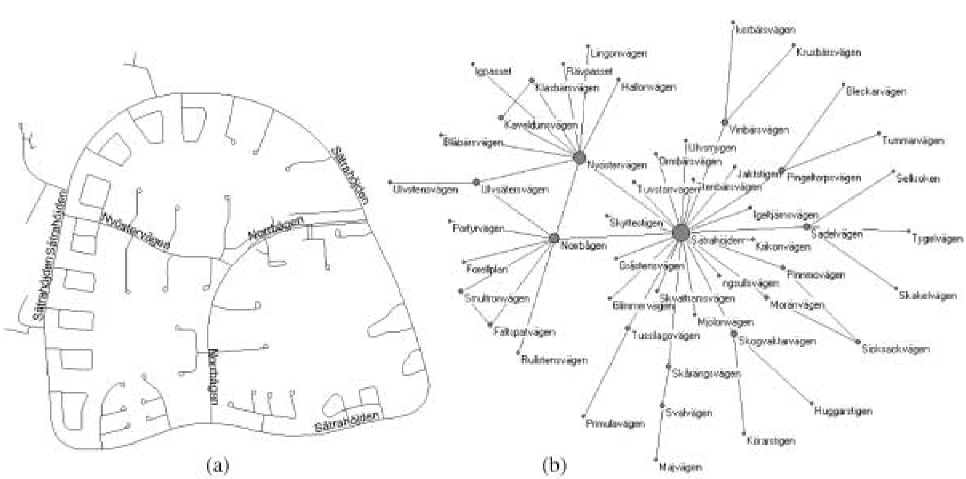


**Figure 2** A small street network (a) and its connectivity graph (b). Note that every node in (b) is labeled by the corresponding street name, and the size of nodes shows the degree of connectivity of individual streets

connectivity graph. In this connectivity graph the streets are modelled as nodes and the junctions as links. From this representation, we observe that this network is relatively closed: a bell-shaped street (Sätrahöjden) constitutes a form of boundary, and it is internally connected by two streets (Norrbågen and Nyöstervägen) that form an internal communication link. These three main streets form the main structure of this network to which other short streets are connected.

### *3.2 Topological properties*

The topological properties of a street network are the properties that are invariant in a homeomorphic (rubbersheet) transformation. Examples of such properties are intersection and connectedness. We assume that there are no prohibited turns or driving directions involved in the street networks, so the connectivity graph introduced above is an undirected graph, and the corresponding adjacency matrix is symmetrical.

Distinct from conventional geometric views of street networks, the connectivity graph takes a topological view of the network. It should be noted that geometric properties, e.g. Euclidean distances, are not stored in a connectivity graph. Concepts such as shortest path and shortest distance are here purely graph-based and should not be confused with their counterparts in a geometric network. In a connectivity graph, the shortest distance is defined as the minimum number of links between two nodes. By using the connectivity graph representation, many topological properties regarding the status of individual streets can be measured. Among many other measures initially developed from Social Network Analysis (SNA) (Scott 2000), centrality is one of important structural properties and it involves three important measures: *degree*, *closeness* and *betweenness* (Freeman 1979).

Degree centrality, also called connectivity, measures the number of streets that interconnect a given street. In a corresponding connectivity graph, degree is the number of nodes that link a given node. Formally, the degree centrality for a given node $v_i$ is defined by:

$$C_D(v_i) = \sum_{k=1}^{n} r(v_i, v_k) \qquad [5]$$

Closeness centrality measures the smallest number of links from a street to all other streets. In a corresponding connectivity graph, it is the shortest distance from a given node to all other nodes within a network. It is defined by:

$$C_C(v_i) = \frac{n-1}{\sum_{k=1}^{n} d(v_i, v_k)} \qquad [6]$$

where $d(v_i, v_k)$ is the shortest distance between nodes $v_i$ and $v_k$.

Betweenness centrality measures to what extent a street is between streets. In a corresponding connectivity graph, it reflects the intermediary location of a node along indirect relationships linking other nodes. Formally it is defined by:

$$C_B(v_i) = \sum_{j=1}^{n} \sum_{k=1}^{j-1} \frac{p_{ikj}}{p_{ij}} \qquad [7]$$

where $p_{ij}$ is the number of shortest paths from $i$ to $j$, and $p_{ikj}$ is the number of shortest paths from $i$ to $j$ that pass through $k$, so $\frac{p_{ikj}}{p_{ij}}$ is the proportion of shortest paths from $i$ to $j$ that pass through $k$.

These three centrality measures describe the street status from a topological perspective. We can remark that degree shows topological relationship between a street and its immediate neighbouring street(s), while closeness shows the structural relationship between a street and all other streets. Both can be characterized as a local measure and a global measure respectively. A street with high connectivity does not guarantee that it will be well connected to all other streets. In addition, a street with few direct connections does not mean that it is less important, since it can play a 'bridge' role, which means that without it a network may be broken into two pieces. This property is controlled by betweenness.

### *3.3 Geometric and semantic properties*

Common sense tells us that long and wide streets tend to be more important. Thus, length and width are two most important geometrical properties. In practice length is often used for selection of a street in a network. For example, in the database specifications from Lantmäteriet (the Swedish national mapping agency) it is stated that *dirt roads* should be represented in the target data set if they:

> "lead to settlements (dirt roads between 100–250 m are represented as ramps) or other cartographic objects (minimum length 250 m), connect roads, are along shores or have a length of more than 500 m" (Lantmäteriet 1997, p. 142; authors' translation).

It is apparent that this rule is, apart from the length of the road, also based on connectivity properties.

Semantic attributes such as function classes (highway, motorway, and normal street) and speed limit; 90 km, 70 km and 50 km are important attributes to consider in the course of selection. Examples of other interesting properties are semantic properties that can be derived from linguistics and ontology. There are different names in English for a street such as road, path, avenue, boulevard, and square. These names usually imply different levels in terms of importance and function class, e.g. a boulevard is more important than a path. Based on such a semantic analysis, each street can be quantized with a number to show different levels, and it can then be considered together with topological and geometric properties as well. However since such a semantic property is language dependent, we leave it out of our current study.

## 4 A SOM-based selection approach considering multiple properties

Cartographic generalization involves two types of processes: model-based (the initial choices of the relevant information to be presented on the map) and geometric-based (the simplification of graphic characteristics of objects) (Weibel 1995). Model generalization takes place prior to geometric generalization. For example, generalising a street network can be divided into the following processes: (1) select the street objects to be presented on the map (model-based generalisation); and (2) simplify, smooth and displace

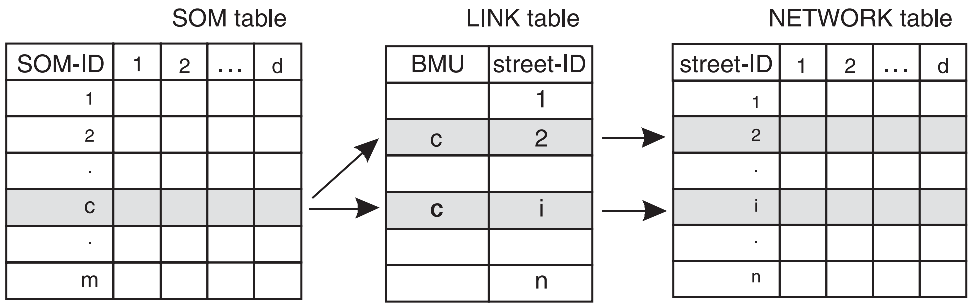


**Figure 3** Linkage between SOM and a street network

the streets to make the map readable (geometric-based generalisation). In this study we just consider model-based generalisation of a street network.

The process of our selection approach is as follows:

1. Each named street constitutes a vector in attribute space. This space is spanned by attributes of the types that were described in section 3.
2. These vectors then are used to train a SOM as described in section 2. In this SOM, neurons correspond to similar streets in terms of attributes introduced.
3. Selection is based on the two basic categories identified by the SOM.

From a more practical perspective, the streets and their corresponding attributes are used to create input vectors. The training process is performed using the SOM Toolbox associated with Matlab (Vesanto et al. 2000). Although the number of output vectors (or neurons) of a SOM can be arbitrarily determined, we usually choose a number that is smaller than the number of input vectors. Through the training process, each street acquires a best matching unit from the set of neurons within the SOM. It helps to set up a linkage between a SOM and corresponding street network. The specific procedure for setting up such a linkage using the ArcView GIS platform is as follows:

1. Create a polygon theme in which each polygon has a hexagonal shape, representing a neuron with output vectors as attributes in a table (SOM table).
2. Create a link table (LINK table) with two fields, namely BMU and street ID.
3. Link the SOM table and LINK table (note fields SOM-ID and BMU are equivalent).
4. Link the LINK table and NETWORK table through common field street-ID.

Through the above procedure, a linkage is set up between a SOM and a corresponding street network (Figure 3).

One principle of our approach is that all streets should be interconnected, which is partially controlled by the betweenness measure. If the original network is an integrated one, it should be kept as one in the reduced map scale and never broken into pieces. To meet this requirement, we must select some less important streets to join separated street clusters. This was achieved interactively.

## 5 A Case Study

To illustrate the selection approach, a case study was carried out using the street network of Munich, which involves a total of 785 streets. We used seven attributes namely

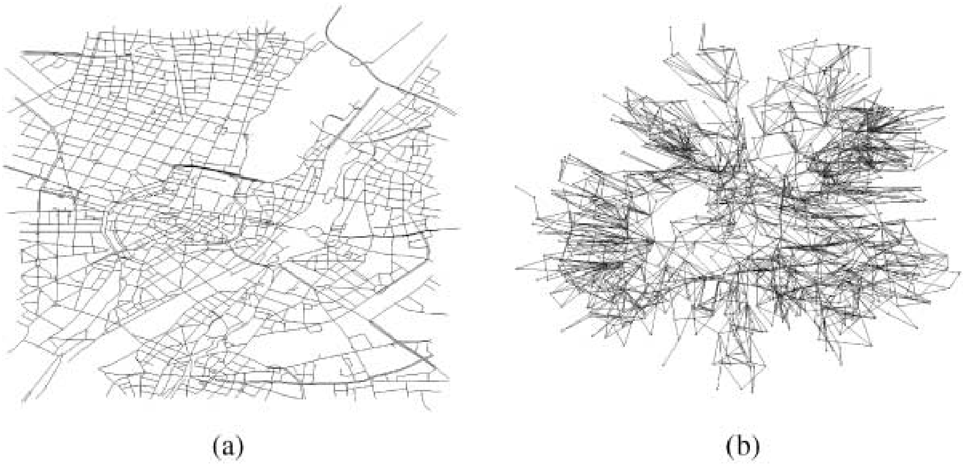


**Figure 4** The Munich street network (a) and its connectivity graph (b)

degree, closeness, betweenness, length, lanes (a modified version of width), speed limit and function class as described in section 3. These attributes nicely describe topological, geometric and semantic properties of individual streets within the network. They were used to define input vectors for the following training process.

### *5.1 Input vectors*

Our first step was to derive a connectivity graph based on the network. The structural representation provides a sense as to how each street interconnects to others. The process, with reference to Equation (4), was performed with an Avenue script with ArcView GIS, i.e. for each street ($i$), check if it intersects every other street ($j$), if yes, $r_{ij} = 1$, otherwise $r_{ij} = 0$. Thus, a connectivity graph can be derived to represent such interconnections. Figure 4 illustrates the street network and its connectivity graph. Based on the connectivity graph, the topological properties can be calculated; it is performed using a network analysis software package called Pajek (Batagelj and Mrvar 1997). The remaining properties are directly derived from a GIS database with the network. Both length and lanes were transformed from street segments to individual streets by taking an average value of each attribute assigned to the street segments. It is important to note that the attributes class and speed are in an ordinal scale, e.g. 1 for the street with the highest speed limit and 8 for the lowest. Furthermore, function class and speed limit are inversely proportional to the importance of a street, i.e. the higher the value, the less important the street. In this case, we take a reciprocal value of these attribute values for the training process. Thus all seven attributes have the same order in terms of the importance of the streets. The seven attributes constitute an attribute space in which each input vector has a unique location. Table 1 shows part of the input vectors of the multi-dimensional dataset containing 785 records.

### *5.2 Initialization and SOM training process*

With the input vectors we are ready to do the second step – namely initialisation and training. To this end, we first define a map size of 100, i.e. a 10 × 10 SOM in a two-dimensional grid. Then using linear initialization we create output vectors, each of

**Table 1** The first 10 input streets 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 |

**Table 2** Parameter settings for the SOM training

| Parameter | Value |
|---|---|
| Size (m) | 100 |
| Dimensionality | 2 |
| Shape | Sheet |
| Map lattice | Hexagonal |
| Neighbourhood | Gaussian |
| Learning rate ($\alpha$) | $\alpha(t) = \alpha_0/(1 + 100t/T)$ |
| Initial learning rate ($\alpha_0$) | 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 ($\sigma_0$) | 5 |
| Final neighbourhood radius | 1.25 for the coarse period |
| | 1 for the fine period |

which has seven attributes. The initialised output vectors were trained based on the input vectors. The variation of individual attributes noted from Table 1 is very large, so we determined to transform the dataset into a unit interval [0, 1] to guarantee that all variables have the same variation. From cartographic generalization practise, we determined that function class has the highest priority in street selection, and this is followed by geometrical and topological properties. Therefore, we adopt the weight vector [1, 1, 1, 2, 2, 2, 3] for the seven attributes in the order: [degree, closeness, betweenness, length, lanes, speed, class]. Users can adopt a different weight vector according to their different generalization purposes. During the training process, a Gaussian neighbourhood function was chosen. Other detailed parameter settings are listed in Table 2.

It is important to note that the size of SOM has a significant impact on detecting clusters (Wilppu 1997), but the size we chose seemed sufficient to detect the pattern. Table 3 illustrates the first 10 neurons of the trained map of the $10 \times 10$ SOM; each row of the table corresponds to a neuron of the SOM. The 10 neurons correspond to the

**Table 3** The 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 |

left most column of the SOM shown in Plate 3. It should be noted that following the training process, the vectors have been transformed back to initial data ranges. The rows in Tables 1 and 3 correspond to input vectors and output vectors respectively in Plate 3.

### *5.3 Visualization of the SOM*

The trained SOM is composed of neurons in a two dimensional space that preserve the initial pattern in the attribute space of the input streets (or input vectors in terms of SOM). In other words, the seven dimensional dataset is now mapped into a two dimensional SOM, retaining the initial pattern. Such a pattern can be seen clearly from visualizing the SOM in terms of individual attributes. It should be noted that each cell or neuron is not a particular street(s), but instead it represents a group of similar streets in terms of the seven attributes. Plate 4 illustrates component views of SOM from the perspective of the seven attributes. We can remark that the smooth colour transitions with the visualizations imply similar neurons being together, and some of attributes have significant correlation, e.g., degree and length.

The two dimensional SOM arranges similar neurons within a neighbourhood, but is does not convey how similar or dissimilar adjacent neurons are. Instead, a unified distance matrix (U-matrix) between the output vectors of adjacent units of a SOM illustrates the degree of similarity by calculating distances among them (Ultsch and Siemon 1990). Thus it helps to illustrate the cluster structure of the SOM. Plate 5(a) is the U-matrix representation, where six hexagons around each neuron show the similarity between neurons, i.e. the lighter the hexagon the greater the similarity. Plates 5(b), (c) and (d) shows various ways of visualizing the U-matrix using different visual variables such as colour intensity, size of cells and colour hues, noting that the D-matrix is a distance matrix from a neuron to its neighbouring neurons. Our aim is not to make a detailed clustering analysis, but rather to make a distinction between those that are selected or eliminated. From the set of visualizations, we note that there are two clusters as indicted by A and B, although the boundary of the two clusters is not so clear-cut, and similarity within cluster B is not as homogeneous as in cluster A. However the distinction between the two clusters might be more clearly reflected in the corresponding network as illustrated in the following subsection.

*5.4 Selection or elimination of streets at different levels of detail*

The above SOM is imported into the ArcView GIS as a polygon theme and each hexagon represents a neuron with an output vector to be stored in the theme table. Using the principle given in section 4, we set up a linkage between the SOM and the Munich network. The 785 streets have been grouped into 100 neurons, and each of them represents a cluster. The training process can be considered as a transformation of generalization and clustering. We have also seen the two clusters in terms of a distance matrix. In the corresponding network, we can see that cluster A represents a set of unimportant streets to be eliminated in the course of generalization. Figure 5 illustrates two levels

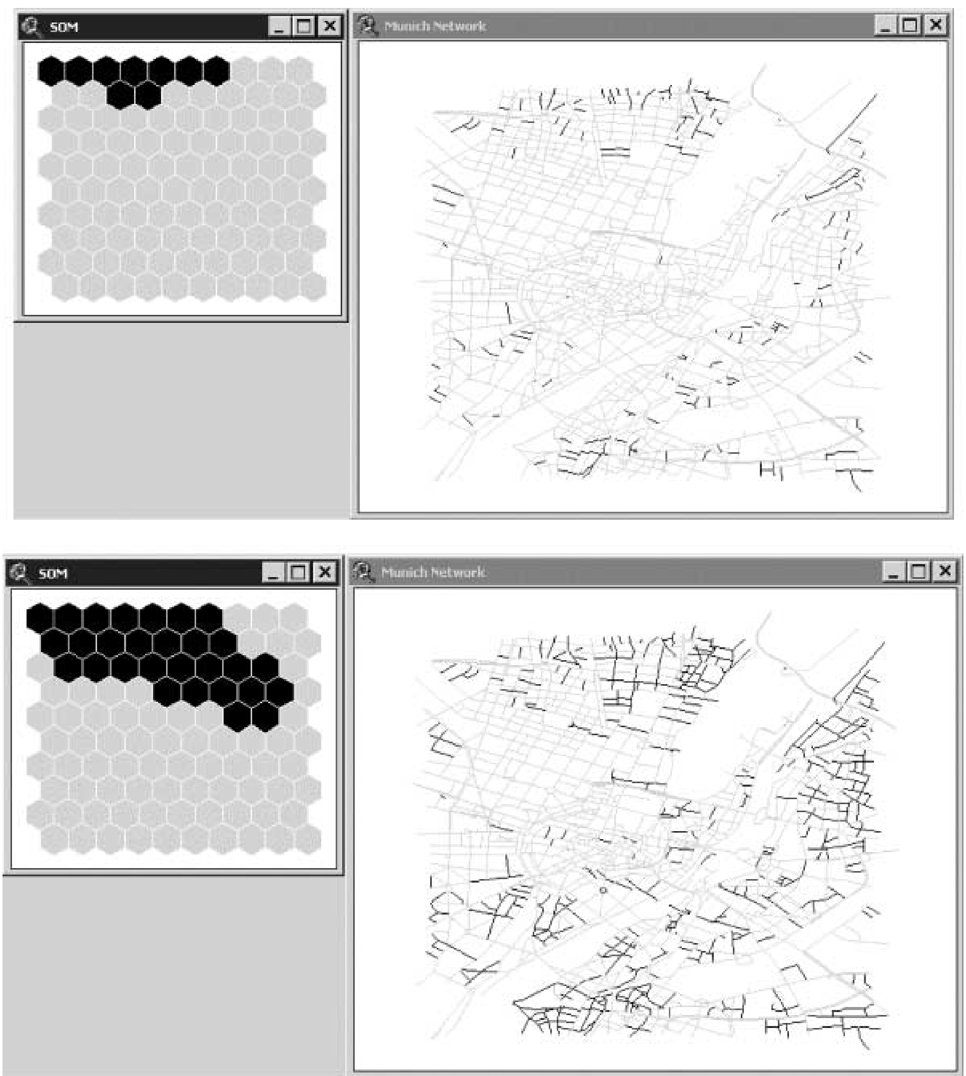


**Figure 5** Two levels of detail of streets to be eliminated

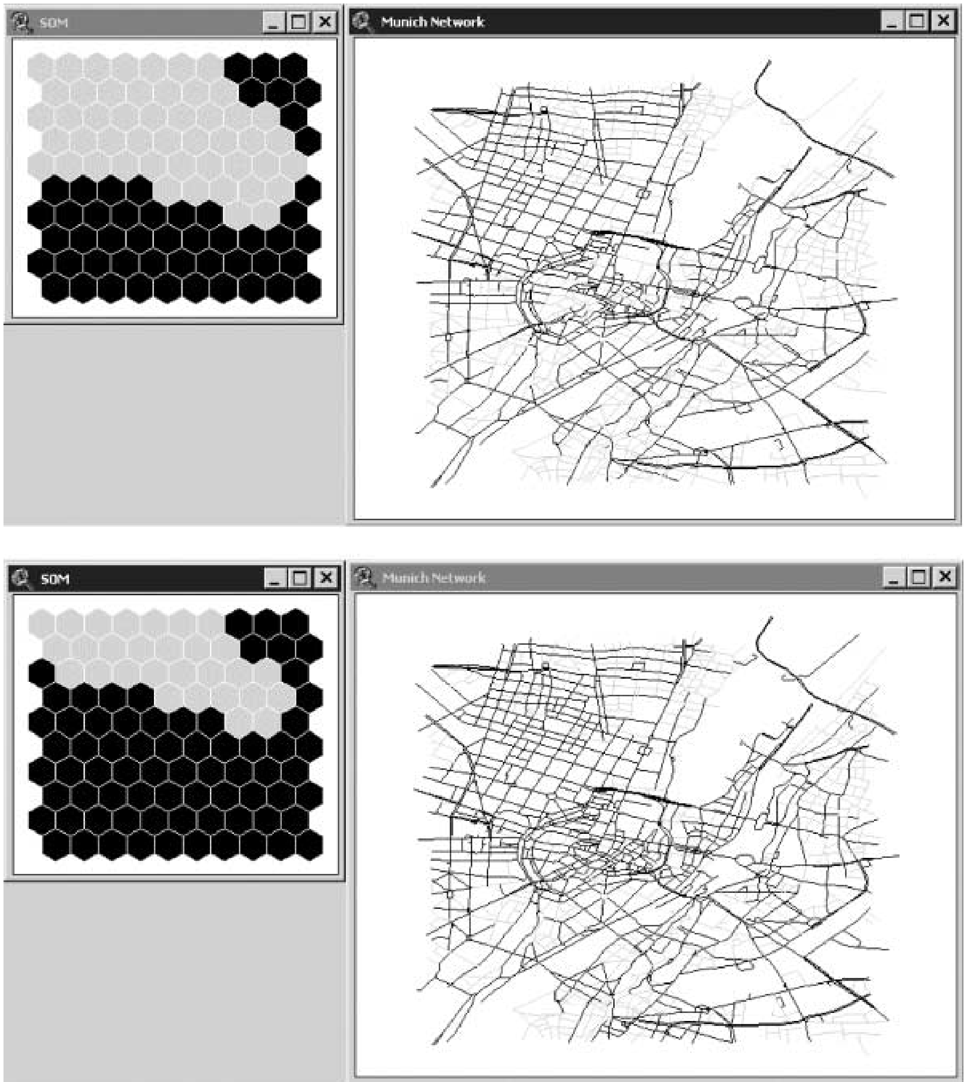


**Figure 6** Two levels of detail of streets to be selected

of detail for streets intended for elimination (dark lines in the network view and dark cells in the SOM view). In the similar way, Figure 6 illustrates two levels of detail for those streets to be retained. We note that the selected neurons in Figure 6 are those neurons with higher values for the various attributes shown in Plate 4.

These figures illustrate how the SOM-based approach can be used to select and eliminate streets from a network. The approach is robust and flexible, as it considers multiple attributes involving topological, geometric and semantic properties. Furthermore, a dynamic linkage has been set up between the SOM and the corresponding network for selection or elimination purposes.

## 6 Conclusions and Future Work

From the model generalization perspective, this paper adopts a street-centred view and considers multiple attributes from topological, geometric and semantic aspects for the selection of streets from a network. Our approach clusters streets in different categories according to the similarity distance in a high dimensional attribute space using the SOM training algorithm. Through a linkage between the SOM and the original street network, end users are able to select streets for model generalization purposes. The case study applied to the Munich network illustrates that the SOM-based approach can be used as an effective method for the selection of streets. It also shows that it is an effective tool for data visualization and exploration for multi-dimensional geospatial data.

It is important to note that for a given dataset and defined SOM properties, the SOM training process is dependent on the parameter settings. The settings we adopted are default settings (Table 2) recommended by the SOM toolbox. This issue deserves further research. Furthermore, the seven variables are certainly not exhaustive; other combinations and weightings might be more appropriate. 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. Our future work will seek to apply the approach to the generalization of other spatial objects.

## Acknowledgements

An earlier version of this paper was presented in Paris at the Fifth Workshop on Progress in Automated Map Generalization, 28–30 April, 2003. The authors thank the referees for their constructive comments and suggestions that significantly improved the quality of this paper. The Munich dataset was provided by NavTech from the year 2000.

## References

Batagelj V and Mrvar A 1997 Networks/Pajek: Program for Large Networks Analysis. WWW document, http://vlado.fmf.uni-lj.si/pub/networks/pajek/

Freeman L C 1979 Centrality in social networks: Conceptual clarification. *Social Networks* 1: 215–39

Gross J and Yellen J 1999 *Graph Theory and its Application*. London, CRC Press

Højholt P 1995 Generalization of Built-up Areas Using Kohonen-networks. In *Proceedings of Eurocarto XIII*, 2–4 October, Ispra, Italy

Jiang B and Claramunt C 2002 A Structural Approach to Model Generalisation of an Urban Street Network. In *Proceedings of the Fourth AGILE Conference on Geographic Information Science*, 24–27 April, Mallorca, Spain: 205–17 (a revised version of this paper has been published in *GeoInformatica* 8: 157–71)

Jiang B and Claramunt C 2004 Topological analysis of urban street networks. *Environment and Planning B* 31: 151–62

Kohonen T 2001 *Self-Organizing Maps* (Third Edition). Berlin, Springer

Lantmäteriet 1997 *Minimimått för T5 Version 2.1* (in Swedish). Gävle, Sweden.

Li B 1998 Exploring Spatial and Temporal Patterns with Self-Organizing Maps. In *Proceedings of GIS/LIS '98*. Bethesda, MD, American Society for Photogrammetry and Remote Sensing: 89–96

Luo J H and Tseng D C 2000 Self-Organizing Feature Map for Multi-Spectral Spot Land Cover Classification. WWW document, http://www.gisdevelopment.net/aars/acrs/2000/ts9/imgp0024.shtml

Mackaness W A 1995 Analysis of urban road networks to support cartographic generalization. *Cartography and Geographic Information Systems* 22: 306–16
Mackaness W A and Beard M K 1993 Use of graph theory to support map generalisation. *Cartography and Geographic Information Systems* 20: 210–21
Morgenstern D and Schurer D 1999 A Concept for Model Generalization of Digital Landscape Models from Finer to Coarse Resolution. In *Proceedings of ICC'99*, Ottawa, Canada: 46–53
Oja E and Kaski S (eds) 1999 *Kohonen Maps*. New York, Elsevie.
Openshaw S 1994 Neuroclassification of spatial data. In Hewitson B C and Crane R G (eds) *Neural Nets: Applications in Geography*. Dordrecht, Kluwer: 53–70
Openshaw S, Blake M, and Wymer C 1995 Using Neurocomputing Methods to Classify Britain's Residential Areas. WWW document, http://www.geog.leeds.ac.uk/papers/95-1/
Scott J 2000 *Social Network Analysis*. Thousand Oaks, CA, Sage Publications
Sester M 2001 Kohonen Feature Nets for Typification. In *Proceedings of the Fourth Workshop on Progress in Automated Map Generalization*, Beijing, China
Thomson R C and Richardson D E 1995 A Graph Theory Approach to Road Network Generalisation. In *Proceedings of the Seventeenth International Cartographic Conference:* 1871–80
Tobler W R 1970 A computer movie simulating urban growth in the Detroit region. *Economic Geography* 46: 234–40
Ultsch A and Siemon H P 1990 Kohonen's Self Organizing Feature Maps for Exploratory Data Analysis. In *Proceedings of the International Neural Network Conference (INNC'90)*. Dordrecht, Kluwer: 305–8
Vesanto J 1999 SOM-based data visualization methods: Intelligent data analysis. *Elsevier Science* 3: 111–26
Vesanto J, Himberg J, Alhoniemi E, and Parhankangas J 2000 *SOM Toolbox for Matlab 5*. Helsinki, Helsinki University of Technology Report No. A57
Weibel R 1995 Three essential building blocks for automated generalisation. In Muller J C, Lagrange J P, and Weibel R (eds) *GIS and Generalization: Methodology and Practice*. London, Taylor and Francis: 56–69
Wilppu E 1997 *The Visualization Capability of Self-organizing Maps to Detect Deviations in Distribution Control*. Turku, Turku School of Economics and Business Administration Technical Report No. 153

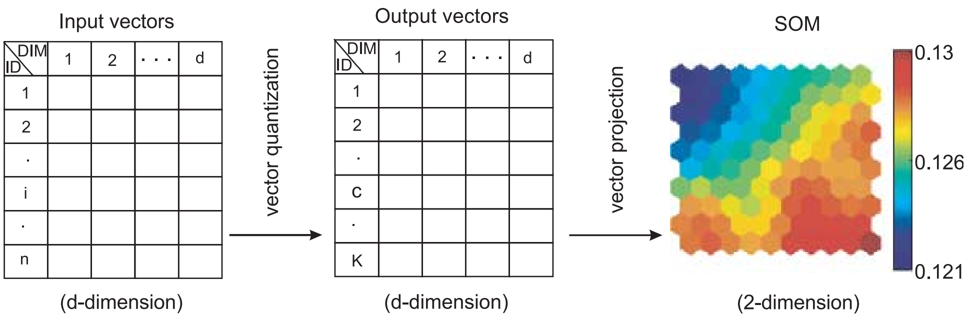


**Plate 3** Illustration of SOM principle

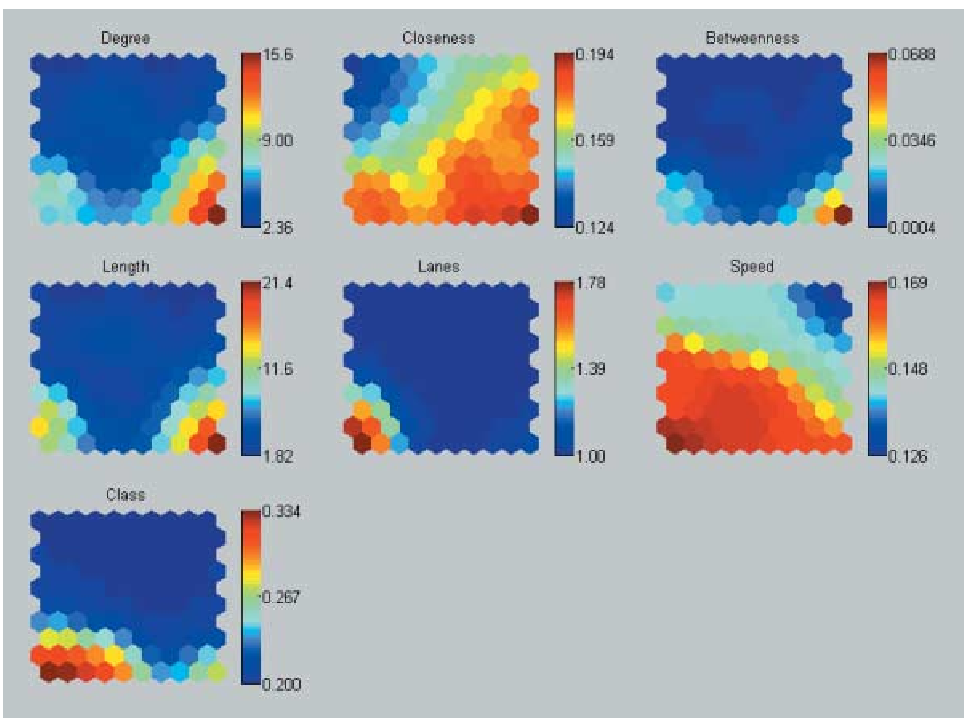


**Plate 4** Component visualizations of the SOM

Plates 3 and 4 from B Jiang and L Harrie. 'Selection of Streets from a Network Using Self-Organizing Maps', 335–350

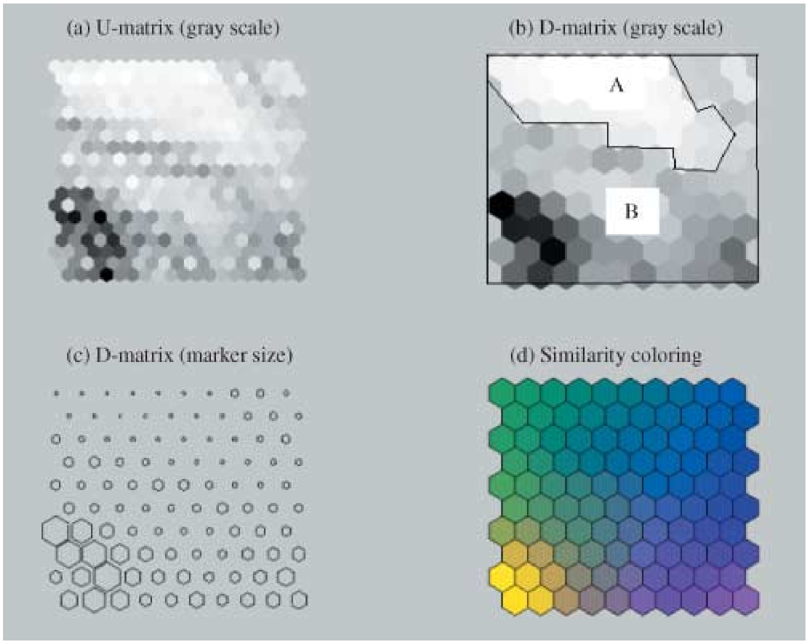


**Plate 5** U-matrix visualizations of the SOM

Plate 5 from B Jiang and L Harrie. 'Selection of Streets from a Network Using Self-Organizing Maps', 335–350