Component Selection for the Metro Visualisation of the Self-Organising Map

Robert Neumayer, Rudolf Mayer, and Andreas Rauber
Vienna University of Technology, Department of Software Technology and Interactive Systems
e-mail: {neumayer,mayer,rauber}@ifs.tuwien.ac.at

Keywords: Component Planes, Visualisation, Cluster Analysis

***Abstract*— *Self-Organising Maps* have been used for a wide range of clustering applications. They are well-suited for various visualisation techniques to offer better insight into the clusterings. A particularly feasible visualisation is the plotting of single components of a data set and their distribution across the SOM. One central problem of the visualisation of Component Planes is that a single plot is needed for each component, which leads to problems with higher-dimensional data. We therefore build on the Metro Visualisation for Self-Organising Maps which integrates Component Planes into one illustration. Higher-dimensional data sets still pose problems in terms of overloaded visualisations – component selection and aggregation techniques are highly desirable. Hence, we propose and compare two methods, one for the aggregation of correlated components, one for the selection of the components that are most feasible for visualisation with respect to a certain SOM clustering.

1 Introduction

The Self-Organising Map (SOM) is an unsupervised neural network used for the mapping of high-dimensional data onto a usually two-dimensional output space. Exploratory data analysis, for example, can be based on the Self-Organising Map principles to allow insight into data that usually cannot be given due to its high dimensionality.

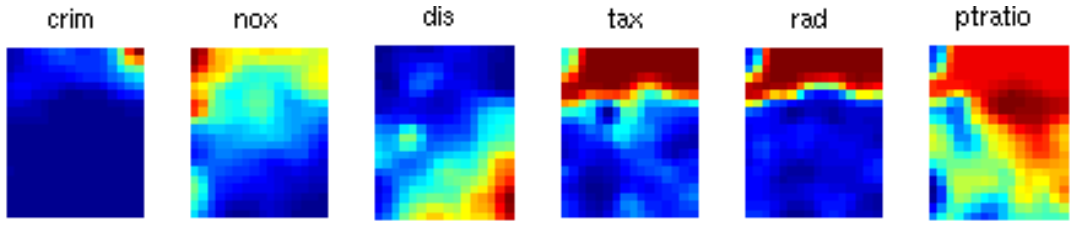
A fact heavily contributing to the popularity of the Self-Organising Map is the wide range of available visualisations. Even though the projection to two dimensions already brings a significant alleviation, the often large number of data vectors and their initially very high dimensionality still leads to difficulties in understanding the coherence within the data. Especially for inexperienced users, plain Self-Organising Map clusterings may seem overwhelmingly difficult.

A well-known method to gain a better understanding of the characteristics of certain areas of the map is the visualisation of *Component Planes*. This visualisation partitions the SOM into projections of single variables or components. However, they are often still hard to make sense of in case of high-dimensional data sets as the number of plots needed for displaying Component Planes is still equal to the number of dimensions. The clustering of component Component Planes in order to obtain groups of common characteristics can often ease this problem.

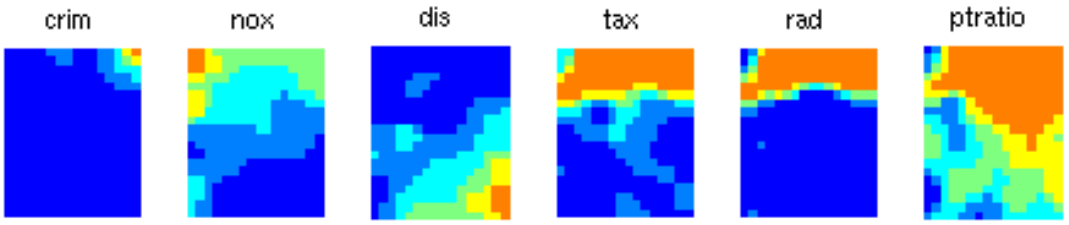
In this paper, we propose an intuitive metaphor of maps of metro lines, which aims at showing a simplified representation of the components in a single illustration – which is a huge simplification compared to the plots of Component Planes. For the Metro Visualisation each variable is represented by differently coloured and connected line segments, called *Component Lines*. The metaphor of metro maps utilises the concept of skewed distances. For these distances do not truthfully represent real-world distances, they are well-suited for our method.

In this paper, we particularly focus on the selection of feasible components for visualisation. More specifically, we emphasise the selection of components according to the SOM clustering and the way these variables were mapped onto the Self-Organising Map. This approach has the main advantage of applying feature selection with respect to the positions and transitions of a clustering's component regions. In other words, we omit components which would result in scattered line visualisations across the map and therefore hard to meaningfully display and apply the Metro Visualisation to the remaining features only. In addition, we propose an aggregation method which groups Component Lines in case they are highly correlated, reducing the amount of redundant information displayed. Many of the steps involve a trade-off between the level of detail and amount of information, and the clarity of the representation. We sometimes deliberately choose to sacrifice accuracy in order to communicate the data in an intuitive manner. The resulting visualisation allows to intuitively communicate relationships between multiple variables and tendencies on a SOM in a single visualisation, abstracting from spurious details and focussing on the dominant attribute value distributions on the SOM.

This paper is structured as follows. Section 2 gives an overview of related work. In Section 3 we describe the method for computing the Metro Visualisation. In Section 4 we elaborate on methods for the selection of most feasible components as well as their aggregation. In Section 5, we apply our approach to the Boston Housing data set. A conclusion and an outlook on future work are given in Section 6.



(a) Plot of single component planes



(b) Plot of component planes grouped into six regions

Figure 1: Selected component planes and grouped component planes for the Boston Housing data set

2 Related Work

In this section, we introduce the Self-Organising Map and related concepts and visualisation techniques.

2.1 Self-Organising Map

The Self-Organising Map is a widely-used, unsupervised neural network model [1]. Its basic operation is the mapping from a high-dimensional input space to a lower-dimensional, mostly two-dimensional, output space. The mapping is preserving the existing topology within the data, i.e. input patterns that are located closely in the input space will also be positioned close to each other in the output space. By contrast, dissimilar patterns will be mapped on opposite regions of the map.

The Self-Organising Map is a low-dimensional lattice, comprising M neurons or units. In this paper we use a two-dimensional lattice and rectangular maps as the topology. For each unit in the output space, a model vector $\mathbf{m}_i$ of the dimensionality of the input space is linked to a position on the map, denoted as $\xi_i = (\xi_i^x, \xi_i^y)$. The model $\mathfrak{M}$ is the set of all model vectors. As part of the training process, the best matching unit is identified by the use of a certain distance function and its model vector and the model vectors of neighbouring units are shifted towards the input vector. Self-Organising Maps have been applied to a multitude of tasks, ranging from data mining [11] to document organisation in digital libraries.

2.2 Self-Organising Map Visualisations

When Self-Organising Maps are to be visualised, one platform to be used is the map lattice itself [10]. In that case, quantitative information is most commonly displayed via colour values or markers of different sizes. The analogy to geography, for example, is exploited in [7]. Another possibility is the usage of an island-like metaphor, taking into account the data vectors itself, to visualise important regions of the map [4]. Many SOM visualisation techniques that rely solely on the model vectors, others take into account the distribution of the data samples.

The unified distance matrix (U-Matrix) [9], e.g., is a visualisation technique that shows the local cluster boundaries by depicting pair-wise distances of neighbouring prototype vectors. The Gradient Field [5] has some similarities with the U-Matrix, but applies smoothing over a broader neighbourhood. It plots a vector field on top of the lattice where each arrow points to its closest cluster centre. This can be used to contrast different groups of Component Planes [6], with a similar goal as the method we describe. Similarly, [11] applies clustering of and projection techniques on the Component Planes with the aim of visually ordering them. Other methods include Smoothed Data Histograms [4], which show the clustering structure by mapping each data sample to a number of map units. The P-Matrix [8] is a density based approach that depicts the number of samples that lie within a sphere of a certain radius around the model vectors. The radius is a quartile of the pair-wise distances of the data vectors. Other techniques adjust the distances in between units during the training process to separate the cluster boundaries more clearly [2].

2.3 The London Underground Map

Metro map visualisations were introduced in the 1930s for the London underground transportation network, and are, with only slight modifications, still used for today's London metro maps. Contrary to previous metro maps it disre-

garded geographical aspects, with the geometric representation of the river Thames being the only link between the map and the actual landform of the area it represented. Today, this kind of schematic representation has become very well-known. It is common knowledge that the distances on metro maps are skewed and do not conform with real-world distances, which is also true for the Metro Visualisations of Component Lines we describe in this paper. Its prime attraction, however, lies in its simplicity, abstracting from spurious details and resulting in a more abstract representation that is more easily memorised and compared across different variations.

3 The Metro Visualisation

The Metro Visualisation of Component Lines has been first presented in [3]. Here, we will give a brief summary of the most important aspects and then explain the component selection process on top of the existing basic visualisation approach in more detail.

3.1 From Components to the Metro Map

The classic Component Plane visualisation for some selected components is shown in Figure 1(a). Values for this component are depicted by colour-coding each unit. Starting from that representation, each component is split into a number n of disjoint ranges. In this paper, the division is performed by calculating the threshold values as equidistant points between the lowest and highest values in the particular Component Planes. This results in n partitions of the SOM. The upper limit l for region k for a component $\mathbf{c_j}$ is defined as follows:

$$l_k(\mathbf{c_j}) = \frac{k \cdot (\max \mathbf{c_j} - \min \mathbf{c_j})}{n} + \min \mathbf{c_j} \qquad (1)$$

where $\mathbf{c_j} \in R^M$ is the $j-th$ component, and $\max \mathbf{c_j}$ and $\min \mathbf{c_j}$ denote the maximum and minimum values for this particular component, respectively. The set of units that fall within these intervals are denoted as:

$$\Theta_{j,k} = \{\xi_i \mid m_{i,j} \in [l_{k-1}, l_k]\} \qquad (2)$$

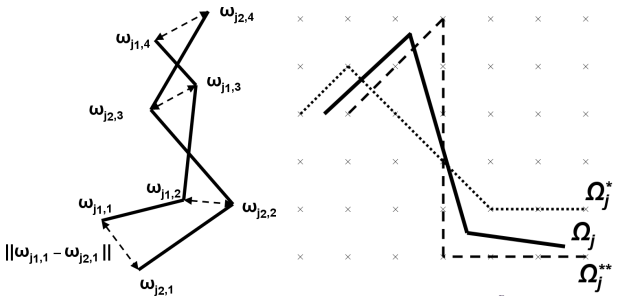
where j is an index over the dimensions or components, k an index over the number of regions. $\mathbf{m_i}$ refers to the model vectors.

Further, region centres $\omega_{j,k}$ for component j and region k are computed as the centres of gravity as follows:

$$\omega_{j,k} = \frac{1}{|\Theta_{j,k}|} \sum_{\xi_i \in \Theta_{j,k}} \xi_i \qquad (3)$$

For being continuous values the coordinates for ω do not necessarily coincide with the integer unit coordinates.

Ω_j denotes the entire tuple of centres $\{\omega_{j,k} \mid 1 \leq k \leq n\}$ and implicitly represents the $n-1$ lines, obtained by linking all centres of regions of a specific component ordered by their value, henceforth referred to as Component Lines.



(a) Measure of distances between lines

(b) Snapping of region centres to units of the SOM

Figure 2: Computation of distances between metro lines (a) and snapping of region centres (b)

3.2 Distances between Component Lines

Figure 1(b) shows the same components as Figure 1(a), but grouped into regions – the main modification that is made for the calculation of the Metro Visualisation. As opposed to the mere Component Planes, segregation of regions on the map is apparently much easier, which also paves the way for the identification of the most feasible components later on. In order to perform subsequent steps, we need to introduce a metric that measures the distance between two component lines Ω_{j_1} and Ω_{j_2}. This function d introduces a concept of dissimilarity, such that pairs of lines that are mutually more similar than others can be identified. We define this measure as

$$d(\Omega_{j_1}, \Omega_{j_2}) = \min \left(\sum_{k=1}^{n} \|\omega_{j_1,k} - \omega_{j_2,k}\|, \sum_{k=1}^{n} \|\omega_{j_1,k} - \omega_{j_2,(n+1-k)}\| \right) \qquad (4)$$

where $\| \cdot \|$ denotes the Euclidean norm. The idea behind this is that the lines are a simplified representation of the gradient of a single variable, which should be visually similar in case the variables are correlated. Thus, Component Lines which share approximately the same path are assigned a low distance. Figure 2(a) illustrates the computation of distances between Component Lines as the sum of the distances between the pairs of centre points of the same indices. Inverting the indices of Ω_{j2} as in the second argument of *min* in Equation 4 stems from the fact that Component Planes can be negatively correlated. For similarity only the absolute value of the correlation is of interest.

3.3 Visual Enhancements

Snapping For a more intuitive and smoother representation, and to more closely resemble the metaphor of a metro map, the locations of the region centres are adjusted so that the lines Ω_j are drawn only horizontally, vertically, or diagonally, i.e. in multiples of 45 degrees angles. In order to

achieve this kind of representation, we compute new Component Lines Ω_j^* where the centres $\omega_{j,k}$ are restricted to the discrete unit positions on the map. The selection of the most feasible Component Lines is performed by minimising the energy function

$$\min_{\Omega_j^*} d(\Omega_j, \Omega_j^*) \mid \Omega_j^* \in \mathfrak{S} \tag{5}$$

where $\mathfrak{S}$ is the set of valid candidate Component Lines as defined above. Figure 2(b) illustrates the process of aligning the original Ω_j to candidate lines Ω_j^* and Ω_j^{**}. The small crosses represent the discrete positions of the map units. In this example, the candidate with the smallest distance to the initial Component Line would be Ω_j^{**}.

Metro Stations and Intersections The centres ω, representing the centres of gravity of single components, are indicated by markers on the Component Lines, intuitively mimicking metro stops. To even more emphasise the metaphor of real-life metro maps, intersections of Component Lines are displayed as metro stations (white circles). These stations more clearly point out the meaning of parallel lines, namely their homogeneity with respect to a certain local similarity.

Iconified Cluster Boundaries As described in Section 2.2, the U-Matrix can be utilised to visualise cluster boundaries. We use this technique in our map to show distinct boundaries between clusters as iconified *rivers* or *lakes*, which often feature in the background of real-world metro maps. Analogously to a city being divided into different areas, our visualisation divides data into clusters.

4 Aggregation and Selection Steps

In this section, we explain two techniques for dimensionality reduction for the Metro Visualisation. One selects the most feasible components with respect to the given clustering, whereas the other combines different components into one. With an increasing number of dimensions in the input space, and therefore an increasing number of Component Plane visualisations, the perception of this visualisation becomes increasingly difficult. Component Lines can combine the Component Plane information in one plot, but, again, this approach is only feasible up to a certain dimensionality.

4.1 Aggregation of Component Lines

We propose an optional step of aggregating similar Component Lines into representative prototypes. This is based on clustering the Component Lines. With the distance measure between two such lines defined in Equation 4, a matrix of pair-wise distances can be calculated. Subsequently, hierarchical clustering with any of the common linkage metrics can be performed. In our approach, we use Ward's clustering, and the resulting model vectors are computed by averaging over the Component Lines within each cluster. A threshold value for the Ward's clustering can be used to influence the level of aggregation and to suit user's subjective information needs or desired levels of aggregation. This aggregation step can either be performed on the full set of components, or after any other kind of pre-selection process.

Algorithm 1 Region detection for Component Planes after the discretisation step

```
for each component c_j do
  for each component region ω_{j,k} do
    for each unit ξ_i do
      if ¬assigned_to_region(ξ_i) then
        #of_assigned_neighbours = 0
        for ξ_l in neighbours(ξ_i) {ξ_l | 2 < l < 4} do
          if assigned_to_region(ξ_l) ∧
          in_component_region(ξ_i, ω_{j,k}) ∧
          in_component_region(ξ_l, ω_{j,k}) then
            add_unit_to_region(region(ξ_l), ξ_i)
            #of_assigned_neighbours+
          end if
          if #of_assigned_neighbours > 1 then
            merge(region(ξ_{l,m}), region(ξ_{l,n}))
          end if
        end for
        if #of_assigned_neighbours == 0 then
          add_unit_to_region(region(ξ_i), ξ_i)
        end if
      end if
    end for
  end for
end for
```

4.2 Selection of Feasible Components

Plotting all possible component lines will overload the illustration already at considerably low dimensionalities – even after the aggregation step as shown in Figure 3(a). Additionally the aggregation might be influenced by noisy components that can hardly contribute to any meaningful visualisation because of their spread and diversity. Therefore, we propose a method to select the most feasible components with respect to their distribution over the map. The rationale behind this idea is that component planes which have a structured distribution over the map will be most influential for forming clusters. We define a measurement for the detection of different inter-connected areas. Further, it is detected over how many areas the regions we divided the Component Plane into are spread across.

Algorithm 1 explains the detection of regions for a trained Self-Organising Map. The maximum number of regions of a SOM is its number of units, that is the case when no adjacent units belong to the same component region. For each component and region, every unit of the

map is inspected sequentially. A unit builds a new region iff none of its neighbours is assigned to a region yet. Otherwise the unit is assigned to its neighbours' regions. The special case, where a unit's neighbours are assigned to different regions requires particular treatment in the form of region merging. Thereby, all regions of a unit's neighbours are merged into one region and the unit itself gets assigned to this new region. Once all units are handled, this process yields to a number of region assignments smaller or equal to the number of units $|M|$.

The smaller the number of areas, the more coherent the regions are, which implies a structured spread of the component values over the map. Thus, such components have a strong influence on the global cluster structure, and are more interesting for analysis. The component region ratio ν is given as the fraction of the number of component regions n and the number of areas of adjacent units on the map m in Equation 6.

$$\nu_{c_j} = \frac{n_j}{m_j} \tag{6}$$

To select the most interesting components, we can either choose the n components with the smallest numbers of areas, or select all components that have a ratio smaller than a certain fraction of $\max \nu$.

5 Experiments

To demonstrate the effect of aggregation and selection techniques, we performed experiments on a data set having a fairly high, yet not unmanageable, input dimensionality. The Boston Housing data set comprises 506 instances containing information collected by the U.S Census Service concerning housing in the area of Boston, Massachusetts. The data is described in 13 continuous and one binary attributes. We trained a SOM consisting of $20 \times 16 = 320$ units. The number of component regions n for the Metro Visualisation is set to 6.

Figure 1 shows the Component Planes for some components of the Boston Housing data set (Figure 1(a)) and its discretisation (Figure 1(b)). The variables 'crim', 'nox', and 'dis' are most valuable in terms of the component region ratio ν. Their tighter distribution becomes apparent, particularly after the discretisation step (Figure 1(b)). On the other hand, the ν values for the variables 'tax', 'rad', and 'ptratio' are the lowest in the data set. In terms of a low spread, the 'crim' variable is the best choice for component visualisation in this example, since it is only spread across 8 regions, resulting in a ν value of $\frac{6}{8} = .75$. By contrast, the 'ptratio', is split across 34 different regions on the map, any visualisation based thereon does not seem feasible.

We first applied the Component Lines aggregation technique to the full data set in Figure 3(a), yielding in a rather crowded illustration of all components in one plot. To overcome the negative impact of noisy or 'ill-distributed' components, we applied component selection and applied the aggregation technique on top of that.

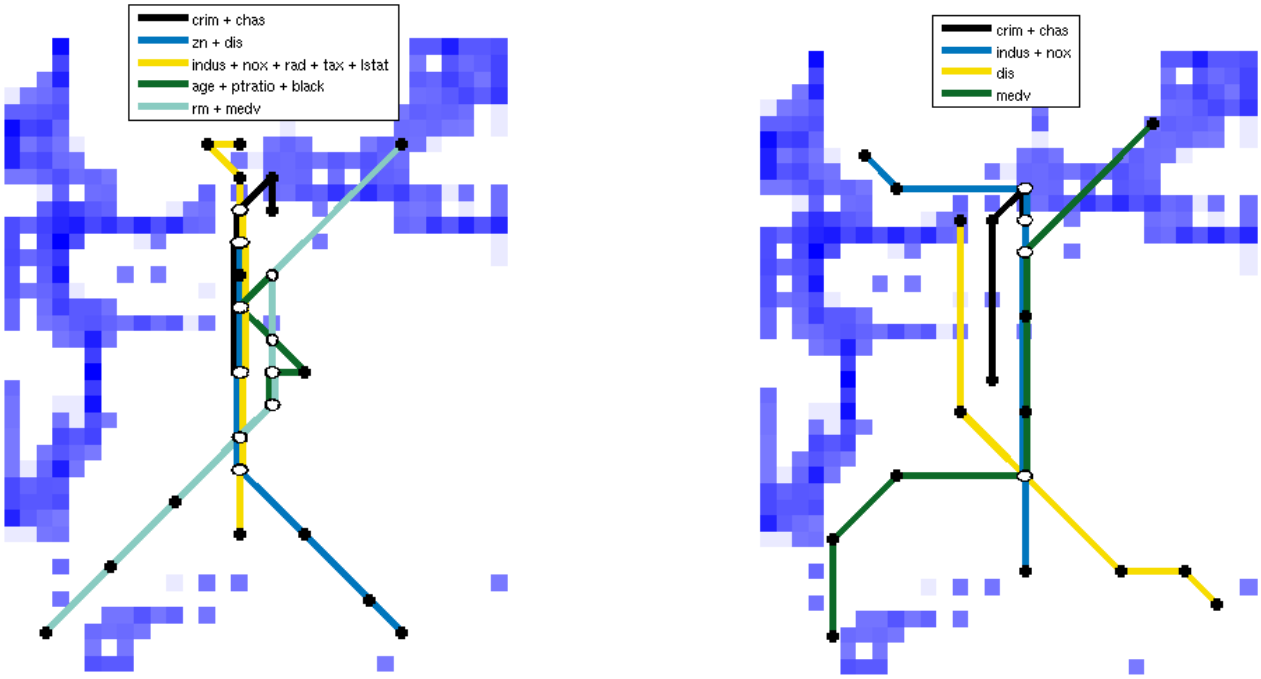
The component selection process we used is parametrised by the top t number of components to select. To that end, the ν values for all components are ranked and the ν value for the t-th component is set as cutoff criterion. Note that this can lead to a number of components selected higher than t; in case of equally valued components, all of them are taken into consideration for visualisation.

Figure 3(b) shows the impact of component selection on the resulting Metro Visualisation. This plot contains the best-suited components in terms of their spreading across the map only. The aggregation performed on top of component selection leads to a rather clear display of the most important groups of components – in accordance with user-defined parameter settings. The negative influence of noisy components is minimised or eliminated. Their shifting effect, i.e. the distortion of the visualisation they have on the important components does not occur.

6 Conclusions and Future Work

We presented a novel method for the detection and selection of most meaningful components with respect to a given SOM clustering. We showed how this approach can be used for the Metro Visualisation, a visualisation technique for the aggregation of multiple component information into one illustration. To this end, the Metro Visualisation utilises a discretisation of single Component Planes. The Metro Visualisation is motivated by the problems arising when trying to plot all components, namely the resulting high number of plots. The experiments presented show that our method is feasible for visualising higher-dimensional feature sets. Furthermore we showed that the pre-selection of single components before the actual aggregation helps in removing noisy components. The resulting visualisation of the aggregated components forms a feasible, abstract way of analysing Self-Organising Map mappings.

Future work will mainly deal with a more thorough investigation of gradient based methods for the detection of feasible components. Another possible research direction are advanced methods for component aggregation and their evaluation. Moreover, possible heuristics for automatically choosing parameter values might further simplify the proposed methods. We further want to investigate the applicability to very high-dimensional data sets, e.g. the clustering of text corpora. Here we will inspect both the scalability of the algorithm as well as the interpretability of the resulting components or terms.



(a) Visualisation of aggregated components

(b) Aggregation on top of component selection

Figure 3: Metro Visualisations of aggregated 3(a) and pre-selected 3(b) component lines for the Boston Housing data

References

[1] Teuvo Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Heidelberg, 1995.

[2] Guanglan Liao, Tielin Shi, Shiyuan Liu, and Jianping Xuan. A novel technique for data visualization based on SOM. In *Proceedings of the 15th International Conference on Artifical Neural Networks (ICANN'05)*, pages 421–426, Warsaw, Poland, September 11-15 2005.

[3] Robert Neumayer, Rudolf Mayer, Georg Pölzlbauer, and Andreas Rauber. The metro visualisation of component planes for self-organising maps. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN'07)*, Orlando, FL, USA, August 12-17 2007. IEEE Computer Society. Accepted for publication.

[4] Elias Pampalk, Andreas Rauber, and Dieter Merkl. Using Smoothed Data Histograms for Cluster Visualization in Self-Organizing Maps. In *Proceedings of 12th the International Conference on Artifical Neural Networks (ICANN'02)*, pages 871–876, Madrid, Spain, August 27-30 2002. Springer.

[5] Georg Pölzlbauer, Michael Dittenbach, and Andreas Rauber. Gradient visualization of grouped component planes on the SOM lattice. In Marie Cottrell, editor, *Proceedings of the Fifth Workshop on Self-Organizing Maps (WSOM'05)*, pages 331–338, Paris, France, September 5-8 2005.

[6] Georg Pölzlbauer, Michael Dittenbach, and Andreas Rauber. Advanced visualization of self-organizing maps with vector fields. *Neural Networks*, 19(6-7):911–922, July-August 2006.

[7] André Skupin. A picture from a thousand words. *Computing in Science and Engineering*, 6(5):84–88, 2004.

[8] Alfred Ultsch. Maps for the visualization of high-dimensional data spaces. In *Proceedings of the 4th Workshop on Self-Organizing Maps (WSOM'03)*, pages 225–230, Kyushu, Japan, September 11-14 2003.

[9] Alfred Ultsch and Hans Peter Siemon. Kohonen's self-organizing feature maps for exploratory data analysis. In *Proceedings of the International Neural Network Conference (INNC'90)*, pages 305–308, Paris, France, July 9-13 1990. Kluwer.

[10] Juha Vesanto. SOM-based data visualization methods. *Intelligent Data Analysis*, 3(2):111–126, 1999.

[11] Juha Vesanto and Jussi Ahola. Hunting for correlations in data using the self-organizing map. In *Proceedings of the International ICSC Congress on Computational Intelligence Methods and Applications (CIMA'99)*, pages 279–285, Rochester, N.Y., USA, June 22-25 1999. ICSC Academic Press.