brains, minds media
ISSN: 1861-1680

REEFSOM - A Metaphoric Data Display for Exploratory Data Mining

Harmen grosse Deters[a], Wiebke Timm[a], Tim W. Nattkemper[a]*

[a] Applied Neuroinformatics Group, Bielefeld University, PO-Box 100131, D-33501 Bielefeld, Germany

* Corresponding Author: Phone: +49-521-1066059, Fax: ++49-521-1066011, email: nattkem@techfak.uni-bielefeld.de

urn:nbn:de:0009-3-3051

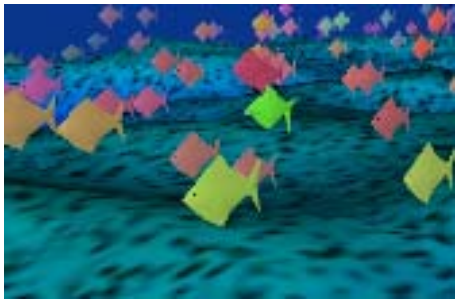
Abstract. In this paper we present a new visualization framework incorporating self organizing maps (SOM) and a metaphoric data glyph approach. The combination of data glyph, U-matrix visualization and SOM creates a virtual 3D underwater cartoon environment from an arbitrary data table. The entire system is called REEFSOM (REndering of Emergent Fish SOM) since the environment simulates an underwater scenario. We propose to use REEFSOM for (a) exploratory data analysis in interdisciplinary research and for (b) teaching in neural information processing. The appeal of the REEFSOM is demonstrated with three case studies.

Keywords: Data mining, exploratory data analysis, information visualization, screen entertainment, self organizing maps, neural networks, neural networks teaching

Citation: grosse Deters H, Timm W, Nattkemper TW (2006). REEFSOM – A Metaphoric Data Display for Exploratory Data Mining. Brains, Minds, and Media, Vol.2, bmm305 (urn:nbn:de:0009-3-3051).

Received January 13, 2006; Accepted April 5, 2006; Published April 28, 2006

Supplementary Material



Article Resources

Additional Resources

Installation Guidelines

Datasheet

1 Introduction

The self-organizing map (SOM) proposed by Kohonen (1982, 2000) is nowadays one of the most prominent architectures for dimension reduction, clustering and visualization. Although the SOM can be outperformed regarding the three application aspects clustering, classification and dimension reduction by other approaches (Flexer 2001) it gained a remarkable popularity especially in the field of data mining and exploratory data analysis. In the last two decades more than 5000 articles have been published about applications and advances in the SOM algorithm (Kohonen et al. 1998, Oja et al. 2003). One reason for this popularity might be that the SOM comprises the above three aspects of data analysis in one architecture and that it is straight forward to implement. Another reason is that there is still no straightforward scheme for designing information visualization systems for multivariate *N*-dimensional data. Interestingly, scientific textbooks about information visualization have been published in the last six years (Card et al. 1999, Spence 2000, Fayyad et al. 2001, Ware 2004, Chen 2004).

Next to the SOM one of the most proposed and straightforward visualization approach to analyze N variables in m observations is to map the variables to the attributes (in general shape, size, color and location) of graphically displayed entities, so called glyphs. However, the comprehensive glyph display for the entire set of *m* samples has just been identified as an interesting scientific engineering problem (Ward 2002):

> *"The placement or layout of glyphs on a display can communicate significant information regarding the data values themselves as well as relationships between data points, ...".*

SOMs are frequently used for exploratory data analysis in data mining projects. Such projects are carried out in interdisciplinary fashion between computer science experts and partner experts from the fields of biology, medicine, finance and so on. After applying the SOM the result needs to be visualized to give insight into the high dimensional data structure. So a meaningful SOM visualization is important for a fruitful interdisciplinary discussion. Nevertheless, we observed in many projects that explaining the meaning of a SOM visualization is often time consuming in the beginning and can become even frustrating for both parties. One reason for this is that explaining the SOM contains a lot of standard vocabulary from the fields of algebra, pattern recognition or artificial neural networks, sometimes unknown to the research collaborators from the fields of biomedicine or economics. In fact, we often observed that the partners had interpretations of terms like *pattern, vector or similarity* which were different from the concepts in the fields of pattern recognition or artificial neural networks.

In addition, the data structure itself can be quite complex and hard to grasp even for an experienced data analyst, for instance regarding the identification of important features. This problem gets serious for SOMs of large numbers of nodes and/or a large data dimension. One also has to consider, that in most data mining projects the data may have *several* structural features to be discovered. Especially the analysis of heterogeneous clusters and outliers can be time consuming. So information analysts may spend some time with the data and its visualizations which can be quite tiresome and boring. In this case one would benefit from displays that catch the attention of the user again and again. This favorable display quality could be called *entertaining*. These observations motivate the research for new SOM visualization strategies to support exploratory data analysis. Before we outline our approach we will give a short motivation.

One of the most powerful tools for explaining structures or relations to people with a different background knowledge is the metaphor, i.e. to compare two seemingly unrelated subjects. Its

explanatory power lies in the opportunity to describe one subject (the SOM) by the comparison with a familiar real world subject, well known to both parties. The basic idea behind this work was to design a metaphoric SOM visualization tool, where the data structure is interpreted as a cartoon for a natural scene, in this case an underwater scenario with a reef full of fishes. An approach for computing a metaphoric description of a projection result would be of valuable help to the computer scientist to discuss his results with his collaborative partners from biology, chemistry etc. Based on a first proposition of the basic idea (Nattkemper 2005) we now present the first version of an integrated software tool. Its usefulness is illustrated with three example applications and discussions of the result visualizations obtained.

To generate a metaphoric display for an arbitrary data set one needs to set up a model of an environment and define a function to map the data values to the model parameters. The basic idea of presenting complex data structures in a metaphoric way as an environment is in principle not new. In the 90s some attempts were made to simulate a natural environment (for instance an office table[1] or a living room[2]) and to use this simulation as an alternative interface to the command level / operating system. These approaches aimed boldly at a metaphoric relation between the graphics and the function (cupboard with sliders = file directory, mailbox = email). The overall aim was to lower the borderline between non-expert personal computer users and the software components installed on the computer. The success of these attempts to overcome some user's inhibitions for integrating the personal computer in their everyday live was limited. Of course this attempt had to fail since a relation on such a level can not be achieved for all functions necessary in a user interface of a personal computer. Some metaphors were straightforward, like the mailbox for starting an email program, or a writing desk to start a text editor program. Some metaphors were not that clear, of course. The proposed metaphoric approach is substantially different from the early proposed ones.

Another more basic problem can be identified looking at the perceptual and cognitive background of information visualization as described by C. Ware in his book (Ware 2004). He argues that the effectiveness of visualization depends on two aspects, arbitrary cultural convention and perception. The perceptual effects result from the basic psychophysical and biological mechanisms in visual perception, like perceiving two bars on a screen as connected or unconnected or perceiving two pairs of colors as having the same pairwise color similarity (the problem in generating perceptually uniform color codes).

The arbitrary cultural conventions are the results of some long term processes of visualization coding inside a social community. The big difference to the perceptual effects is, that in principle, any graphical construct can be linked to any meaning. As a consequence for instance, the basic colors have very different meanings in different cultures. Successful information visualization systems are based solely on perceptual cues or they are hybrids, incorporating both mechanisms, perception and arbitrary cultural convention. A visualization just based on cultural convention must lack on uniformity in the interpretation by members of different social communities.

The visualization approach presented in this paper is to combine both effects, perception and convention, by rendering hybrid metaphoric information visualization displays of complex multivariate datasets from trained self organizing maps (SOM). The visualization aims at an integrative approach

[1] General Magic. Magicap, 1994. Sunnyvale CA.
[2] Microsoft Corporation. Bob, 1995. Seattle WA.

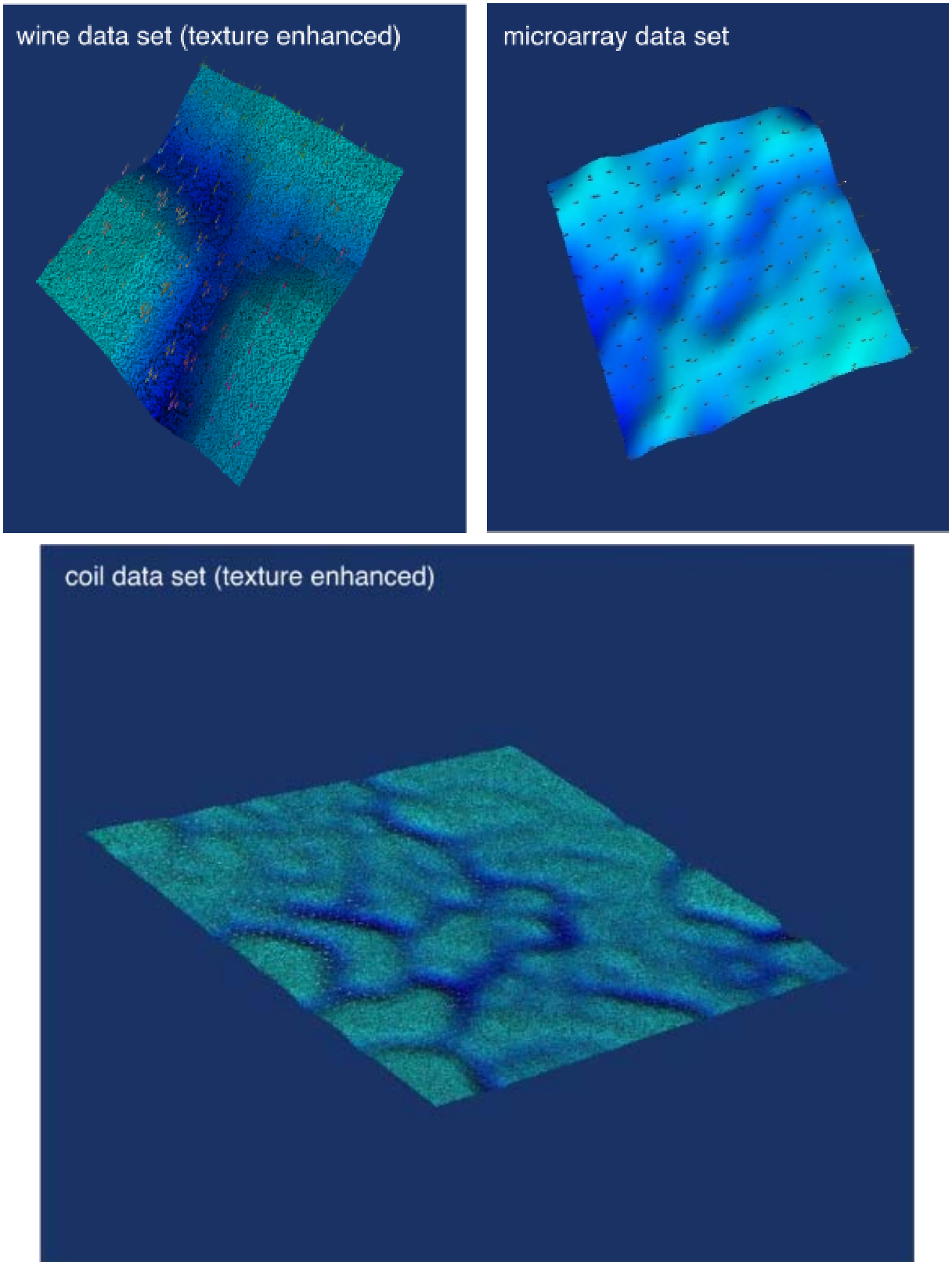


Figure 1: The U-matrix approach is used to render the sea bed of an underwater habitat. Deep and blue valleys represent feature space regions with a lesser data density, i.e. larger inter node distances in the SOM. The fishes represent the prototype vectors of the SOM nodes. To enhance the three dimensional structure of the sea bed, a texture can be rendered on the surface, as displayed in the middle and the left case. Information about the data sets is given in the Applications section.

for simultaneous analysis of global as well as local data features. The overall metaphor used is a natural habitat with one type of animal, in this particular case a reef and coral fishes.

The reasons for selecting an underwater habitat are more or less arbitrary. First, underwater scenarios gained some popularity and are increasingly distributed across all kinds of media, from motion pictures to documentary films on TV. Second, fish live as loners as well as in swarms, thus clusters of fish as well as single fish appear as natural. And in data mining applications one is usually interested in both structural features of data: clusters and outliers. Third, fishes have lots of features that can be parameterized straightforward and easily. In biology, fish species are generally grouped in clusters according to their physiological features (like blue whiptail, paradise whiptail, double whiptail and so on) and each cluster can be represented by a prototype (i.e. whiptail, Lieske and Myers 2002) similar to the idea of vector quantization.

The sea bed of the reef is rendered to display the global features of the data set. To this end a SOM with a large number of nodes is trained and visualized as described in the second section. Following Shneiderman's famous mantra for interactive visual data exploration (*"Overview first, zoom and filter, then details on demand"*, Shneiderman 1996) the drill down to an analysis of single data items is realized using the strategy of glyph representation (see Keim 2002 for an overview). A new glyph type suiting the idea of an underwater habitat is introduced in section three. The interface of the integrated software tool is explained in section four, followed by some example applications in section five. The last section sums up the results, first experiences and conclusion.

2 SOM-based sea bed rendering

The self-organizing map (SOM or Kohonen map) as proposed in (Kohonen 1989) provides an unsupervised learning algorithm for dimension reduction and visualization which is easy to implement (Kohonen 2000). The SOM consists of a grid of N x M ordered nodes $\mathbf{n}_{n,m}^{(k)}$, each associated with a prototype vector $\mathbf{u}^{(k)}$. The prototype vectors are of the same dimension as the feature vectors $\{\mathbf{x}^{(i)}\}$ of the multivariate training data set. The training scheme of the SOM is similar to the online *k*-means clustering. In each learning step, a training example $\mathbf{x}^{(i)}$ is selected and the nearest neighbor node $\mathbf{n}_{n',m'}^{(\kappa)}$ fasdfbfb

(also referred to as *winner node*) is identified, evaluating:

$$\kappa = \arg\min_{k}\{|\mathbf{u}^{(k)} - \mathbf{x}^{(i)}|^2\}.$$

The prototype vectors $\mathbf{u}^{(k)}$ and its neighbors are updated using the following formula:

$$\mathbf{u}^{(k)}(t+1) = \mathbf{u}^{(k)}(t) + h_{k\kappa}(t)[\mathbf{u}^{(i)} - \mathbf{u}^{(k)}], \forall k,$$

where *t* = 0, 1, 2, ... is an integer time coordinate which represents the iterations of the training process. The function $h_{k\kappa}(t)$ acts as a neighborhood function on the grid, centered at the winner node

grid position $\mathbf{n}^{(\kappa)}$. For the solution to converge, it is necessary that $h_{i\kappa} \to 0$ for increasing t. In the literature, $h_{k\kappa}$ is frequently defined in terms of the Gaussian function,

$$h_{k\kappa}(t) = \alpha(t) \exp\left(-\frac{||\mathbf{n}^{(\kappa)} - \mathbf{n}^{(k)}||^2}{2\sigma^2(t)}\right)$$

for grid nodes $\mathbf{n}^{(k)}$. The function $\alpha(t)$ is another scalar valued termed as learning rate factor, and the parameter $\sigma(t)$ defines the width of the neighborhood function. Both $\alpha(t)$ and $\sigma(t)$ are monotonically decreasing functions of time. Some authors proposed to use other grid topologies than a regular square grid, like a hexagonal one or a torus to avoid quantization problems at the edge of the grid. However, in this work we may focus on the standard square grid, since this work does not aim at the best possible SOM training result but on a new visualization tool for SOMs.

To visualize the trained SOM, several approaches have been proposed: The feature density of the trained SOM prototype vectors is displayed based on smoothed histograms (Vesanto 1999), the U-matrix (Ultsch 1993), or by clustering the prototype vectors (Vesanto and Alhoneimi 2000, Wu and Chow 2004). For the special case of very large SOMs, fish eye view or fractal view have been proposed (Yang et al. 1999). In addition, the SOM visualization can be augmented by text labels, as for instance the WEBSOM (Honkela et al. 1997) or a single feature analysis with a component plane view (Kaski et al. 1998). Also automatic feature selection has been proposed to render icons for displaying the SOM prototype vectors on a grid (Rauber and Merkl 2001).

The U-matrix as proposed by Ultsch (1993) is probably the most applied visualization framework for SOM, especially for SOM with a large number of neurons. The U-matrix visualizes the data structure by a display of approximated data densities at the SOM grid nodes. To this end, pairwise distances between SOM node prototype vectors are computed and arranged in a low-dimensional array at positions corresponding to the grid node positions. These intensities are displayed by a height profile or by a colored plane (or by both). Thus, the U-matrix itself can already give a metaphoric description of the data density by an image of mountains. In this work we visualize the U-matrix as a colored height profile. We use a color scale which has been adjusted manually to simulate the color changes of the sea bed depending on the depth, i.e. a scale from *cyan* to *blue* to *black*.

In most applications the U-matrix is displayed as a height profile, with the height being proportional to the distance between prototype vectors. So in the display clusters of very different features are separated by a ridge of mountains. Since we consider an underwater scenario we visualize the U-matrix the other way round, i.e. we draw the *depths* of the sea bed proportional to the feature distances. An example of three sea beds computed for three training sets is shown in Figure 1. A description of the training sets is given in the later Applications section.

3 The fish glyph

Glyphs (or icons) are parameterized geometrical models that are used for an integrated display of multivariate data items. The idea is to map the variables of one data item to the parameters of one glyph so that the visual appearance of the glyph encodes the data variables.

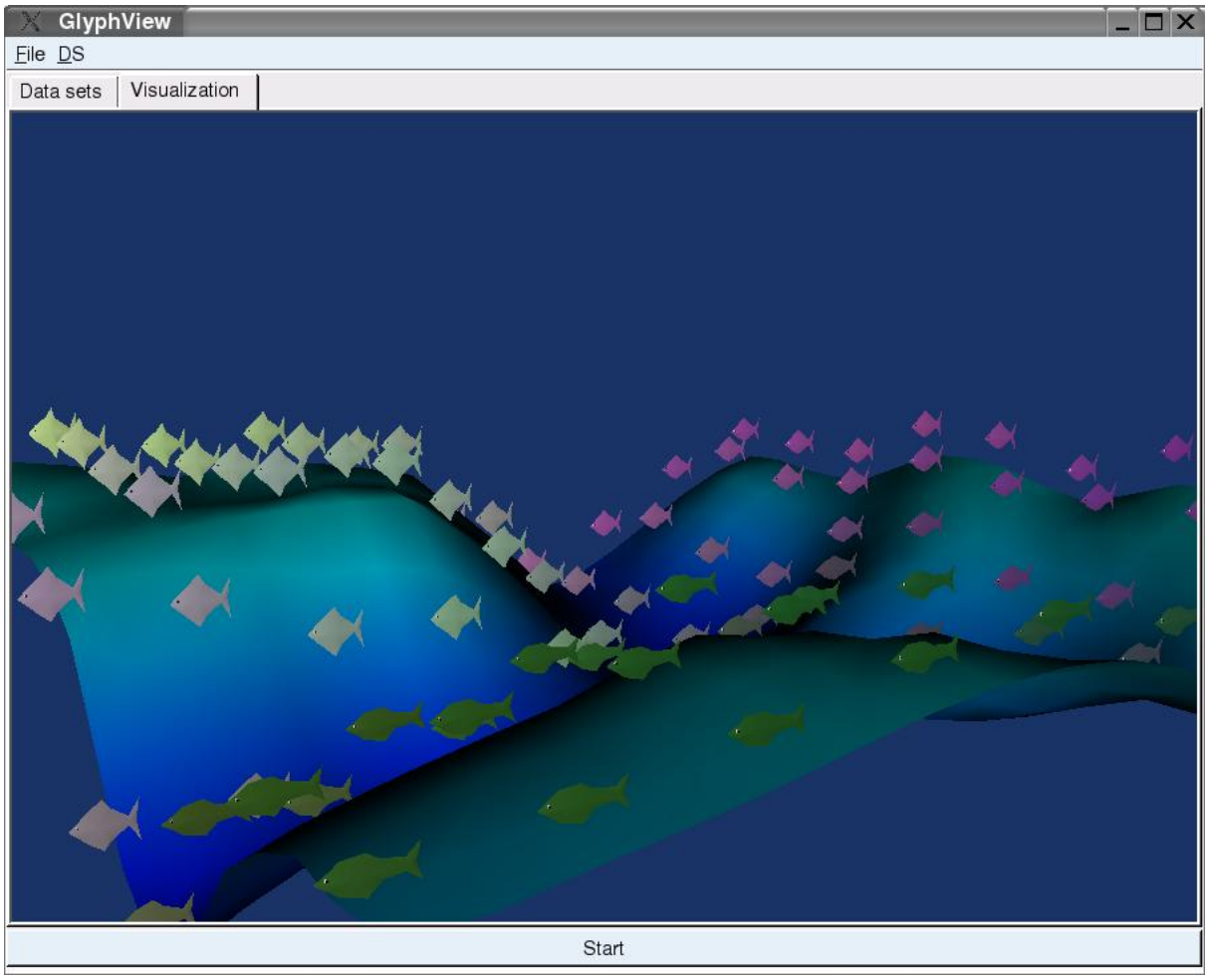


Figure 2: A snapshot of a REEFSOM visualization. A SOM is trained on the wine dataset (see Applications and Results for details) and visualized using the REEFSOM software tool. The data set is separated into three densely clustered regions which can be identified as three plateaus. The data items in these regions are displayed by fish glyphs. The typical features for each feature space region can be easily identified by color and/or shape features.

Glyph approaches can be classified as being *abstract* or *metaphoric*. Abstract glyphs are basic geometric models without direct symbolic or semantic interpretation like profiles (Du Toit et al. 1986), stars (Siegel et al. 1972), and boxes (Hartigan 1975). To display more variables or also data relations, abstract glyphs can get quite complex like the customized glyphs (Ribarsky et al. 1994, Kraus and Ertl 2001), shapes (Shaw et al. 1999) or infochrystals (Spoerri 1993). Such glyphs can be powerful tools for a compact display of a large number of variables and relations. However, the user must spend considerable time for training to be able to use these tools effectively. Since the idea of using metaphoric display is quite natural, metaphoric glyphs have been proposed in the earliest years of information visualization already. In 1970, the well known Chernoff faces (Chernoff 1971) were introduced for multivariate data display. The idea of rendering data faces may get new stimuli from advances in computer graphics and animation (Noh and Neumann 1998) since a large range of algorithms exist to render faces in different emotional states. However, the successful application of Chernoff faces seems to be restricted to data with a one-dimensional substructure, like social and economic parameters as in (Dorling 1994, Alexa and Mueller 1998, Smith et al. 2002). Similar approaches use stick figures (Pickett and Grinstein 1988), a parameterized tree (Kleiner and Hartigan 1981) or wheels (Chua and Eick 1998). To visualize the SOM in a metaphoric manner, we need to

synchronize the designs of the U-matrix landscape and the data glyphs. To this end we developed a fish shaped glyph.

The fish glyph is used to display (i) the prototypes of the SOM or (ii) all the items of the data set or (iii) both. This visualization mode can be chosen in the graphical user interface (see section 4 and Figure 3 for details). In mode (ii) and (iii) the data set items are to be visualized on top of the sea bed, i.e. the SOM. But, the computation of an appropriate two dimensional grid position for each data item on the SOM (relative to the SOM node coordinates) is a nontrivial problem. The most naive approach is to take the grid coordinates of the winner node $\mathbf{n}^{(\kappa)}$. This approach must fail, if the number of data items per winner node exceeds one, since in this case two fishes must be rendered at the same position. A more advanced solution is to interpolate the two dimensional position of $\mathbf{x}^{(i)}$ from the grid node positions of several nodes. In the literature, some approaches have been proposed, most of them applying advanced interpolation algorithms. In our first version of the software, we disclaim an exact positioning of the data items $\mathbf{x}^{(i)}$ on the SOM and render each data item at a random position in the close vicinity of its winner node. On first sight, this strategy looks a bit crude, but it is motivated by several arguments. First, several solutions to the interpolation problem have been proposed and there is not one solution which is accepted by the entire community. Second, one important feature of each data item is its cluster prototype, i.e. its nearest neighbor. If the interpolation leads to suboptimal results, the data item, or its glyph, is rendered at a position closer to another node $\mathbf{n}^{(k)}, k \neq \kappa$ which makes it visually infeasible to identify the winner node correctly. Third, the random strategy is the computationally least expensive one. In this proposed software version, the fish is rendered based on a grid model which has 17 graphical attributes. They consist of 14 geometric parameters (6 angles and 8 arc length) and three color values (RGB) as displayed in Fig.4. In Fig. 2 a REEFSOM snapshot shows, how the color and shape of fishes, rendered on top of a U-matrix sea bed, varies. In a first attempt we proposed a simpler fish model (Nattkemper 2005) which led to quite unnatural fish shapes. The new redesign guarantees glyphs with biologically plausible shapes.

4 Software features

As already summarized in (Levkowitz 1997, Ward 2002), humans' abilities for perceiving graphical attributes of glyphs vary considerably. Thus, the software is designed to allow a convenient customization of mapping variables to graphical parameters.

The graphical user interface (GUI) of the REEFSOM consists of two windows, the visualization GUI and the parameter GUI. The Visualization GUI has two modes which are selected by two tabs (Figure 3 a) and d)). The first mode is shown in Figure 3. In this mode, the user selects data sets and trained SOMs for an exploration session (Figure 3 b). In the pull down menu DS (Figure 3 c) the user can apply different normalization procedures to the data matrix. The data matrix can be normalized to a range of [0; 1]. This can be done for the entire data set (which is sensitive to outliers) or for each variable separately. After selecting and preprocessing one data set the variables $x_j^{(i)}$, j =1,…,n of the n-dimensional feature vectors $\mathbf{x}^{(i)} \in [0;1]^n$ are associated to the graphical fish parameters p_k, k = 0,…,16. Three parameters set the red, green and blue color hue of the fish.

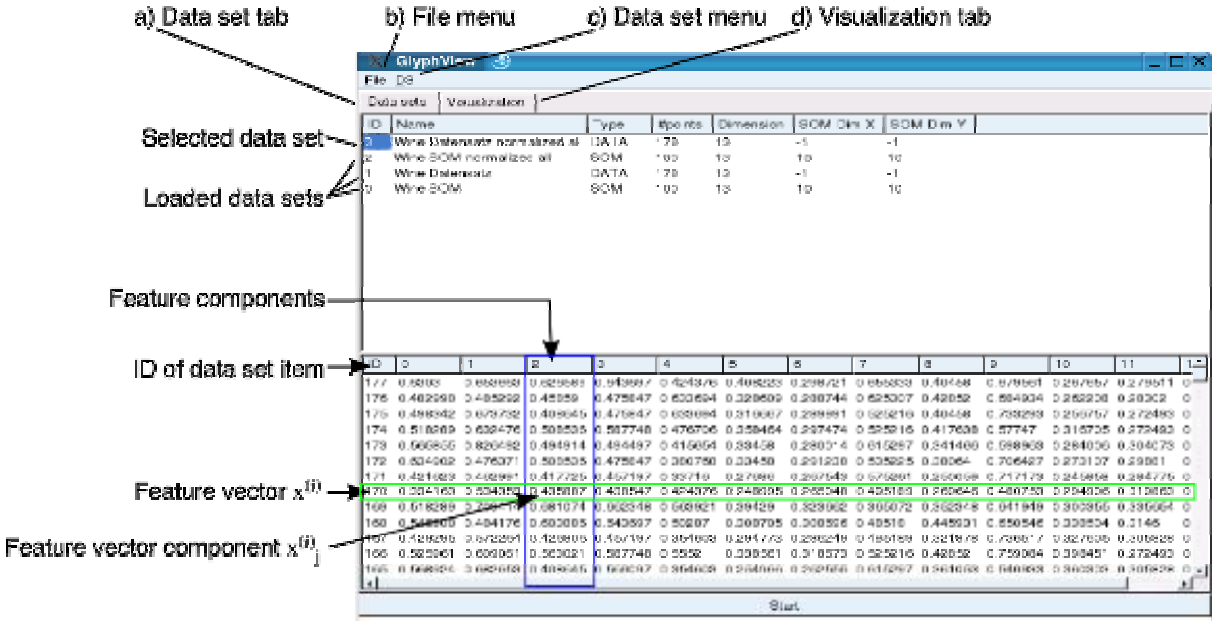


Figure 3: The visualization GUI: Tabs a) and d) are used to switch between the two modes Data sets and Visualization of the GUI. In the Data sets mode, the user reads data sets into the program and applies selected normalization steps to the data. The user can also choose the visualization mode and load new data sets via the pull-down menus b) and c). In the lower half of the window the data matrix is displayed for the purpose of visual control. If the visualization mode is selected by activating d), the SOM is displayed in this window as shown in Figure 2 or Figure 5 and Figure 6.

The remaining fourteen parameters determine the geometrical shape of the fish as displayed in Figure 4. In the fish GUI four additional parameters can be tuned to improve the visualization (see lower part of Figure 4):

- the distance between the fishes and the sea-bed
- the distance between nodes (or the size of the sea-bed)
- the distance between the highest and lowest U-matrix point
- the level of detail (to enhance the exploration)

The GUI is implemented using the QT toolbox[3], the REEFSOM is rendered using the Visualization Toolkit (VTK)[4].

To spare time, REEFSOM can compute a default mapping of variables to parameters. For each of the variables the variance is calculated. The variables are ranked with decreasing variance and mapped to glyph parameters according to their rank positions. The order of the glyph parameters is: Red, Green, Blue, ark lengths, angles.

[3] Trolltech. http://www.trolltech.com/products/qt/index.html.
[4] Kitware. http://public.kitware.com/VTK/.

5 Applications

The SOM reef is computed and displayed for three data sets.

Wine data set: This data set of 178 items is a result of a chemical analysis of 178 italian wines (Forina 1991). The 13 variables describe the continuous values of chemical properties like x_0 = alcohol, x_1 = malic acid, x_2 = ash, x_3 = alcalinity of ash, x_4 = magnesium etc. The wines are classified into three different classes. The result is shown in Figure 5 as a flight into the SOM.

Breast cancer microarray data: We use the data of the van't Veer study (van't Veer 2002). Around 25000 expression levels of genes were analyzed in 78 primary breast samples. For each gene and sample the logarithm of basis 10 of the intensity and the ratio (in [-2, 2]) are provided. In three steps, the original gene pool was reduced (mainly by using statistical methods) to 5000, 230 and finally 70 genes forming sets of marker genes for the prediction of breast cancer outcome. For visual cluster analysis, a 15 x 15 SOM was applied to the data set of n = 70 genes, with each component $x_j^{(i)}$ representing the expression level of gene j in patient tissue sample i. The result is displayed in Figure 6.

COIL data set: The COlumbia Image Library (COIL) provides images of 20 different objects viewed from different directions (Nene et al. 1996). On the entire data set a principal component analysis (PCA) is performed. The eigenvectors of the ten largest eigenvalues account for most of the signal intensity variance and are used to project each image to a ten dimensional vector. A 50 x 50 SOM is trained with this set and the result SOM is displayed as a SOM reef in Figure 1.

5.1 Results

In the wine data application the sea bed shows three plateaus divided by a y-shaped abyss. Using the parameter mapping window we tried out different variable selections for the three color basics. The idea is to find variable selections for fish colors that correspond to the shape of the sea-bed. Such a selection would help to identify interesting cluster specific variables. In this example, we rapidly found, that the selection RED = *alcohol*, GREEN = *alkalinity of ash* and BLUE = *flavanoids* results in fish colors that fit to the sea-bed. In Figure 5 a zoom into the visualization is shown. In Figure 5 c) one can easily identify a swarm of green/yellow fishes (above the front plateau), a swarm of magenta/red fishes in the back left and a blue/cyan fish swarm in the back right. The data seems to have a clear global structure especially regarding these three variables. And inside the swarms, one can observe to which extend the other variables determine the fish shape (as for instance the bottom fins of the two yellow-green fishes in the front in figure d)). Also local outliers can be identified easily as for instance the few green fishes in the blue/magenta swarms in the back of the reef (see Figure 5 e). We observed that although the color of the fishes dominate the preattentional perception of the fish swarms, the user is able to evaluate the other shape features also, especially inside a swarm with a more or less homogeneous color.

In case of the microarray data, the number of variables is much higher as for the wine data. Thus, the automatic variable selection for the geometric parameters is selected. The three features with the strongest variance are mapped to the colors. The sea-bed visualization is enhanced by activating the texture mapping option. This is done since the u-matrix has not such a clear structure (see Figure 6) as the wine data reef. Zooming into the reef, we observe that the fishes change their colors from

green/yellow to red to magenta to blue (see Figure 6 c) and d)). Browsing though the fishes we see that the shape stays quite stable, except for some outliers (see the green fish with the different 'nose' in the upper middle of Figure 6 e). So the global structure of the data seems to be strongly determined by the three variables mapped to colors, but the local data features can be easily identified again.

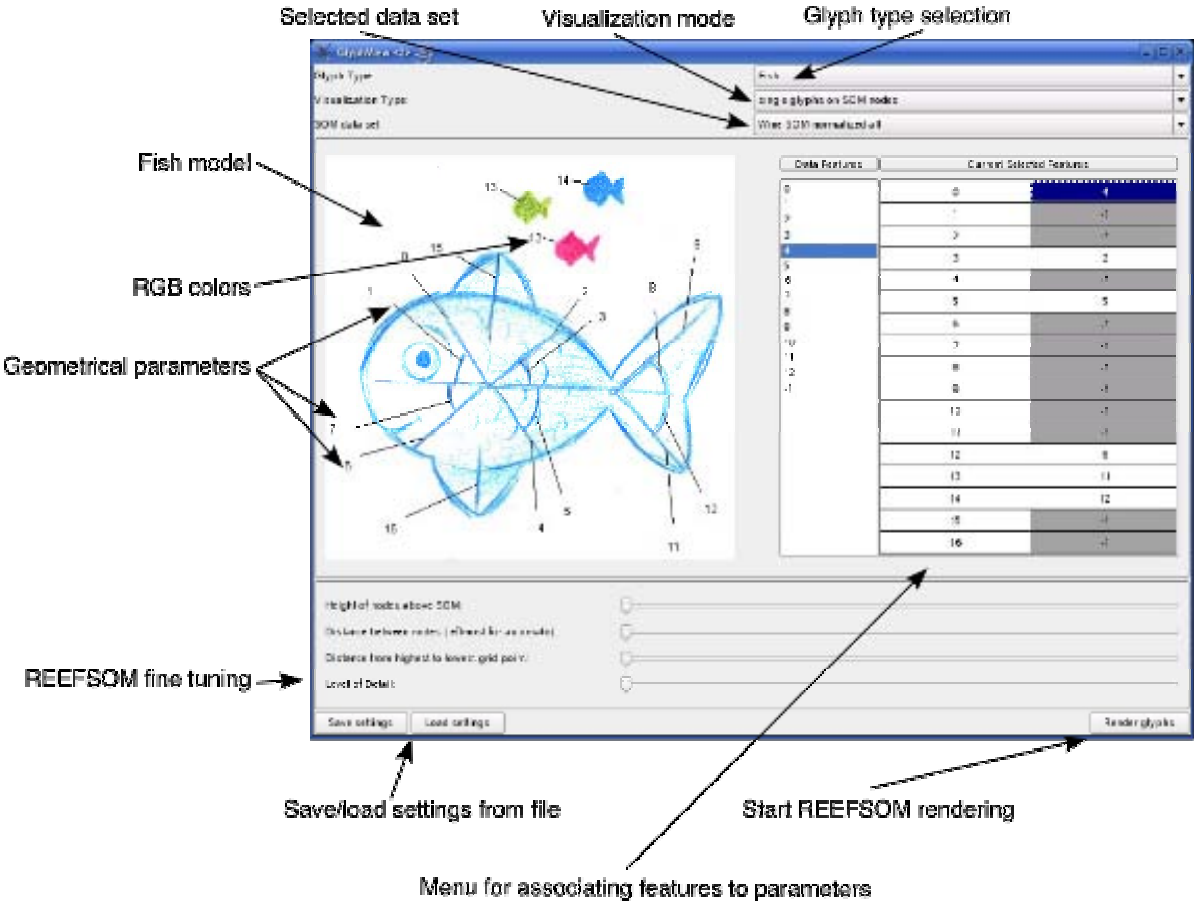


Figure 4: The fish glyph GUI: On the top the data set, the visualization mode and the glyph type are selected. In the left half a fish cartoon shows the geometrical parameters p_k of the fish glyph. This is used for associating the single variables $x_j^{(i)}$ to the parameters. Parameters p_{12} to p_{14} encode the RGB color of the fish. For a manual mapping to the parameters, the variables are displayed in the left column (in this case j = 0, ..., 12), the 17 fish glyph parameters in the middle column and feature components associated to the particular parameter in the right column. A value of -1 encodes, that no variable is associated to this parameter. In this case a default value is taken. In the lower part of the GUI, fine tuning can be applied, parameter settings can be stored and loaded and the rendering process of the REEFSOM can be triggered.

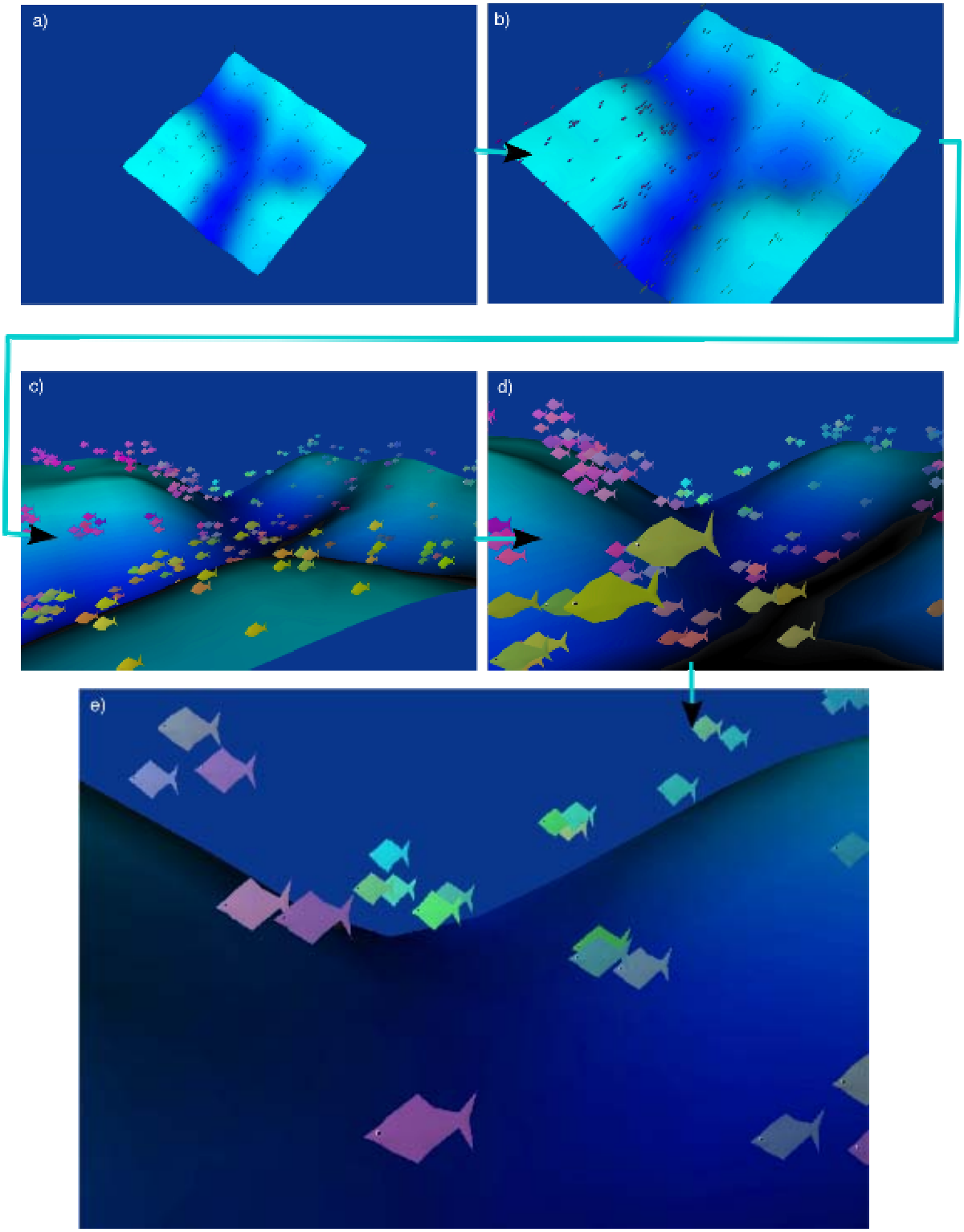


Figure 5: A flight into the REEFSOM of the wine data set is shown. The sea bed shows the three cluster structure of the data as three reefs divided by an abyss. A detailed discussion of the results regarding the glyphs can be found in the Results section.

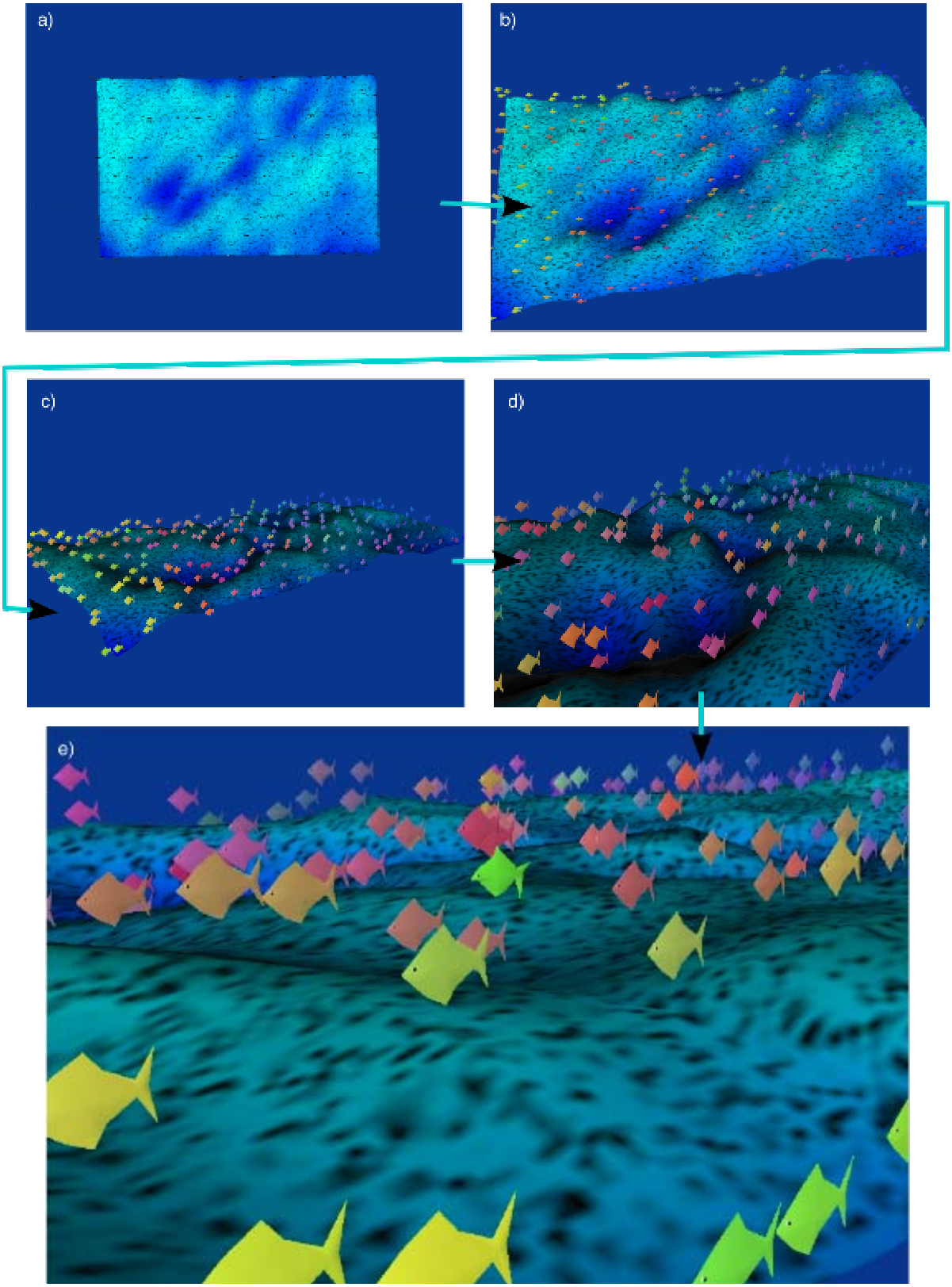


Figure 6: A flight into the REEFSOM of the microarray data set is shown. An inspection of the data points, i.e. the fish glyphs reveals, that the fishes hardly form clusters in isolated regions and are also placed in the abyss. See the Results section for details.

6 Summary and Discussion

A new approach for SOM visualization has been proposed. In contrast to other works, the approach aims at a metaphoric explanation of the SOM to non-expert observers. The metaphoric display consists of visualizing the SOM U-matrix as an underwater sea bed using color and texture plus rendering single feature vectors as fish shaped glyphs. The glyph interface allows easy and convenient mapping of variables to glyph parameters. The examples show, that shape and color of the fishes can represent feature variables and the appealing look of the REEFSOM. We believe that the REEFSOM will improve SOM based data analysis by (a) making the SOM inspection more entertaining and (b) providing easy-to-interpret metaphoric SOM display for non-expert users. Since interesting variables can be identified using the REEFSOM, we implemented an additional option to support the analysis of single variables. Instead of the u-matrix the user can choose one of the component planes (Kohonen 2000) to be rendered as a sea-bed. A component plane is a display of a grid of one selected component *i* of the prototype vectors $\mathbf{u}^{(k)_i}$. For a SOM trained on a D-dimensional data set, the user can select one of the D component planes (please see the supplementary material of this article for further download information). They demonstrate that REEFSOM is easy to use, entertaining and a valuable contribution for bridging the gap between neural networks and data mining applications. The supplementary material also offers an executable software demo and further information. A first prototype of the system has been presented on the 5th Workshop on Self-Organizing Maps (Nattkemper 2005).

References

Alexa M and Mueller W (1998). Visualization by metamorphosis. In Wittenbrink CM and Varshney A, editors, IEEE Visualization 1998 Late Breaking Hot Topics Proceedings, pages 33–36.

Card SK, Mackinlay JD, and Shneiderman B (1999). Readings in Information Visualization. Morgan Kaufmann Publishers.

Chen C (2004). Information Visualisation & Virtual Environments. Springer.

Chernoff H (1971). The use of faces to represent points in n-dimensional pace graphically. Technical Report RN NR-042-993, Dept. of Stat., Stanford Univ.

Chua M and Eick S (1998). Information rich glyphs for software management. IEEE Computer Graphics and Applications, 18:24–9.

Dorling D (1994). Cartograms for visualizing human geography. In Hearnshaw HM and Unwin DJ, editors, Visualization in geographical Information Systems, pages 85–102, Chichester. John Wiley & Sons.

Fayyad U, Grinstein GG, and Wierse A (2001). Information Visualization in Data Mining and Knowledge Discovery. Morgan Kaufmann Publishers.

Flexer A (2001). On the use of self-organizing maps for clustering and visualization. Intelligent Data Analysis, 5(5):373–384.

Forina M et al. (1991). Arvus - an extendible package for data exploration, classification and correlation. http://www.radwin.org/michael/projects/learning/about-wine.html.

Hartigan J (1975). Printergraphics for clustering. Journal of Statistical Computing and Simulation, 4:187–213.

Honkela T, Kaski S, Lagus K, and Kohonen T (1997). Websom - self-organizing maps of document collections.In Proc. of WSOM.

Kaski S, Nikkil J, and Kohonen T (1998). Methods for interpreting a self-organized map in data analysis. In Proc. of ESANN 1998.

Keim DA (2002). Information visualization and visual data mining. IEEE Trans. Visualization and ComputerGraphics, 7(1).

Kleiner B and Hartigan J (1981). Representing points in many dimension by trees and castles. J. Am. Stat. Ass., 76:260–9.

Kohonen T (1982). Self-organized formation of topologically correct feature maps. Biol. Cybernetics, 43:59–69.

Kohonen T (1989). Self-Organization and Associative Memory. Springer-Verlag, Berlin.

Kohonen T (2000). Self-Organizing Maps, volume 30 of Series in Information Sciences. Springer.

Kohonen T, Kaski S, and Kangas J (1998). Bibliography of self-organizing map (som) papers: 1981–1997. Neural Computing Surveys, 1:1–47, 1998. http://www.cse.ucsc.edu/NCS/.

Kraus M and Ertl T (2001). Interactive data exploration with customized glyphs. In Skala V, editor, WSCG 2001 Conference Proceedings, 2001.

Levkowitz H (1997). Color Theory and Modeling for Computer Graphics, Visualization and Computer Graphics. Kluwer, Boston, USA.

Lieske E and Myers R (2002). Coral Reef Fishes: Indo-Pacific and Caribbean. Princeton University Press.

Nattkemper TW (2005). The som reef - a new metaphoric visualization approach for self organizing maps. In Proc. of WSOM 2005, 5th Workshop On Self-Organizing Maps, Paris, France.

Nene SA, Nayar SK, and Murase H (1996). Columbia object image library (coil-20). Technical report, Columbia University.

Noh JY and Neumann U (1998). A survey of facial modeling and animation techniques. Technical Report 99-705, USC Technical Report.

Oja M, Kaski S, and Kohonen T (2003). Bibliography of self-organizing map (som) papers: 1998-2001 addendum. Neural Computing Surveys, 3:1–156. http://www.cse.ucsc.edu/NCS/.

Pickett RM and Grinstein GG (1988). Iconographics displays for visualizing multidimensional data. In Proc. IEEE Conference on Systems, Man, and Cybernetics, pages 514–19.

Rauber A and Merkl D (2001). Automatic labeling of self-organizing maps for information retrieval. Journal of Systems Research and Information Systems (JSRIS), 10(10):23–45, December 2001.

Ribarsky MW, Ayers E, Eble J, and Mukherjea S (1994). Glyphmaker: Creating customized visualizations of complex data. IEEE Computer, 27(7):57–64, July 1994.

Shaw CD, Hall JA, Blahut C, Ebert DS, and Roberts DA (1999). Using shape to visualize multivariate data. In Workshop on New Paradigms in Information Visualization and Manipulation, pages 17–20.

Shneiderman B (1996). The eyes have it: a task by data type taxonomy for information visualization. In Proc. of the IEEE Symp. on Vis. Lang., pages 336–43.

Siegel J, Farrell E, Goldwyn R, and Friedman H (1972). The surgical implication of physiologic patterns in myocardial infarction shock. Surgery, 72:126–41.

Smith M, Taffler JR, and White L (2002). Cartoon graphics in the communication of accounting information for management decision making. Journal of Applied Management Accounting Research, 1(1):31–50.

Spence R (2000). Information Visualization. Addison Wesley Longman.

Spoerri A (1993). Infocrystal: a visual tool for information retrieval & management. In Proceedings of the second international conference on Information and knowledge management, Washington, D.C., United States. ACM Press.

du Toit S, Steyn A, and Stumpf R (1986). Graphical exploratory data analysis. Springer.

Ultsch A (1993). Self organizing neural networks for visualization and classification. In Opitz O, Lausen B, and Klar R, editors, Information and Classification, pages 307–13. Springer.

van't Veer LJ, Dai H, van de Vijver MJ, He YD, Hart AA, Mao M, Peterse HL, van der Kooy K, and Marton MJ (2002). Gene expression profiling predicts clinical outcome of breast cancer. Nature, 415:530–6. The data is available for download via Rosetta Inpharmatics LLC: http://www.rii.com/publications/2002/vantveer.html

Vesanto J (1999). Som-based visualization methods. Intelligent Data Analysis, 3:111–126.

Vesanto J and Alhoneimi E (2000). Clustering of the self-organizing map. IEEE Transactions on Neural Networks,11:586–600.

Ward MO (2002). A taxonomy of glyph placement strategies for multidimensional data visualization. Information Visualization, 1:194–210.

Ware C (2004). Information Visualization. Morgan Kaufmann Publishers.

Wu S and Chow TWS (2004). Clustering of the self-organizing map using a clustering validity index based on inter-cluster and intra-cluster density. Pattern Recognition, 37:175–188.

Yang CC, Chen H, and Hong KK (1999). Visualization tools for self-organizing maps. In Proc. of the 4th ACM conf. on Digital libraries, pages 258–9.