

Visualization of Gene Combinations

Christian Tominski & Heidrun Schumann
Institute for Computer Science
University of Rostock
{ct,schumann}@informatik.uni-rostock.de

Abstract

Advances in the field of microarray technology have attracted a lot of attention in recent years. More and more biological experiments are conducted based on microarrays. The challenge researchers face today is to analyze and understand the collected data.

We present a visual approach to support understanding microarray data. In contrast to other visualization techniques, which represent expression of genes, we go one step further and make a switch to combinations of genes. Gene combinations bear more information, and hence, can lead to new hypotheses about the data. However, the increased amount of information imposes several challenges to an interactive visualization approach. We propose analytical and visual methods to deal with these challenges.

Keywords— Information Visualization, Visual Analysis, Microarray Data.

1 Introduction

In recent years, advances in bio-sciences have opened up new possibilities to investigate biological systems in high detail. New technologies allow researchers to collect large volumes of data. These data can lead to new insights and understanding of primitive organisms or larger biological systems. The ultimate goal is to derive models that mimic nature as close as possible. In order to achieve this ambitious goal, biologists have to analyze the data collected in their laboratories or generated by existing models. Besides sophisticated data analysis and mining methods, visualization has proven a helpful tool to assist the process of generating insight into new, not yet understood data. Particularly the interactive exploratory character of visual representations is advantageous when biologists seek to crystallize new hypotheses about their microarray data.

In this paper, we present a novel visualization approach to support the visual analysis of microarray data. Microarray data describe activity of genes (commonly known as gene expression or gene regulation) in different experiments or at different time steps. A variety of approaches

is known to visualize microarray data, let alone the many techniques required to make microarray data analyzable on computer systems. However, while other approaches aim to visualize genes as is, we take the analysis one step further and focus on the visualization of gene combinations. This step is motivated by the fact that combinations of a small number of genes are more likely to have an effect on certain processes in a cell than just a single gene might have. The approach we present here allows biologists to interactively explore and to visually compare different combinations of genes and to detect potentially interesting combinations for further investigation. Our approach manifests in an analysis pipeline that integrates analytical and visual components. Both analytical and visual methods have been designed to cope with the increased number of data items that we face when advancing from genes to gene combinations. In particular, we describe the following concepts in more detail:

- a pipeline for the visual analysis of gene combinations
- a flexible filter mechanism to pre-select potentially interesting gene combinations and neglect less relevant ones,
- a novel concept for visualizing gene combinations, including a novel color coding method, and
- visual enhancements to further improve the applicability and usefulness of our approach.

All concepts have been implemented and are available not only as the stand-alone application ViGeCo [18], but also as a plug-in for the microarray analysis framework Mayday [4].

In the following section we provide a brief overview on related work in the field. Section 3 will introduce the new approach to visualizing gene combinations. We will elaborate on our analysis pipeline, aspects of color coding, and additional on-demand visual clues. In the last section we will summarize our work and indicate directions for further research.

2 Related Work

Visual methods have always been used to support the analysis of microarray data. Classic techniques like line plots, scatter plots, or parallel coordinates are wide-spread. Even recent publications in microarray analysis make use of classic techniques, as for instance [15, 20]. Heatmap displays can also be considered a classic (e.g., [6, 12]). Heatmaps correspond to the common spreadsheet interpretation of data and use a red-black-green color scale for the visual encoding. They are easy to implement and yield relatively good results. A second advantage is their acceptance among biologists and medical experts.

However, when it comes to visualizing larger volumes of data or to emphasizing special aspect of the data, more sophisticated approaches offer better results. To handle larger data sets, analytical methods are mandatory. Clustering is a commonly used approach. As described in [17], self-organizing maps are a good choice to support visual interfaces for microarray data. Other approaches make use of principal component analysis [9]. Application scenarios that focus on the temporal aspects of microarray analysis need dedicated methods. A technique that supports not only the visual representation of temporal aspects, but also the interaction – direct manipulation in this case – with the time axis can be found in [3].

Many other visually driven approaches to microarray analysis have been described in the literature [14]. First efforts have been taken to develop general frameworks for combined analytical and visual analysis [7]. In many applications, gene expression is the focus of investigation. Analytical components abstract from expressions of single genes and lead to groups of similar genes. However, clustering methods usually involve complex computations, which are difficult to understand, let alone to parameterize.

In the next sections, we will see that switching from genes to gene combinations also leads to data abstraction, and that this abstraction can be described as a simple and easy to steer pipeline. To our best knowledge, we cannot tell of any other approach that uses the idea of gene combinations.

3 Visual Analysis of Gene Combinations

To lay the ground for our approach we introduce some basic notions. Microarray data contain information on the expression of genes $g \in G$ for several samples $s \in S$ (e.g., time steps or experiments). This can be described as a function $exp_G : G \times S \rightarrow \mathbb{R}$. Classic heatmap visualizations represent exactly that relationship between genes, samples, and corresponding expression. In this work, we focus on gene combinations. A gene combination is a subset $GC \subset G$. Expression of gene combinations is hence modeled as a function $exp_{GC} : \mathfrak{P}(G) \times S \rightarrow \mathbb{R}$, where

\mathfrak{P} denotes the powerset. Instantiations of exp_{GC} need to aggregate the expression of those genes participating in a gene combination. In other words, an aggregated expression value determines the level of regulation of an entire gene combination. Usually, we want to use the average, but it is also possible to consider other aggregates if needed for a particular application.

As indicated in the previous section, the visualization of combinations of genes has not yet been considered; there is no accepted approach to follow. Therefore, we analyzed the requirements to be fulfilled when switching from genes to gene combinations. Obviously, a gene combination contains more information than a single gene. This fact holds with respect to the plain data, but also with regard to the visual representation of gene combinations. Apparently, our approach has to cope with a larger volume of data. From the data perspective, analytical methods are required to analyze the raw data before visualization. On the other hand, the visualization has to take the increased number of data items into account; the visual representation should also communicate the fact that gene combinations bear further information inside. Since interactive exploration is an important goal, interaction techniques and performance issues are not to be left out without mention.

In conclusion, our approach has to face the following challenges: (a) In order to handle the volume of data to be visualized, we need data analysis; (b) In order to account for the higher level of information encapsulated in gene combinations, we need a dedicated visualization solution; (c) In order to allow for easy visual exploration, we need a high degree of interactivity. Considering these requirements, we have developed an approach whose basic idea can be best described as a pipeline.

3.1 Analysis Pipeline

The main objective of the analysis pipeline is to condense the data under investigation considerably before the visual mapping. Only then can we avoid presenting indigestible large pieces of information to the analyst. Moreover, proper data analysis helps to achieve interactive frame rates during data exploration. The analysis pipeline (see Figure 1) implements the following main tasks:

- Filtering of genes
- Generation of combinations from relevant genes
- Filtering of gene combinations
- Interactive visualization of relevant gene combinations.

The first step of the pipeline performs an analysis of the microarray data at hand. The goal of this step is to find potentially relevant genes and to eliminate less relevant ones.

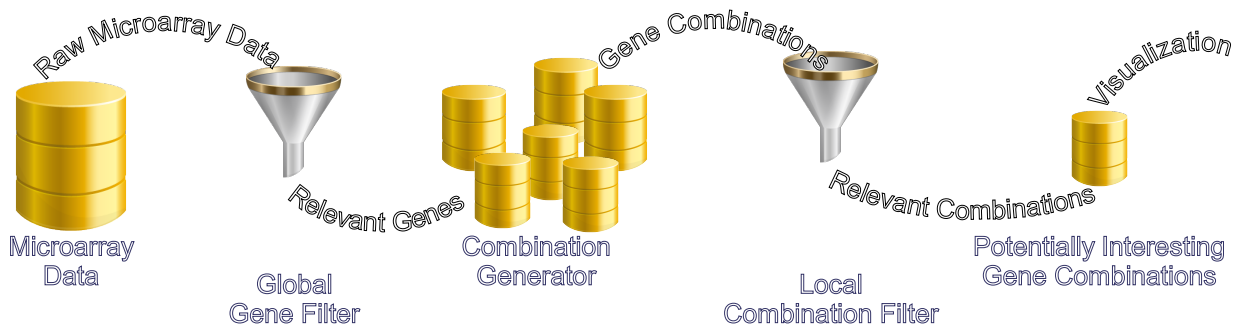


Figure 1: Analysis pipeline.

The genes that have passed the first step are used to generate gene combinations in the second step. However, creating all possible combinations of genes would lead to an exponential increase in the number of data items. To cope with this problem, the generation process is tightly intertwined with step three – the filtering of gene combination. The outcome of step three is a set of relevant gene combinations. Only these relevant data items are handed over to the visualization step. The visualization as well as the data analysis can be steered interactively by the user.

In the following sections we provide detailed discussion of the filtering processes involved, the visual means used, and the interaction mechanisms available.

3.2 Filtering Genes and Combinations Flexibly

The filtering that occurs in the analysis pipeline determines which gene combinations are brought to display, and hence, which can possibly be identified as interesting ones by the analyst. Self-evidently, the filtering is crucial. However, the question, what makes a good or at least an adequate filter is hard to answer. Therefore, we opted for a flexible two-tier filtering mechanism that consists of:

- a global gene filter and
- a local gene combination filter.

The task of the first filter is to extract interesting genes. This filter can operate globally on the microarray data. By globally we mean that the filter has access to the whole data set. Commonly, a gene is considered interesting if it exhibits high expression – positive or negative. Hence, our implementation of the filter blocks genes with low expression. That is, a gene $g \in G$ is blocked if $\forall s \in S : |exp_G(g, s)| < thres_{exp}$, where $thres_{exp}$ is the filter threshold. The result of the gene filter is a set of relevant genes. These are potential candidates to form gene combinations with other genes.

Generating gene combinations is a serial process that synthesizes a combination and immediately checks it

against the second filter level, the local gene combination filter. Here, the word local indicates that the filter does not have access to all possible gene combinations, but only to the one just created by the generator. In that sense, the gene combination filter is semantically restricted compared to the gene filter. However, since the set of all possible gene combinations exponentially large, it is impossible to act globally. Even access to a subset of all possible gene combinations is difficult to realize due to the enormous memory requirements. The restriction to local information is the reason why it is not so easy to tell what a good filter for gene combinations is. With the feedback biologists gave us, we achieved quite good results with an implementation that considers both the regulation of genes participating in a gene combination and the variance of regulation within a gene combination. This is motivated by the fact that biologists are interested in gene combinations that contain either relatively similar genes or rather contrasting genes. We use a function $sim : \mathfrak{P}(G) \rightarrow \mathbb{R}$ that determines dis/similarity within a gene combination, normalized to the interval $[-1, 1]$. Internally, sim summarizes and normalizes the pairwise dis/similarity for all genes in a gene combination. Negative values for sim indicate dissimilarity, positive values stand for similarity. The gene combination filter blocks a gene combination $GC \subset G$ if $thres_{dis} < sim(GC) < thres_{sim}$, where $thres_{dis}$ and $thres_{sim}$ are thresholds for dissimilarity and similarity, respectively.

We have to mention that in some cases the described basic filters alone might not be sufficient. Biologists certainly have special requirements for different kinds of experiments, which may raise the need for different semantics of the filters [5]. Therefore, the filter mechanism has been designed to be modular, and hence, easily extensible. This is the key to flexibility of the analysis pipeline. The only restriction that applies is that gene combinations can only be filtered with respect to local information, but not in comparison with other gene combinations. This is an aspect to investigate further in future work.

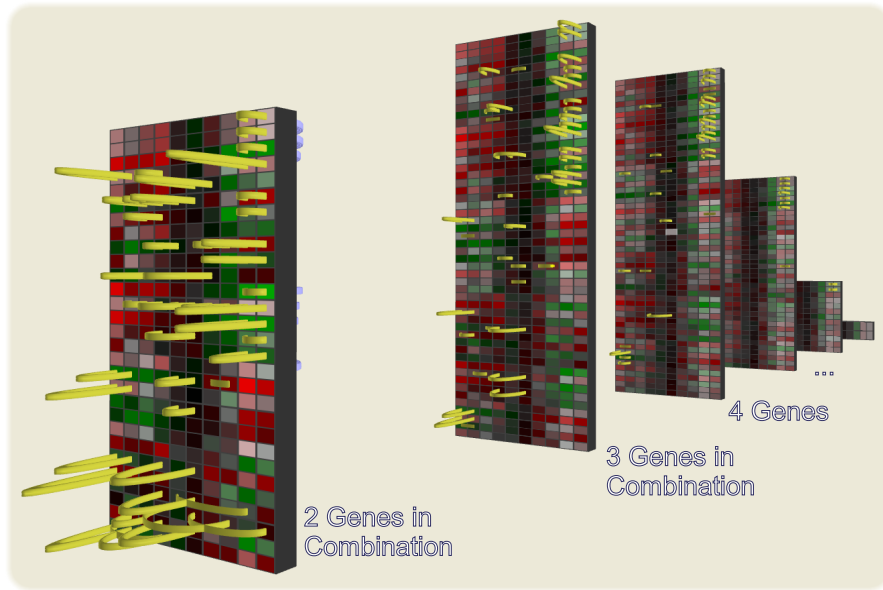


Figure 2: Gene combinations of different size are visualized on different panels in 3D.

3.3 Representing Gene Combinations Visually

The potentially interesting gene combinations extracted from the data are input to the visualization process. We designed the visual representation in close collaboration with biologists, who are used to interpret heatmaps. The challenge was to extend what biologists are familiar with to fit the requirements of gene combinations. In order to account for the increased amount of information to be presented, we made a step from 2D visualization (e.g., classic heatmaps) to 3D visualization. This gives us one additional dimension for the visual mapping process.

The basic layout of the visualization consists of multiple panels floating in 3D. Each panel consists of a matrix of boxes. Each box represents the expression $exp_{GC}(GC, s)$ of a gene combination $GC \subset G$ with respect to a sample $s \in S$. Each panel represents the expression of gene com-

binations of a different size: The first panel shows all combinations of size two $|GC| = 2$, the second panel shows all combinations of three genes $|GC| = 3$, and so forth. Optionally, an additional panel visualizes the expression exp_G of the original genes, but only of those having passed the gene filter (i.e., genes that can occur in gene combinations). Figure 2 illustrates the basic layout. Gene combinations are mapped to the vertical axis of a panel, whereas samples are mapped to the horizontal axis. The expression is mapped to color. Since color coding usually requires careful thinking [2, 16], let us take a closer look.

Visual representations in bio-science contexts often use a color scale from red, indicating high positive expression, to black, no expression, to green, high negative expression (see Figure 3(a)). In our case, however, the expression of a gene combination is an aggregated value. This can lead

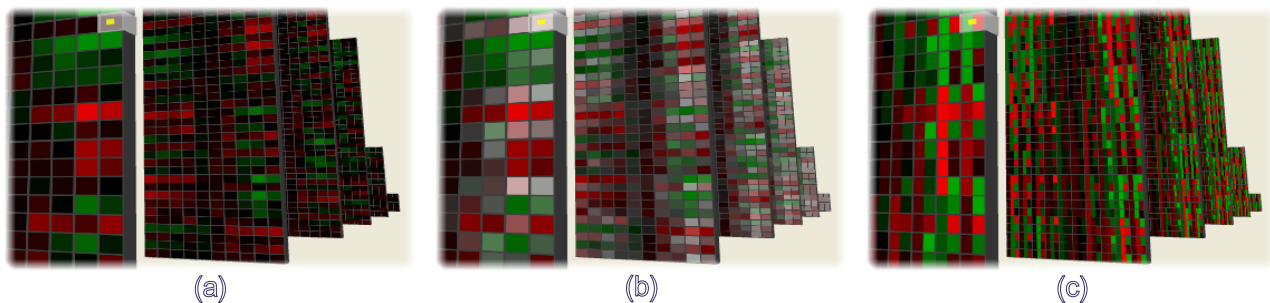


Figure 3: Different visual encodings. (a) Classic encoding; (b) Variance encoding; (c) Small multiples.

to situations where, for instance, two highly negatively expressed genes compensate two highly positively expressed genes. The visual result would be a blackish color, which does not really reflect the actual activity within the combination of the four genes. To address this concern, we developed a color scale that accounts for the variance within a gene combination (see Figure 3(b)). As a visual aid for the analyst we encode the variance to the saturation channel of a color, in addition to the classic mapping of expression to the color’s brightness (or value).

More precisely, for a given gene combination $GC = \{g_1, \dots, g_n\}$ and a given sample $s \in S$ the hue-saturation-value color components are computed as follows:

$$\begin{aligned}
 sum &= \sum_{i=1}^n exp_G(g_i, s) \\
 abs &= \sum_{i=1}^n |exp_G(g_i, s)| \\
 avg &= \frac{sum}{n} \\
 dev &= \frac{abs - |sum|}{n} \\
 hue &= (avg > 0) ? \text{red} : \text{green} \\
 saturation &= 1 - \frac{dev}{dev + |avg|} \\
 value &= dev + |avg|
 \end{aligned}$$

Note that we assume dev , $saturation$, and $value$ to be in the interval $[0, 1]$, and avg to be in $[-1, 1]$.

Although the variance encoding allows for better insight into gene combinations, biologist had the desire to see details. We chose small multiples (also see [1]) to provide for the sought details: Upon request, panel cells can be subdivided to accommodate subcells for each gene participating in a gene combination. The subcells can then be color coded with the classic red-black-green color scale (see Figure 3(c)). Enabling users to see detail has already been mentioned as an advantage of small multiples. However, with increasing numbers of genes within a combination, the subcells get thinner and thinner, which makes them harder to distinguish. Therefore, we suggest using the variance encoding for an overview, and the small multiple encoding to see details on demand only.

The basic display described in this section is capable of representing several hundreds of gene combinations. The novel variance encoding allows biologist to spot interesting combinations of genes even in cases where aggregation might have extinguished high, but contrary expression values. The small multiple display provides for the necessary detail and facilitates comparison of gene combinations.

3.4 Comparing Gene Combinations Effortlessly

Besides locating genes with high expression or finding new interesting gene combinations, visual comparison is one of the most important tasks biologists pursue with visualization. Even though the colors of cells, and hence, the expression of gene combinations can be compared visually, the cognitive efforts involved are not negligible. Visual aids are required to facilitate comparison of gene combinations. Such aids have to support easy detection of similarities. But also extreme contrasts are of interests to biologists. To support both tasks, we provide what we call similarity arcs and dissimilarity arcs.

Arcs have proved useful in many visualization approaches (e.g., [19] or [13]). They are normally used to visually link entities that are associated with each other. In our case, we consider panel cells the entities, and similarity and dissimilarity the semantics for association. Since our approach uses a 3D presentation space, arcs are relatively straight-forward to embed. We went for similarity arcs attached to the front plane of a 3D panel and dissimilarity arcs to its back. To make both types of arcs easy to distinguish, they are shown in different colors: yellowish for similarity arcs and blueish for dissimilarity arcs (see Figures 4(a) and 2). To emphasize highly dis/similar data items, the strength of dis/similarity is encoded to the height of the arcs. This is different to other arc-based approaches, where arc height usually depends on the distance of the entities linked. To avoid cluttering the display with large numbers of weak dis/similarity arcs, we again apply a filter mechanism that allows analysts to set separate thresholds for similarity and dissimilarity interactively.

Until now, the arc display connects samples for which a gene combination shows dis/similar behavior (horizontal arcs). This is particularly useful when samples are associated with different time steps of an experiment and a visual comparison is conducted with respect to time. However, arcs could also be used to connect dis/similar gene combinations with regard to the same sample (vertical arcs). It could even make sense to connect dis/similar expression values regardless of particular gene combinations or samples (arbitrary arcs). The later possibilities can be implemented easily once biologists see a need for comparing in these dimensions.

3.5 Exploring Gene Combinations Interactively

Interaction is an important aspect of visualization. Visual exploration of unknown data is hardly possible without carefully designed interaction. Interaction facilities have to support analysts in accomplishing the task at hand, which includes selecting alternative views on the data, selecting appropriate filter thresholds, or selecting visual aids as necessary.

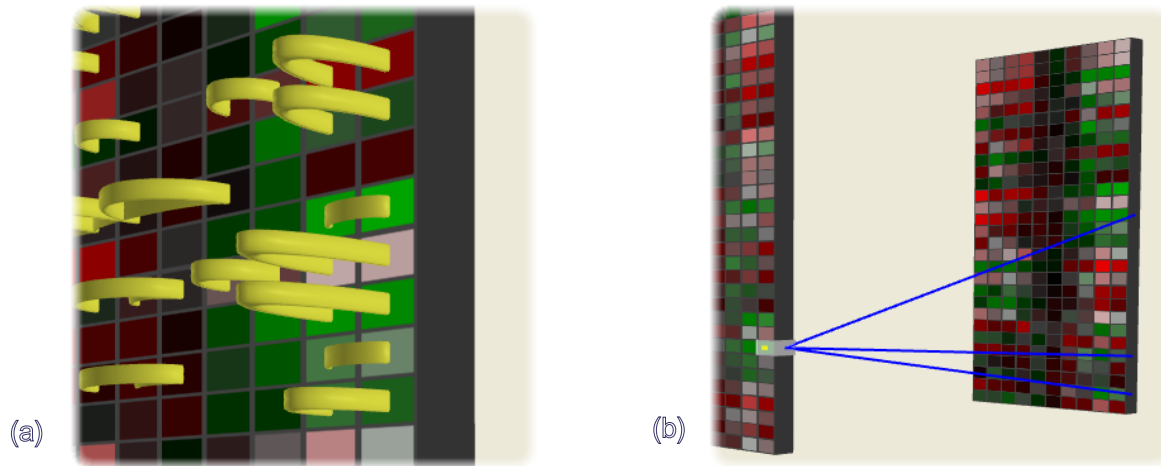


Figure 4: Visual enhancements. (a) Arc display; (b) Gene containment links.

Our approach integrates several interactive features, which we will describe in this section. The first and most obviously required interaction is navigation in 3D. This is particularly important, since only if navigation is effortless in 3D will biologists accept the switch from classic 2D heatmaps to our 3D display. Virtual trackballs are excellent tools for 3D navigation [8]. They allow for easy rotation of panels in space. We found it helpful to provide two modi of navigation: rotation of all panels at once or rotation of a selected panel. In order to adjust the center for rotation, users can select any cell of any panel; an animated transition smoothly moves the virtual camera accordingly. Zooming is available as well.

The selection of a panel cell not only moves the rotation center, but also serves as selection for interactive labeling. We provide labeling of gene combinations and samples per panel. A complementary 2D heatmap labels the expression of genes in the currently chosen gene combination. This way, users can see which genes are contained in which combination. However, if the analyst wants to extend the analysis to combinations that contain the same genes as the currently investigated one, labeling alone is not sufficient to find such combinations.

To help users recognize gene combinations that share a common set of genes, we provide a further extension called gene containment links. One can imagine gene containment links as a kind of similarity arcs with the difference that similarity is not related to the expression value, but to the content of gene combinations. Gene containment links emanate from cells of a panel (let it be the i -th panel) and connect to those cells of the next ($i + 1$ -th) panel that contain the same genes (and of course one further gene). The links can be shown for all cells or for the currently selected cell only (as shown in Figure 4(b)). Successively selecting

cells that are connected via gene containment links can be helpful to gain structural insight.

All extensions described in this section have on-demand character, that is, users are free to use them depending on the task at hand.

4 Conclusions and Future Work

In this work, we presented a novel approach to visualizing microarray data. Switching from genes to gene combinations marks an important innovation of our approach. The involved challenge of dealing with larger volumes of data has been tackled via an integration of analytical, visual, and interaction methods. The analytical part has been designed as a filter-based analysis pipeline. The visualization is based on heatmap-like panels arranged in a 3D presentation space. The presented variance-aware color coding is not only useful in the case of gene combinations, but can also be applied to other visualization approaches that have to deal with aggregated data values. Several interactive on-demand enhancement, including dis/similarity arcs and gene containment links, further assist users during the exploration of microarray data.

The concept of visualizing gene combinations has been implemented in close collaboration with neurobiologists. Noteworthy about the interactive analysis tool ViGeCo [18] is the fact that the implementation is available as a plug-in for the microarray analysis framework Mayday [4]. The advantage of the plug-in implementation is the potential integration of analytical methods of Mayday and visual methods of ViGeCo. Furthermore, Mayday is capable of handling multiple views, which is useful to provide different views on the data. Linking mechanisms help to coordinate interaction among different views.

Further adjusting the visual interface of ViGeCo to the needs of biologists is only one task we face in future work. One aspect in this regard is sorting of panel rows in order to assign similar gene combinations to rows that are close to each other. This could be realized by applying SOM as described in [11]. Another issue to be solved in the future is the automatic derivation of good filter settings depending on characteristics of the data. As tests have shown, leaving novice users alone with the task to set filter thresholds interactively is impractical and hinders user acceptance. These and other tasks are to be accomplished within the scope of a graduate school with special focus on simulation and modeling of regenerative systems (see <http://wwwmosi.informatik.uni-rostock.de/diemosiris>).

Acknowledgements

We gratefully acknowledge the help of Clemens Holzhüter in implementing the ViGeCo tool [10]. We thank Christian Lange and the cooperation partners from the neurobiology department for valuable discussion and for providing test data.

References

- [1] J. Bertin. *Semiology of Graphics: Diagrams, Networks, Maps*. University of Wisconsin Press, 1983.
- [2] D. Borland and R. Taylor. Rainbow Color Map (Still) Considered Harmful. *Computer Graphics and Applications*, 27(2), 2007.
- [3] P. Craig, J. Kennedy, and A. Cumming. Towards Visualising Temporal Features in Large Scale Microarray Time-Series Data. In *Proc. of International Conference Information Visualisation (IV)*, 2002.
- [4] J. Dietzsch, N. Gehlenborg, and K. Nieselt. Mayday - a Microarray Data Analysis Workbench. *Bioinformatics*, 22(8), 2006.
- [5] S. Drăghici. *Data Analysis Tools for DNA Microarrays*. Chapman & Hall/CRC, 2003.
- [6] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proc. of the National Academy of Sciences of the United States of America*, 95(25), 1998.
- [7] N. Gehlenborg, J. Dietzsch, and K. Nieselt. A Framework for Visualization of Microarray Data and Integrated Meta Information. *Information Visualization*, 4(3), 2005.
- [8] K. Henriksen, J. Sporning, and K. Hornbaek. Virtual Trackballs Revisited. *IEEE Transactions on Visualization and Computer Graphics*, 10(2), 2004.
- [9] M. A. Hibbs, N. C. Dirksen, K. Li, and O. G. Troyanskaya. Visualization methods for statistical analysis of microarray clusters. *BMC Bioinformatics*, 6, 2005.
- [10] C. Holzhüter. Visuelle Techniken zur Analyse von Microarraydaten. Master's thesis, University of Rostock, 2006. (in German).
- [11] M. John, C. Tominski, and H. Schumann. Visual and Analytical Extensions for the Table Lens. In *Proc. of Visualization and Data Analysis (VDA)*, 2008.
- [12] R. Kincaid. VistaClara: an interactive visualization for exploratory analysis of DNA microarrays. In *Proc. of the ACM Symposium on Applied Computing (SAC)*, 2004.
- [13] P. Neumann, S. Schlechtweg, and S. Carpendale. ArcTrees: Visualizing Relations in Hierarchical Data. In *Proc. of Joint Eurographics - IEEE VGTC Symposium on Visualization (EuroVis)*, 2005.
- [14] P. Saraiya, C. North, and K. Duca. An Evaluation of Microarray Visualization Tools for Biological Insight. In *Proc. of IEEE Symposium on Information Visualization*, 2004.
- [15] N. Shah, V. Filkov, B. Hamann, and K. I. Joy. GeneBox: Interactive Visualization of Microarray Data Sets. In *Proc. of the International Conference on Mathematics and Engineering Techniques in Medicine and Biological Sciences (METMBS)*, 2003.
- [16] S. Silva, J. Madeira, and B. S. Santos. There is More to Color Scales than Meets the Eye: A Review on the Use of Color in Visualization. In *Proc. of International Conference Information Visualisation (IV)*, 2007.
- [17] M. Sultan, D. Wigle, C. Cumbaa, M. Maziarz, J. Glasgow, M. Tsao, and I. Jurisica. Binary tree-structured vector quantization approach to clustering and visualizing microarray data. *Bioinformatics*, 18(Suppl.1):S111–119, 2002.
- [18] C. Tominski, C. Holzhüter, A. Unger, and H. Schumann. Interactive Poster: Visualization of Gene Combinations. In *Poster Compendium IEEE Conference on Information Visualization (InfoVis)*, 2007.
- [19] M. Wattenberg. Arc Diagrams: Visualizing Structure in Strings. In *Proc. of IEEE Symposium on Information Visualization (InfoVis)*, 2002.
- [20] K. Yano, K. Imai, A. Shimizu, and T. Hanashita. A new method for gene discovery in large-scale microarray data. *Nucleic Acids Research*, 34(5), 2006.