Connecting Genes with Diseases

Heimo Müller[1], Robert Reihs[1], Stefan Sauer[1], Kurt Zatloukal[1],
Marc Streit[2], Alexander Lex[2], Bernhard Schlegl[2], Dieter Schmalstieg[2]
(1) Medical University of Graz, (2) Graz University of Technology

Abstract

We present a visual data mining application using the combination of clinical data, pathways and biomolecular data. Using pathways to navigate and filter the clinical and molecular data allows a more systematic and efficient investigation of problems in modern life science. A multiplicity of hypotheses can be evaluated in the same period of time, enabling a much better exploitation of the data. We present a system for data preprocessing and automatic classification, a set of visualization views and finally the integration of the views in the Caleydo visualization framework, which enables the "coupling" of molecular and a broad spectrum of clinical data. With the help of the Caleydo framework the medical expert can identify connections between genetic parameters, patient subgroups, and drug responses in an intuitive way.

1. Introduction

Today masses of data are being produced in science and engineering applications, promising new insights. But how can an expert find meanings in terabytes of data? To successfully search for new hypotheses in large datasets, we must find unexpected patterns and interpret evidence in ways that frame new questions and suggest further explorations. Visual analytics methods will help us to

- overview large datasets, as the human visual sense is optimized for parallel processing,
- connect the global view with detail information, e.g. the selection of a single gene can modify all views,
- provide different contextual views depending on users' needs and experience level,
- deal with inhomogeneous data sets and a broad range of data quality.

In order to achieve these goals, we developed a set of methods for data preprocessing, visualization and interaction (see Figure 1). With the ability to integrate a broad range of medical data, to filter either by clinical parameters or by gene expression data (i.e. molecular data), dynamically reload pathways in which a gene plays a role, and to synchronize different visualizations through linked views, an expert can, in the truest sense of the word, travel through the data space.

2. Related Work

The essential but to date unsolved problem in the emerging field of personalized medicine is the question of how to identify connections between genetic variants and their corresponding diseases or the response to certain drugs and treatments, respectively. It is therefore necessary, for example to connect gene data and clinical data in order to categorize specific subgroups of patients with certain diseases. The huge amount of data provided by molecular analytical methods (genetic polymorphisms, gene expression data, proteomics) can only be accomplished by applying bioinformatical and statistical methods. However, standard methods of statistics and bioinformatics fail when the data is inhomogeneous – as is the case with clinical data – and when data structures are obscured by noise and dominant patterns.

This has led to a stronger demand on data visualization, which addresses the problem of the very large datasets and the particularities of medical data analysis. We build our work on visual data exploration methods of large datasets, especially hierarchical data structures as described by Hege et al. [1], Keim and Kirgel [2,3], Grinstein and Meneses [4] and Fekete and Plaisant [5]. Related work on the integrated visualization of clinical and health record data was done by Jiye et al. [6], Chittaro [7], Ganslandt [8], and Aigner and Miksch [9]. Further related work can be found in the description of the single processing and visualization steps.

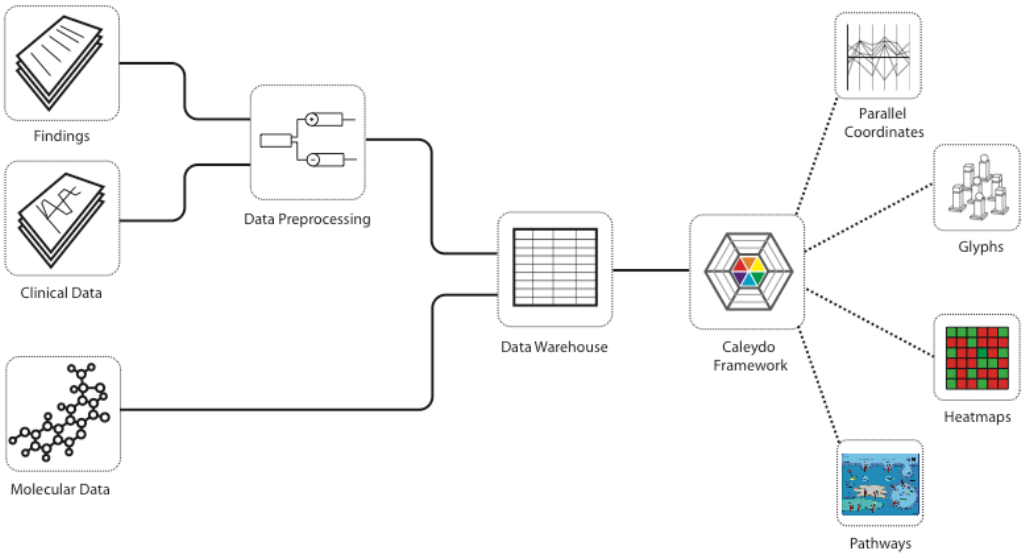

Figure 1 - Overall architecture

3. Data Preprocessing

Many hospitals and medical universities have a large medical data pool, which contains information of great relevance for biomedical research. In order to utilize the knowledge of these assets, it is necessary to search data of medical records in a structured way.

The starting point of our undertaking is the tissue collection of the Institute of Pathology in Graz, which contains approx. 2.9 million samples from 800.000 patients representing a non-selected patient group characteristic for Central Europe, which is now the core of the BioBank of the Medical University of Graz [10] and part of the Central Research Infrastructure for Molecular Pathology (CRIP) [11]. The scientific value of the tissue collection is not only characterized by its size and its technical homogeneity (all samples have been processed in one institute under constant conditions for more than 20 years), but also by its population-based character. These features provide ideal opportunities for epidemiological studies and allow the validation of biomarkers for the identification of specific diseases and the response to treatment regimes.

Each tissue sample is linked to a histopathological diagnosis and additional medical data such as staging and grading of tumors as well as information on patient survival. Furthermore, for some samples whole genome gene expression data is available. Medical data is given as free text in German language. While working with this data we realized that it is very difficult to extract the information for the visualization pipeline, with an on-demand search in the plain text findings. The alteration of terms in past years, the change of classification, misspellings in the texts and different description of clinical findings pose the main challenges in this area.

In order to extract well-structured medical information from plain text findings, we use a simple rule-based text mining approach. However, before the text mining step, all findings, which belong to the same patient and disease, need to be merged. The results build the starting point for the classification process and are stored in a relational database for back referencing.

Our classification is based on ICD-10 (International Classification of Diseases 10th revision) [12] and ICD-O-3 (International Classification of Diseases for Oncology 3rd revision) [13], but also other classification systems can be supported. In a separate module tumor stagings and organ receptors states are extracted. The information is then stored in a relational database in a well-structured and searchable way.

The data preprocessing can be divided into three steps, as depicted in Figure 2:

(A) Data Import and Merge

In the first step we import the unstructured data from an existing system (for example an Access Database, XML data, *.csv, *.txt or from other databases like Oracle or PostgreSQL). The extracted text blocks are then inserted into a single relational database.

(B) Cleanup

In the second step a text cleanup module corrects misspellings and replaces abbreviations. Additionally, when available, some separate fields are processed with dictionary ID entries, to stay abreast for the relational database design. For example, the physician's name is not stored in a plain text field for every case he was involved but only with an ID, which then points on a table with all physicians. For this mapping of text fields to dictionary fields we use spell checks to find all different (mis)spellings of the physicians. We also provide a special system for the title of the physicians because in the course of time these can change. The text cleanup runs through the database when a new mistake or abbreviation is found in the text blocks.

(C) Classification

The core part of the system is the classification module. In the first step we merge certain words to terms. For example such a term in our represented pathological data is "Metastase eines" (metastasis of a) which means that the following tumor is not the tumor itself – it is a metastasis of the tumor. In this step the text is also split into single terms, left and right neighbors, sentences and findings.

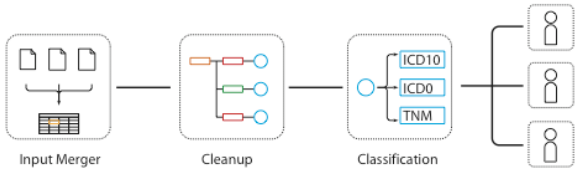

Figure 2 – Data preprocessing steps

In our text mining approach we use a classification tree-based system. Every node of the classification tree describes the matching word by a regular expression pattern for different spellings and a set of processing rules. These rules contain flags about the valid position of the term ("foreword", "ending", "negation", "in sentence", "in finding", "in the whole case").

In Figure 3 a single node of the decision tree can be seen. The node shows the rule for the synonym "Neuroendokrin". The color of the node depicts the node type: root node (orange), rule node (green), negation node (red). The pattern is a regular expression to match different spellings of the synonym. The numbers on the left and right side indicate how many words have to be between two matches.

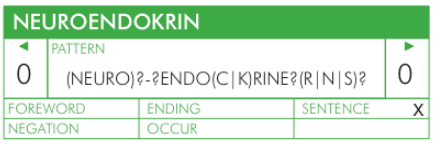

Figure 3 – Node of the decision tree

Currently we use a set of 104 classification trees with an overall number of 2670 nodes. The classification trees were created by medical specialists at the Institute of Pathology in Graz during the last 2 years and are currently in the evaluation phase at several other pathology institutes.

Figure 4 shows the complete tree for the classification of ICD-10 codes and ICD-O codes related to mamma carcinoma. The decision tree consists of the start node (orange), rule nodes (green) and negation nodes (red). The resulting classification can be either ICD-10 (blue circle) or ICD-O (violet circle).

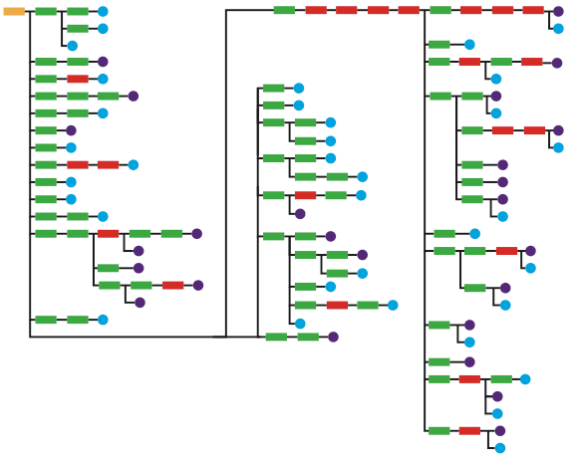

Figure 4 – Structure of the decision tree for the classification of mamma carcinoma

The evaluation of the data preprocessing and automatic classification shows that we have a recall of 86,1% and a precision of 83,9% for the ICD-10 codes. For the ICD-O codes we have a precision of 90,3%, and a recall of 93,1%.

4. Visualizations

The Caleydo framework (www.caleydo.org) [14], developed at the Graz University of Technology, describes clinical/patient data, gene expression data and pathway graphs. The software suite is written in Java and implements state-of-the-art visualization techniques such as multiple coordinated views, linking & brushing and details on demand. The rendering of 2D and 3D views uses the Java OpenGL (JOGL) library, a Java binding that provides access to the OpenGL 2.0 specification. The building blocks of the presented solution are a set of synchronized views:

- Multilevel Data Glyphs
- Parallel Coordinates
- Hierarchical Heat Map
- Pathway Graphs

4.1 Multilevel Data Glyphs

Ropinski and Preim [15] investigate glyph-based visualization techniques in medical visualization. They build a glyph taxonomy based on the way information is processed when interpreted and propose guidelines for the usage of glyphs. Ward gives a general introduction to multivariate glyphs [16] and describes taxonomy of glyph placement strategies. He distinguishes between data-driven and structure-driven approaches and introduces strategies to avoid the overlapping problem and a novel space-filling layout of hierarchically structured data. We developed data glyphs [17] described by:

- a set of graphical primitives, organized into level of detail combined with a description of the visual capabilities of each graphical primitive,
- mapping of data variables to graphical primitives,
- rendering algorithms for each level and
- spatial positioning algorithms.

Data glyphs are modeled as 3D objects. This allows a high information density - at the highest level of detail a data glyph visualizes up to 15 variables. However, problems introduced with this approach are occlusion, perspective distortion, complex navigation and orientation in 3D space for inexperienced users.

To avoid these problems we have restricted the 3D space to an isometric view, where only the 2D position of glyphs can be altered. An isometric view is known to users from technical illustrations and from the early years of computer and video games. In an isometric projection data glyphs can be compared independently of their spatial position and no perspective distortion is applied. Furthermore several performance optimization strategies, e.g. bitmap caching, can be applied because of the restricted 3D projection. Our glyph designs also ensure, that all geometric primitives are visible in the isometric projection.

In order to achieve well-graded and consistent levels of details for data glyphs, we use the semantic zoom approach and rely on the principle that the dominant visual variable of level n is also the strongest visual variable in level n+1. We use three levels of detail for a single glyph:

A.) Primary level, the pixel view
In the primary level one data variable determines the color of the glyph. This color is also the dominant color in all higher levels. A glyph is rendered in the pixel view when its screen size is very small, e.g. below 4x4 pixels. By using data glyphs in the pixel view level we can interact with several millions of elements at a time.

B.) Secondary level, the iconic view
In the secondary level we add 4 additional data variables. A glyph is rendered in the iconic view when its screen size is approx. between 4 by 4 and 64 by 64 pixel. By using data glyphs in the iconic view we can interact with several thousands of elements at a time.

C.) Tertiary level, the detail view
In the tertiary level we add approximately 10 geometric primitives to the data glyph, which results in an overall number of no more than 15 data variables mapped to a single glyph. A glyph is rendered in the detail view when its size is greater than 64x64 pixels. By using data glyphs in the detail view we can compare several hundreds of glyphs.

Figures 5 shows three glyph types: Cubic glyphs are well suited to visualize multivariate data sets with a broad range of data types, crystal glyphs are optimized for more homogenous data sets and human glyphs are well suited for the visualization of person related data, e.g. the localization of metastases or laboratory values.

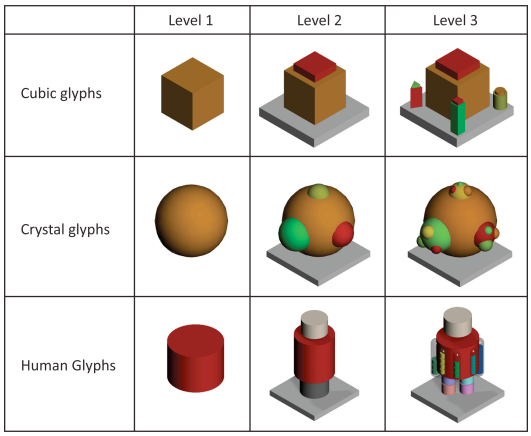

Figure 5 – Multillevel data glyphs

Figure 6 shows data glyphs arranged in a scatterplot. At the bottom of the window, the value distribution is shown for each attribute (i.e. staging attributes, sex and disease free survival) as an interactive histogram. With the help of the histogram the user can select subgroups for further visualization steps.

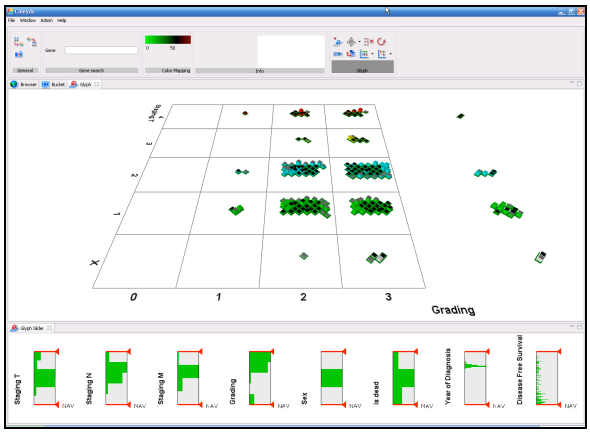

Figure 6 – Level 2 glyphs arranged in a scatterplot by T-Staging and Grading

4.2 Parallel Coordinates

We use parallel coordinates [18] to visualize clinical data as well as gene expression data. Our implementation of parallel coordinates uses one-dimensional brushes as well as angular brushes [19] to select a subset of the data. In gene expression analysis a common task is to remove all genes that are neither up nor down regulated for all experimental conditions. Therefore, we implemented a global brush, visible on the left in Figure 7, which removes all genes that never leave the region the blue bar is spanning.

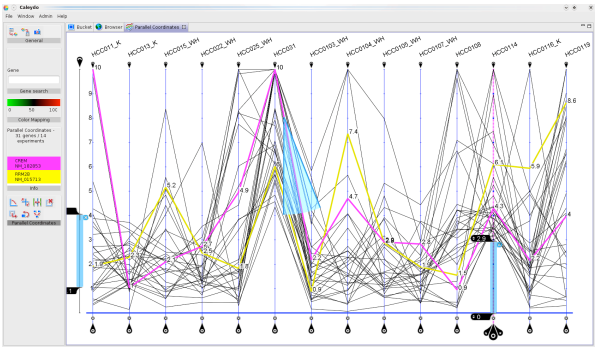

Figure 7 - Parallel coordinates showing 14 experiments as axis and genes as polylines. The blue brushes (1D and angular) allow filtering of the dataset. The leftmost brush filters out all elements that never leave the spanned region.

Furthermore, our parallel coordinates implementation allows to switch between polylines and axes at runtime. For gene expression analysis this means that in one mode the genes are the polylines, while in the other the experiments are (and obviously vice versa for the axes). The only limitation is the number of axes – which should not be more than about 50 for a meaningful analysis. This allows the analysis of a limited number of genes for a large number of experiments and furthermore immediately identifies experiments which run against a trend visible in others.

Our implementation can visualize up to 5000 polylines interactively on a Notebook with an Intel Core2 Duo CPU with 2 GHz and a NVIDIA Quadro NVS 140M with 128 MB VRAM.

We use a random sampling approach to allow the exploration of much larger data sets. Thereby, the sampling only affects the visualization – all operations are always executed on the whole data set. If random sampling is used, the system always displays a predefined number of lines. If the number of lines to be visualized drops below the threshold due to filtering, the visualization renders every line. This approach allows us to give users a representative overview of large datasets while still allowing manipulation on individual elements when filtering is used.

4.3 Hierarchical Heat Map

Heat maps are a common way of visualizing gene expression data [20]. The genes are arranged in rows and the experiments in axes. The color encodes the regulation of the gene. Due to the large number of genes it is not feasible to visualize all values simultaneously on a traditional computer screen.

Therefore, we have implemented a hierarchical approach [21].

The hierarchy consists of three levels. On the leftmost side an overview of all genes is shown, which helps to localize the current position in the dataset. The next level shows a selection of about 500 genes. In this view individual elements can already be recognized. In the detailed view on the right a set of 10-80 genes are shown, with labelling for each individual gene.

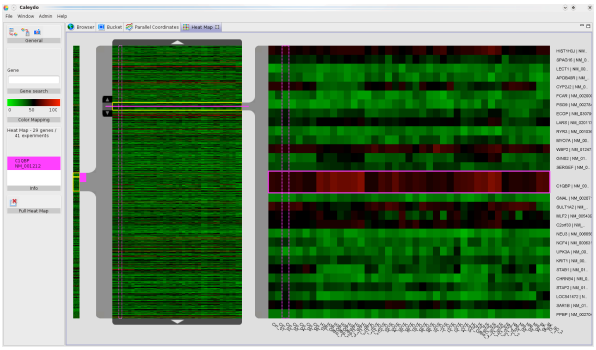

Figure 8 - Hierarchical heat map. The heat map shows a total of 41 experiments and 4634 genes. The three layers from left to the right provide different levels of detail while preserving the context.

Selections are highlighted in all three levels. Multiple selections that occur outside of the detailed views are highlighted in the overview levels thus permitting to rapidly switch to the equivalent entity. We are currently working on integrating clustering into the heat map, which will significantly increase the meaning of the localization in the different levels.

4.4 Pathway Graphs

Pathways are models of cellular functions represented by graphs. Nodes in pathway graphs are enzymes/genes/proteins (depending on the biological level) and chemical compounds. The edges are signals or chemical reactions on the cellular level. Genes occur in different pathways and therefore perform various roles depending on its biological context. We integrate approx. 700 pathways from two major public databases: KEGG [22] and BioCarta [23]. Figure 9 shows an exemplary KEGG pathway.

When a particular gene in a pathway is selected the corresponding row in the heat map as well as the polyline in the parallel coordinates plot is highlighted. Vice versa, nodes in the pathways are highlighted upon selections performed in connected views.

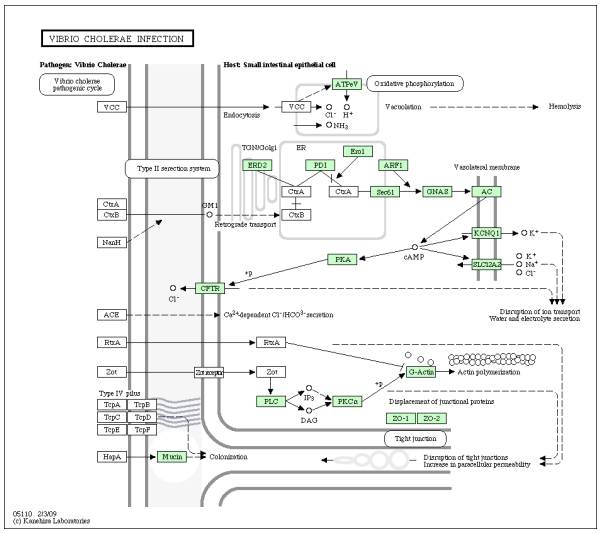

Figure 9 – Sample KEGG pathway graph "Vibrio Cholerae Infection".

5. Connecting Genes with Diseases

Due to the comprehensive collaboration of experts from various fields (pathologists, geneticists, molecular biologists and oncologists) within the project, we are in the fortunate position to have access to datasets that contain contributions from all domains. The following figures are based on a dataset consisting of 180 patients (experiments), each consisting of various clinical parameters, like sex, age, disease free survival and other personal information. It also includes disease related information like tumor staging and medication. Furthermore, for each patient the full gene expression data (37,632 regulation values) is available.

Different aspects of the data are depicted in separate coordinated visualizations. In addition to this classic multiple view approach, Caleydo supports an arbitrary placement of 2D views in a 3D scene and enriches the highlighting of selected entities (genes, patients) by using visual links [14,24].

An integral part of the Caleydo framework is the so-called Bucket which allows the management of up to 20 related views (see Figure 10). The setup consists of a view in the center and four contextual views forming the side walls of the bucket. Related views which are not of immediate interest are placed as thumbnails on the rim. In addition the framework provides zoom features as well as drag and drop support for views.

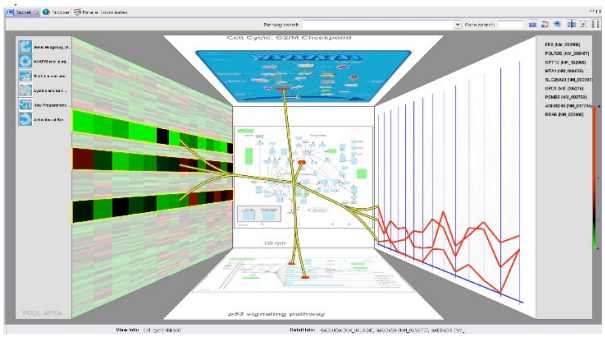

Figure 10 – The Bucket is a concept that arranges related views in a 2.5D scene. Identity relations of elements in different views are connected by visual links (yellow).

Figure 11 shows the integration of a glyph view, a heat map, parallel coordinates and several pathways inside the Bucket.

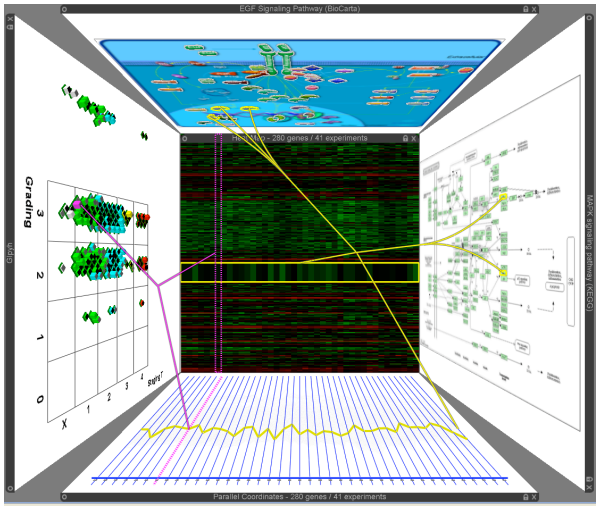

Figure 11 - Linking clinical and gene expression data inside the Bucket.

This way genetic data is mixed with clinical data in one scene. Colored visual linking trees are connecting data entities from the same data space. The mutual basis of these data spaces are the heat map and parallel coordinates. For example a patient is represented by a glyph and visually connected to the axis of the parallel coordinates as well as to the column of the heat map (pink connection tree). In turn, a gene, represented by a polyline in the parallel coordinates and by a row in the heat map, is linked to pathway nodes depicting their biological context (yellow connection tree).

This holistic approach has the potential to give the domain expert a deeper understanding of possible coherences between gene functions and diseases.

6. Conclusions

In order to analyze huge medical datasets (several hundreds of experiments, several thousands of genes) we applied methods from the field of visual analytics. This was on the one hand done by the development of new visualization methods and on the other hand through the integration of these methods into the Caleydo framework.

Our medical visualization kit consists of multilevel glyphs encoding patient data, parallel coordinates, a heat map view focused on the analysis of gene expression data, and pathway graphs showing the biological processes which are highly influenced by the regulation of genes.

The Caleydo framework integrates views in a linked 3D environment and supports filtering and visual links for a broad range of medical data types. In the visualization process the heat map and the parallel coordinates are the crossing point between clinical and biomolecular data. Each cell in the heat map can be visually linked to either visualizations of clinical data, molecular data or the pathways containing the linked genes

The visualization strongly depends on the quality of the input data. The real world usage of our tools has shown that a lot of effort is necessary in the clearing of the input data. The Caleydo framework was very useful to find blank spots in the data space and to monitor the data quality.

7. Acknowledgments

This work was funded by the FIT-IT program (813 398), the Fonds zur Förderung der wissenschaftlichen Forschung (FWF) (L427-N15) and the Zukunftsfonds Steiermark (3007). Medical data has been provided in the context of the Austrian Genome Program GEN-AU and the CRIP project. We thank H. Sültmann (German Cancer Research Center Heidelberg, Germany) for providing gene expression data. The study has been approved by the ethical committee of the Medical University of Graz.

8. References

[1] Hege H., Hutunau A., Kähler R., Merkzky A., Radle T., Sedel E., Ullmer B., Progressive retrieval and hierarchical visualization of large remote data, Proceedings of the Workshop on Adaptive Grid Middleware, 2003.

[2] Keim D.A., Visual Exploration of Large Data Sets Communications of the ACM, August 2001, Volume 44, Issue 8, 38–44.

[3] Keim D., Kirgel H.P-, VisDB: Database Exploration Using Multidimensional Visualization, IEEE Computer Graphics and Applications, Volume 14, Issue 5, 1994.

[4] Grinstein G., Menses C., Visual Data Exploration in Massive Data Sets, in Information Visualization in Data Mining and Knowledge Discovery, Morgan-Kaufmann Publishers, 2001.

[5] Fekete J., Plaisant C., Interactive Information Visualization of a Million Items, Proc. of IEEE conference on Information Visualization, Boston, pp. 117-124, 2002.

[6] Jiye A., Xudong L., Huilong Duan, Integrated Visualization of Multi-Modal Electronic Health Record Data, Proceedings of the 2nd International Conference on Bioinformatics and Biomedical Engineering, pp. 640-643, 2008.

[7] Chittaro L., Visualization of Patient Data at Different Temporal Granularities on Mobile Devices, Proceedings of AVI 2006: 8th International Conference on Advanced Visual Interfaces, ACM Press, New York, May 2006, pp. 484-487.

[8] Ganslandt T., Jantsch S. Mascher K. Prokosch H.U., Digging for Hidden Gold: Time-Bases Visualization of Heterogeneous Clinical Data, J. Qual. Life Res. Vol. 3, Issue 2, 2005.

[9] Aigner, W., Miksch, S.: CareVis: Integrated Visualization of Computerized Protocols and Temporal Patient Data, Artifical Intelligence in Medicine (AIIM), Vol. 37, No. 3, p. 203-218, Elsevier, July, 2006.

[10] Bio Ressource Med, www.bioresource-med.at/ last visited March 2009.

[11] CRIP - Central Infrastructure for Biomedical Research involving humantissue repositories www.crip.fraunhofer.de/en/site_overview, last visited March 2009.

[12] ICD-10: international statistical classification of diseases and related health problems: tenth revision. World Health Organization 2004.

[13] Percy, C., Fritz, A, Jack, A., Shanmugarathan, S., Sobin, L., Parkin, D.M., Whelan, S., International Classification of Diseases for Oncology (ICD-O), World Health Organization, 2000.

[14] Streit M., Kalkusch M., Kashofer K., Schmalstieg D., Navigation and Exploration of Interconnected Pathways, Proceedings of EuroVis2008, Eindhoven, Netherlands, May 2008.

[15] Ropinski T., Preim B., Taxonomy and Usage Guidelines for Glyph-based Medical Visualization, Proc. of Simulation and Visualization, pp. 121-138 , 2008.

[16] Ward M. O., Multivariate Data Glyphs, Principles and Practice, in Chun-Houh Chen, Wolfgang Härdle and Antony Unwin, Handbook of Data Visualization, Springer, 2008.

[17] Müller H., Zatloukal K., Streit M., Schmalstieg D., Interactive exploration of medical data sets, Proceedings of the Fifth International Conference on Information Visualization in Medical and Biomedical Informatics, London, 2008.

[18] Inselberg A., Modern parallel coordinates. In Eurographics Tutorials, volume 2, pages 967–1060, 2006.

[19] Hauser H., Ledermann F., Doleisch H., Angular brushing of extended parallel coordinates. In INFOVIS '02: Proceedings on Information Visualization, pages 127–130, Washington, DC, USA, 2002. IEEE Computer Society.

[20] Eisen M.B., Spellman P.T., Brown P.O., Botstein D., Cluster analysis and display of genome-wide expression patterns. Proc. Natl. Academy of Science USA, 95(25):14863–14868, December 1998.

[21] Seo J., Shneiderman B., Interactively exploring hierarchical clustering results. Computer, 35(7):80–86, 2002.

[22] Kanehisa, M. et al, KEGG for linking genomes to life and the environment, Nucleic Acids Research, 36, 480-484, 2008.

[23] www.biocarta.org, last visited March 2009.

[24] Collins, C., Carpendale, S., VisLink: Revealing relationships amongst visualizations. IEEE Transactions on Visualization and Computer Graphics, Proceedings of the IEEE Conference on Information Visualization (InfoVis '07), 13(6), 2007.