Scalable, Dynamic Analysis and Visualization for Genomic Datasets

Grant Wallace[1], Matthew Hibbs[1,2], Maitreya Dunham[2],
Rachel Sealfon[1], Olga Troyanskaya[1,2], and Kai Li[1]

[1]Department of Computer Science
[2]Lewis-Sigler Institute for Integrative Genomics
Princeton University, Princeton, New Jersey 08544
{gwallace,mhibbs,maitreya,sealfon,ogt,li}@princeton.edu

Abstract

A challenge in data analysis and visualization is to build new-generation software tools and systems to truly accelerate scientific discoveries. The recent focus of Princeton's next-generation software project is to investigate how to develop new-generation data analysis and visualization capabilities for genomic scientists to analyze high-throughput genomic datasets. This paper describes the software tools we have recently developed to enable dynamic, large-scale data analysis and visualization of multiple datasets on large-scale, high-resolution display wall systems. Our initial experience with the deployed tools at Princeton's Lewis-Sigler Institute for Integrative Genomics is very encouraging. Scientists can effectively learn new knowledge from multiple datasets, find new insights, and generate new hypotheses that are not possible with current methods.

1 Introduction

Princeton systems researchers in the Scalable Display Wall project have been working together with genomic researchers in the functional data analysis lab to explore new methods to develop next-generation software tools and systems for dynamic, scalable data analysis and visualization for multiple large genomic datasets.

The rationale for our focus is due to the fact that scientific disciplines have been experiencing a data avalanche and scientific research has been limited by data analysis. As a result of the technological revolution in the computer industry, science is being transformed -- most scientific disciplines are becoming data rich[1]. The most exciting aspect of the data explosion is the potential to effectively use these data to bring an explosion of knowledge in a scientific discipline.

Biology, like other science disciplines, has experienced generation of previously unprecedented amounts of genomic data. The most abundant sources of such data are high-throughput gene expression microarray experiments, which provide measurements of each gene's level across many biological conditions. During the past decade, the number of datasets and the size of each dataset have been increasing dramatically, whereas data analysis and visualization tools are severely limited due to several challenges.

The first challenge is the ability to analyze multiple large datasets. Microarray technology examines the expression level of all genes in a genome simultaneously in one experiment. Most investigations perform dozens to hundreds of such experiments, resulting in datasets containing millions of pieces of information (with each row measuring levels of expression of one gene across many experiments). As more laboratories perform these types of experiments, the compendium of available data is increasing beyond the capacity for traditional biology to analyze. For example, well over a quarter billion microarray measurements have been generated by several laboratories studying various aspects of cancer, making microarrays the richest source of genome scale functional data. Despite this wealth of data, existing software for the analysis of microarray data focuses on the scale of individual datasets, leaving these methods unable to handle the sheer volume of data that is currently available[2; 3].

The second challenge is to provide the ability for scientists to compare patterns of expression not only within their own datasets, but also among previously published datasets in order to address critical biological questions, such as identifying global patterns of gene transcription regulation, observing similarities between responses of cells to different drugs, and analyzing stages of disease progression. This capability is vitally important for genomic research, but no current approach allows investigators to view data on this scale.

The third challenge is to allow scientists to visualize genomic data in a large context, such as whole dataset or multiple datasets, as well as in a local context, such as a set of similarly regulated genes, in detail. A typical genomic dataset now includes 6,000 to 50,000 gene measurements over hundreds of experiments[4].

1-4244-0910-1/07/$20.00 ©2007 IEEE.

Scientists need to visualize tens of such datasets simultaneously. Today's 2-million-pixel, 30-inch desktop display can only visualize a tiny percent of such visualization task at a time. Using large-format scalable display walls can improve the visualization capability by about two orders of magnitude due to high resolution and scale[4; 5].

We have been developing several software tools to solve these challenging issues. We propose a novel software technique for the visualization and analysis of microarray data in a manner appropriate to integrating results from all publicly available data. Specifically, we focus on enabling researchers to place their own results in the greater context of related data available in the literature. Such an approach requires several functional aspects: an ability to simultaneously observe the same data variable across a large number of different datasets, a way to visualize the context of these variables in each of their individual datasets, intuitive search and exploration methods to find patterns, and higher-level functionality to perform statistical analysis. Due to the vastness of the available data, and the desire of researchers to visualize as much of that data at once as possible, large-scale display devices are an excellent option for these approaches.

We have developed a dynamic, scalable visualization software for large-format displays called ForestView. Our software allows researchers to dynamically view and explore multiple microarray datasets at once, to see context within those datasets, to make comparisons between datasets, and provides an excellent platform for expansion with additional tools and techniques. Our approach is scalable for use in both a desktop/laptop setting and for use on very large-format display devices.

Scientists who are using our deployed systems have identified examples of biological observations that can be easily made using our approach, but are not evident using existing software techniques. Our initial investigations suggest that data visualization and analysis techniques such as this one will play a critical role in advancing scientific discoveries as it allows researchers to draw conclusions and form hypotheses that are otherwise impossible or very challenging to generate.

2 Multi-dataset visualization

We have developed an initial system to help meet the challenges described in the introduction. Central to these challenges is the ability to do analysis and visualization across multiple microarray datasets simultaneously. Previous groups have used large-scale displays to visualize massive datasets such as network packet activity[6], but this type of work is typically domain specific and none has focused on genomics datasets. Visualization tools for genomics have typically been limited to viewing individual datasets rather than multiple datasets. In addition they have not integrated the visualization and analysis processes into a seamless method.

To be effective, our visualization system must satisfy several key requirements. Shneiderman presents a taxonomy of data visualization with a common theme of "Overview first, zoom and filter, then details-on-demand"[7]. We target 2D genomics data within this taxonomy but must meet overview, zoom and filter demands. The visualization must provide both global and zoom views of the data in order to identify and explore patterns across the datasets. It is also necessary to readily identify select groups of genes at both the global and zoom views in order to understand the context and tightness of grouping among those genes. In order to do filtering it is necessary to provide an interface that integrates access to the multiple datasets so that analysis programs can operate across all datasets easily. To provide details-on-demand we must present both synchronized and unsynchronized dataset views. Synchronized views would show the gene expression data aligned across all datasets. This would allow users to scan across a row of data to see how genes from one dataset are expressed in the others. Unsynchronized views would allow users to explore how a grouping of genes from one dataset gets grouped in other datasets and allow exploration of the gene ordering within each dataset. In addition the software should be platform independent to accommodate scientists that work across varied computer platforms including Windows, Mac and Linux.

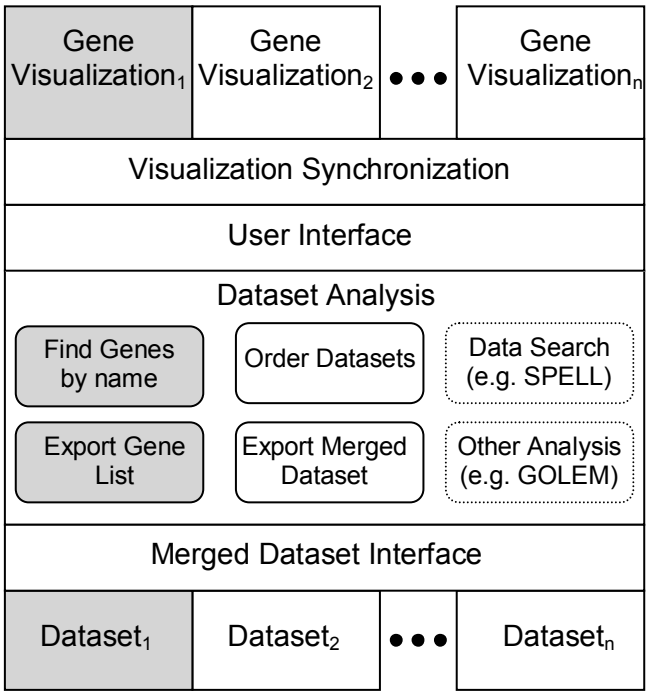

Figure 1: Software architecture of ForestView system.

To accomplish these tasks we base our development on the architecture depicted in figure 1. At the bottom level are the microarray datasets typically accessed through cdt or pcl files. A dataset interface is needed to manage access to all datasets and present a simple three dimensional array interface that allows analysis routines to easily access the data. The analysis routines then do data processing such as finding and ordering genes and datasets based on certain criteria. The user interface allows for user interaction like selecting clusters of genes or tree nodes, panning and zooming views, and adjusting color and display settings. The gene data is then visualized in either a synchronized or unsynchronized fashion.

We desire to leverage existing software for this task. There are visualization tools that operate on a single dataset such as Java TreeView[3], and analysis tools that operate independently of a complete visualization package. Our goal is to leverage these systems and create a new system that meets our multi-dataset requirements.

Java TreeView forms a good starting point for the visualization component. It operates on only one dataset, but can perform many of the visualization and basic data analysis tasks needed and is written in Java which gives cross-platform support. The highlighted grey boxes in figure 1 represent the functionality that Java TreeView implements. We extend Java TreeView by adding the un-highlighted (white) components in figure 1. These include the interface that merges access to the datasets, the extended user interface, a layer that synchronizes the gene views, and the integration of other data analysis procedures. The resulting application, called ForestView, provides for multi-dataset visualization and analysis.

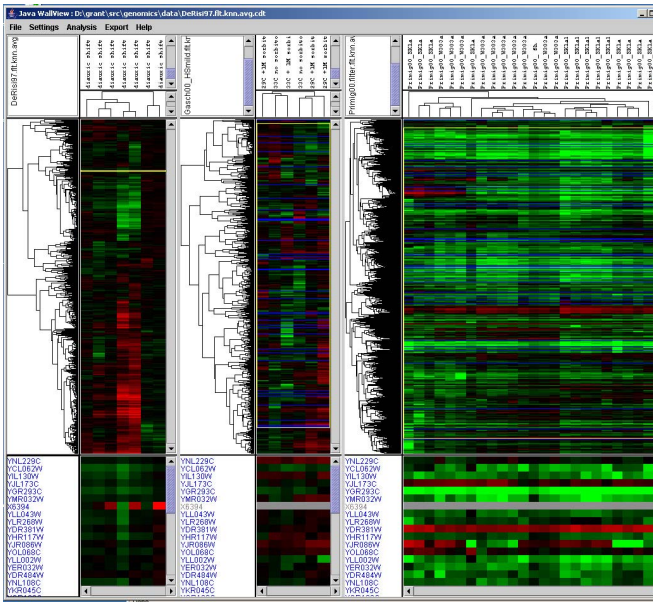

Figure 2: ForestView application displaying a gene subset across three datasets.

The ForestView display is divided into multiple vertical panes, each pane displaying one dataset. Each dataset pane shows a global view of the whole genome and a zoom view showing details of selected genes or a selected region. The gene and array hierarchies are also shown along with annotation information (figure 2).

An important feature of ForestView is the synchronous viewing of gene expression data. When a set of genes is selected, the zoom view for each dataset shows the gene expression data in exactly the same order and same scroll position. This allows the user to scan horizontally across a row of expression data where each row corresponds to data for the same gene even though it crosses multiple datasets. If desired it is possible to turn off synchronous viewing in order to see the selected subsets in the underlying gene order of each dataset.

There are several methods available for choosing a gene subset. A common method is simply using the mouse to highlight a region within the global view of one dataset. Once highlighted, all of the other datasets will search for occurrences of those genes and highlight their position in the global view with a line. The zoom view will then contain the expression data for genes in the subset. Another method is to search over the gene annotation information by entering a list of search criteria. The search is conducted across all datasets and the synchronized results are displayed. Finally, the most adaptive method is to provide selection information from an analysis application. This can help to iteratively adjust the viewed gene subsets in tandem with statistical analysis.

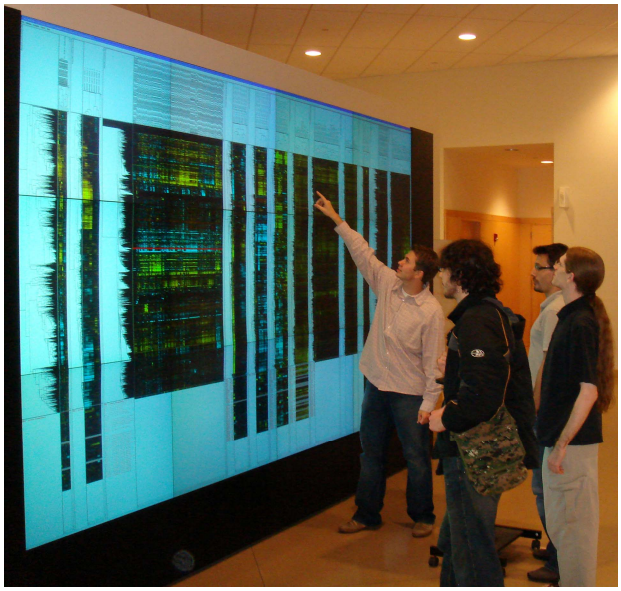

Figure 3: A group of collaborators using the ForestView application on the display wall at the Lewis-Sigler Institute for Integrative Genomics.

When an interesting gene subset is identified, the user can export the gene list, and if desired all of the

expression data, for further analysis in another application. This subset can also be loaded into the ForestView display as a dataset. ForestView also allows users to change user preferences on a per-dataset basis. For instance the scaling of the global and zoom view, the annotation information and the expression level colors can be adjusted independently for datasets or applied to all datasets.

Lastly, the ForestView framework is scalable for use on displays ranging from desktop or laptop screens to large-scale display devices. Using ForestView on a very large display enables users not only to simultaneously view more data at once, but also helps foster a collaborative environment for data analysis (figure 3).

3 Extensions with additional analysis and visualization tools

In addition to methods that allow researchers to view and interact with large amounts of biological data, it is very critical to integrate higher-level analysis and visualization functionality for dynamic search and examination of the data. To address this challenge, we plan to integrate the powerful visualization framework of ForestView with additional approaches to microarray data analysis and visualization that we developed.

The first of these methods is SPELL (Serial Patterns of Expression Levels Locator)[8]. SPELL is a similarity search method designed to work with very large compendia of gene expression microarray data to perform biologically relevant searches. The basic paradigm of this method is to take a small query of related genes from a user, examine all of the available data to identify datasets where these genes are most related, then within those datasets identify additional genes that relate back to the query set.

SPELL's key contribution lies in that rather than searching through a collection of data by text matches, SPELL uses the information within the data to find biologically relevant relationships and perform meaningful searches. The output of SPELL is both an ordered list of genes and an ordered list of datasets, which can be very difficult to visualize and present back to researchers using existing techniques. This type of search is imperative to the biology community in order to explore the vast amount of data available.

Currently SPELL runs on a pre-defined collection of microarray data through a web interface (figure 4), however, this system does not allow a user to observe the dataset context of the returned genes and severely limits the amount of post-search analysis that can be done. The framework provided by ForestView is ideal to display the results of a SPELL search. The datasets returned can be displayed in decreasing order of relevance to the query, and the top n genes can be selected and highlighted within each dataset. This type of output allows researchers to visually verify and understand the results of this search algorithm, which will enable more complete data exploration.

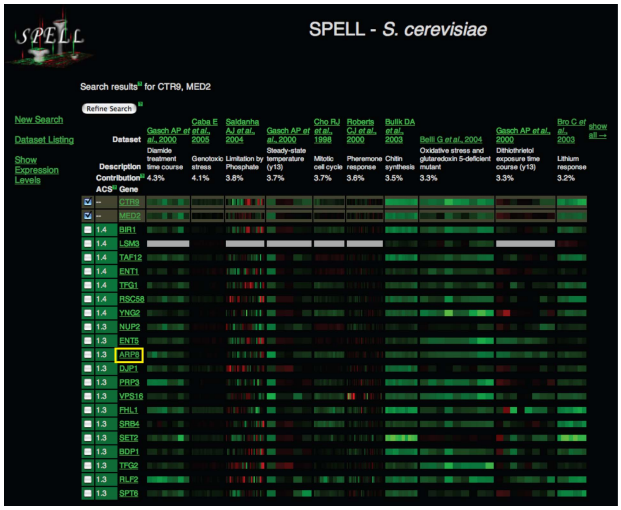

Figure 4: The SPELL web interface displaying the results of a search through a very large compendia of microarray data.

In addition to microarray search, another common task biologists perform when analyzing clustered microarray data is to consider whether a group of genes is active (at a statistically significant level) in a particular biological process or pathway. This type of analysis is often accomplished with the aid of the gene ontology (GO) hierarchy[9]. GO organizes known biological information into a hierarchical graph structure appropriate for use in evaluating hypotheses, observing functional relationships, and categorizing results. However, the large amount of information contained in both the structure and annotations of GO can be difficult to work with in the plain text format it is provided in.

We have developed a system called GOLEM (Gene Ontology Local Exploration Map)[10] to visualize the GO graph structure and to perform statistical enrichment analysis. GOLEM provides a powerful framework for quantifying the statistical functional enrichment of lists of genes. The combination of analysis and visualization in GOLEM allows researchers to not only perform robust statistical analyses of clusters, but also to view how those results relate to each other in the larger context of the GO hierarchy (figure 5).

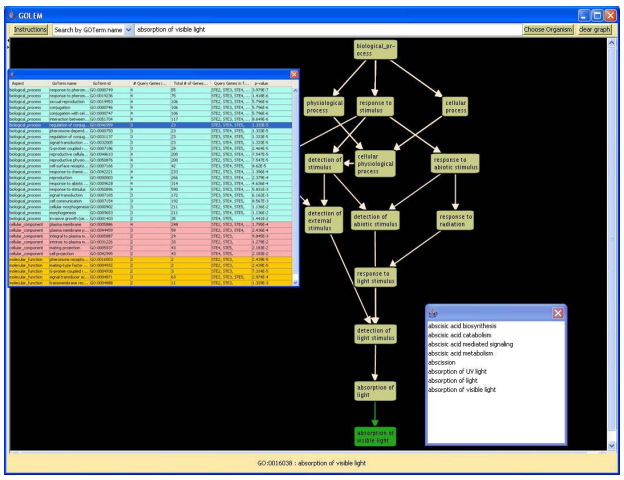

Figure 5: A portion of the gene ontology (GO) hierarchy displayed using the GOLEM system.

While functionality like that of GOLEM is valuable as a standalone program, the tasks that it performs are often desired within the context of analyzing microarray data. For example, a researcher may be interested in a group of genes that appear to behave in a similar fashion over a selection of data. Currently a researcher would export a list of genes selected from the microarray browsing software and input that list into another program to find any statistical enrichment of biological functions. Or, perhaps more often, the researcher will simply examine the list of selected genes by eye to find patterns.

We plan to incorporate the functionally of GOLEM with the microarray analysis and visualization capabilities of ForestView. Tight integration of these methods will allow researchers to quickly and easily quantitatively assess patterns in the data they work with. By viewing the original microarray data along with statistical enrichment values and the greater context of the GO hierarchy, users will be able to see a more complete picture of the information available.

Combining powerful analysis capabilities with intuitive, interactive visualization creates a complete system for researchers to use while analyzing gene expression microarray data. Such a system can provide a great deal of relevant information in a coherent manner to facilitate deeper insights and analysis (figure 6).

While the inclusion of search methods and functional enrichment analysis are natural extensions of microarray visualization and analysis functionality, the framework provided by ForestView is flexible enough to include additional extensions as they become available. As the state of the art of biological experimentation and analysis develops at a very fast rate, is it vital for visualization and analysis tools to adapt and develop as well.

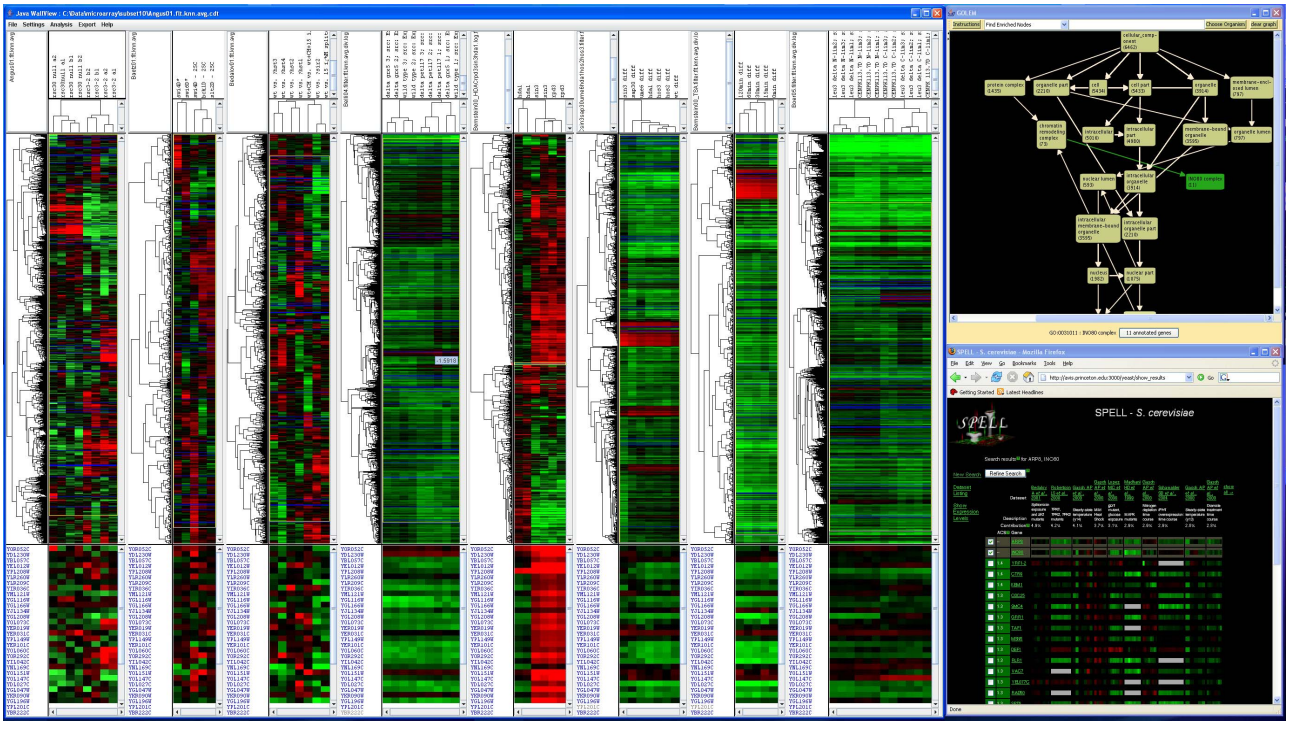

Figure 6: The ForestView system (left) viewed with two other microarray analysis and visualization tools, GOLEM (upper right) and SPELL (lower right).

4 Biological insights

The combination of robust statistical analysis methods with intuitive, interactive visualization methods results in a system that provides a powerful framework for biological microarray analysis. Although we are still in the process of integrating these aspects, we have already seen cases where our collaborators were able to quickly make biological insights using ForestView that were not apparent using any other technique.

For example, a collaborator interested in studying stress response and growth rate effects in yeast was able to draw several novel conclusions from previously published data. We were able to simultaneously examine the expression levels of genes in a set of standard stress response datasets[11] as well as results from a nutrient limitation study[12] and a collection of gene knockout experiments[13]. The biological question our collaborator wished to examine is whether or not the traditional global stress response signal is present in other types of data.

Using ForestView, we were able to easily find and select clusters of genes in the nutrient limitation and knockout studies that our collaborator suspected may be the result of a stress response effect, and then examine how those genes related to each other within the standard collection of stress datasets. Performing this type of analysis is simple in ForestView; however using previously existing techniques we would need to launch over a dozen independent instances of a program and continually cut and paste selections between instances.

Our collaborator identified several groups of genes in these datasets that exhibited a strong pattern of correlation within the stress response datasets as well. This suggests that the effect on gene expression of various nutrient limitations and gene knockouts may be superceded by the more general stress response effect. Our collaborators are currently performing further analysis, both in the lab and with ForestView to better characterize this phenomenon. Clearly, tools like ForestView enable researchers to view their data in a novel manner that facilitates new discoveries.

5 Conclusions

Effective visualization methods that utilize large-scale displays are becoming increasingly necessary for effective analysis of scientific data. We have developed new software tools for genomic data analysis and visualization on large-format, scalable display wall systems. These systems have proven highly beneficial to the researchers in the field, and have the potential of being adopted in other laboratories.

6 Acknowledgements

This work was supported in part by NSF grant CNS-0406415 and EIA-0101247.

References

[1] Gray J, Szalay A. Where the Rubber Meets the Sky: Bridging the Gap between Databases and Science. *Technical Report, MSR-TR-2004-110,* 2004.

[2] Saeed AI, Sharov V, White J, Li J, Liang W, Bhagabati N, Braisted J, Klapa M, Currier T, Thiagarajan M, Sturn A, Snuffin M, Rezantsev A, Popov D, Ryltsov A, Kostukovich E, Borisovsky I, Liu Z, Vinsavich A, Trush V, Quackenbush J. TM4: a free, open-source system for microarray data management and analysis. *Biotechniques,* 34(2): 374-8, 2003.

[3] Saldanha AJ. Java Treeview--extensible visualization of microarray data. *Bioinformatics,* 20(17): 3246-8, 2004.

[4] Wallace G, Anshus OJ, Bi P, Chen H, Chen Y, Clark D, Cook P, Finkelstein A, Funkhouser T, Gupta A, Hibbs M, Li K, Liu Z, Samanta R, Sukthankar R, Troyanskaya O. Tools and applications for large-scale display walls. *IEEE Comput Graph Appl,* 25(4): 24-33, 2005.

[5] Li K, Chen H, Clark D, Cook P, Damianakis S, Essl G, Finkelstein A, Funkhouser T, Klein A, Liu Z, Praun E, Samanta R, Shedd B, Singh JP, Tzanetakis G, Zheng J. Building and Using a Scalable Display Wall System. *IEEE Comput Graph Appl,* 20(4): 29-37, 2000.

[6] Wei B, Silva C, Koutsofios E, Krishnan S, North S. Visualization Research with Large Displays. *IEEE Comput Graph Appl,* 20(4): 50-54, 2000.

[7] Shneiderman B. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualization. *Proceedings of IEEE Visual Languages,* 336-343, 1996.

[8] Hibbs MA, Hess DC, Myers CL, Huttenhower C, Troyanskaya OG. Exploring the functional landscape of gene expression: directed search of large microarray compendia. *Submitted for publication,* 2007.

[9] Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet,* 25(1): 25-9, 2000.

[10] Sealfon RS, Hibbs MA, Huttenhower C, Myers CL, Troyanskaya OG. GOLEM: an interactive graph-based gene-ontology navigation and analysis tool. *BMC Bioinformatics,* 7443, 2006.

[11] Gasch AP, Spellman PT, Kao CM, Carmel-Harel O, Eisen MB, Storz G, Botstein D, Brown PO. Genomic expression programs in the response of yeast cells to environmental changes. *Mol Biol Cell,* 11(12): 4241-57, 2000.

[12] Saldanha AJ, Brauer MJ, Botstein D. Nutritional homeostasis in batch and steady-state culture of yeast. *Mol Biol Cell,* 15(9): 4089-104, 2004.

[13] Hughes TR, Marton MJ, Jones AR, Roberts CJ, Stoughton R, Armour CD, Bennett HA, Coffey E, Dai H, He YD, Kidd MJ, King AM, Meyer MR, Slade D, Lum PY, Stepaniants SB, Shoemaker DD, Gachotte D, Chakrabortty K, Simon J, Bard M, Friend SH. Functional discovery via a compendium of expression profiles. *Cell,* 102(1): 109-26, 2000.