TECHNICAL RESEARCH REPORT

Understanding Clusters in Multidimensional Spaces: Making Meaning by Combining Insights from Coordinated Views of Domain Knowledge (2004)

by Jinwook Seo, Ben Shneiderman

TR 2005-50

Understanding Clusters in Multidimensional Spaces: Making Meaning by Combining Insights from Coordinated Views of Domain Knowledge

Jinwook Seo and Ben Shneiderman*

Department of Computer Science
and Human-Computer Interaction Laboratory
Institute for Advanced Computer Studies
University of Maryland
College Park, MD 20742

*** Correspondence:**
Ben Shneiderman
Department of Computer Science
University of Maryland
College Park, MD 20742
Phone: (301) 405-2680
Fax: (301) 405-6707
Email: ben@cs.umd.edu

Key Words: Cluster Analysis, Interactive Design, Information Visualization, Coordination, Domain Knowledge, Graphical User Interfaces, Dynamic Queries

Acknowledgements
This work was supported by N01 NS-1-2339 from the NIH.

Understanding Clusters in Multidimensional Spaces: Making Meaning by Combining Insights from Coordinated Views of Domain Knowledge

Jinwook Seo and Ben Shneiderman
1/28/2004

Department of Computer Science & Human-Computer Interaction Laboratory, Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742 USA

Abstract

Cluster analysis of multidimensional data is widely used in many research areas including financial, economical, sociological, and biological analyses. Finding natural subclasses in a data set not only reveals interesting patterns but also serves as a basis for further analyses. One of the troubles with cluster analysis is that evaluating how interesting a clustering result is to researchers is subjective, application-dependent, and even difficult to measure. This problem generally gets worse as dimensionality and the number of items grows. The remedy is to enable researchers to apply domain knowledge to facilitate insight about the significance of the clustering result. This article presents a way to better understand a clustering result by combining insights from two interactively coordinated visual displays of domain knowledge. The first is a *parallel coordinates view* powered by a direct-manipulation search. The second is a *domain knowledge view* containing a well-understood and meaningful tabular or hierarchical information for the same data set. Our examples depend on hierarchical clustering of gene expression data, coordinated with a parallel coordinates view and with the gene annotation and gene ontology.

1. Introduction

Cluster analysis is used in numerous research domains, including business, economical, sociological, and biological analyses. Data sets in these domains are usually large (tens of thousands of items) and have more than 3 attributes/variables, making them multidimensional or multivariate [1]. A cluster is a group of data items that are similar to others within the same group and are different from items in other groups. Clustering enables researchers to see overall distribution patterns, and identify interesting unusual patterns, and spot potential outliers. Moreover, clusters can serve as effective inputs to other analysis method such as classification.

Researchers in various areas are still developing their own clustering algorithms even though there already exist a large number of general-purpose clustering algorithms. One reason is that it is difficult to understand a clustering algorithm well enough to apply it to a new data set. A more important reason is that it is difficult for researchers to validate or understand the clustering results in their own way or in terms of their knowledge of the data set. Even the same clustering algorithm might generate a completely different clustering result when the distance/similarity measure changes. A clustering result could make sense to some researchers, but not to others because validity of a clustering result

heavily depends on users' interest and is application-dependent. Therefore, researchers' domain knowledge plays the key role in understanding/evaluating the clustering result.

A large number of clustering algorithms have been developed, but only a small number of cluster visualization tools are available to facilitate researchers' understanding of the clustering results. Current visual cluster analysis tools can be improved by allowing researchers to incorporate their domain knowledge into visual displays that are well coordinated with the clustering result view. This paper describes additions to our interactive visual cluster analysis tool, the Hierarchical Clustering Explorer [3]. These two additions are coordinated views for the researchers' domain knowledge:

- a *parallel coordinates view* enables researchers to search for profiles similar to a candidate pattern, which is specified by direct-manipulation.
- a *domain knowledge view* allows users to compare their clustering results with well-understood and meaningful tabular or hierarchical information of the same data set

Visual analysis by techniques such as dynamic queries has been successfully used in supporting researchers who are interested in analyses of multidimensional data [2][7]. Well-designed visual coordination with researchers' domain knowledge facilitates users' understanding of the analysis result.

We first briefly explain the interactive exploration of clustering results using our current version, HCE 3.0. In section 3, the design considerations for the direct-manipulation search tool and the dynamic queries are explained in detail. Section 4 presents a tabular view showing gene annotation and the gene ontology browser and section 5 covers some implementation issues.

2. Interactive Exploration of Clustering Results with HCE 3.0

Some clustering algorithms, such as k-means, require users to specify the number of clusters as an input, but it is hard to know the right number of natural clusters beforehand. Other clustering algorithms automatically determine the number of clusters, but users may not be convinced of the result since they had little or no control over the clustering process. To avoid this dilemma, researchers prefer the hierarchical clustering algorithm since it doesn't require users to enter a predetermined number of clusters and it allows users to control the desired resolution of a clustering result. HCE 3.0 is an interactive knowledge visualization tool for hierarchical clustering results with a rich set of user controls (dendrograms, color mosaic displays and etc.) (Figure 1). A hierarchical clustering result is generally represented as a binary tree called dendrogram whose subtrees are clusters. HCE 3.0 users can see the overall clustering result in a single screen, and zoom in to see more detail. Considering that the lower a subtree is, the tighter the cluster is, we implemented two dynamic controls, minimum similarity bar and detail cutoff bar, which are shown over the dendrogram display. Users can control the number of clusters by using the minimum similarity bar whose y-coordinate determines the minimum similarity threshold. As users pull down the minimum similarity bar, they get tighter clusters (lower subtrees) that satisfy the current minimum similarity threshold. Users can control the level of detail by using the detail cutoff bar. All the subtrees below the detail cutoff bar are rendered using the average intensity of items in the subtree so that we can see the overall patterns of clusters without distraction by too much detail.

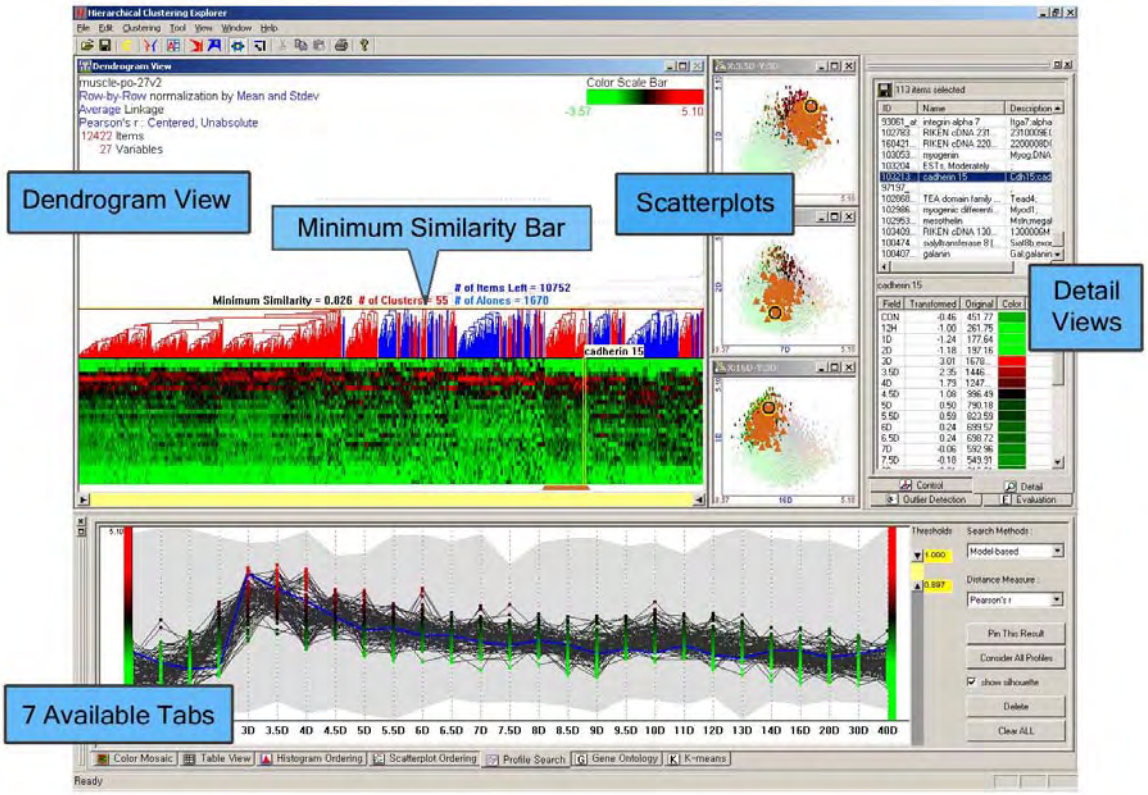

Figure 1. Overall layout of HCE 3.0. Minimum similarity bar was pulled down to get 55 clusters in the Dendrogram View. A cluster of 113 genes is selected in the dendrogram view and they are highlighted in scatterplots, detail view, and parallel coordinates view tab window (see section 3). Users can select a tab among the seven tab windows at the bottom pane to investigate the data set coordinating with different views. Users can see the names of the selected genes and the actual expression values in the detail views.

Since we get a different clustering result as a different linkage method or similarity measure is used in hierarchical clustering, we need some mechanisms to evaluate clustering results. HCE implements 3 different evaluation mechanisms. Firstly, HCE 3.0 users can compare two dendrograms (or hierarchical clustering results) in a single dendrogram view to visually compare the effects of different clustering parameters. Two dendrograms are shown face to face, and when users double-click on a cluster of a dendrogram, they can see the lines connecting items in the cluster and the same items in the other dendrogram. Secondly, HCE 3.0 users can compare a hierarchical clustering result and a k-means clustering result. When users click on a cluster in the dendrogram view, the items in the cluster are also highlighted in the k-means clustering result view (the last tab in Figure 1) so that users can see if the two clustering results are consistent. Thirdly, HCE 3.0 enables users to evaluate a clustering result using an external evaluation measure (F-measure) when they know the correct clustering result in advance. Through these three mechanisms, HCE 3.0 helps users to determine the most appropriate clustering parameters for their data set.

HCE 3.0 was successfully used in two case studies with gene expression data. We proposed a general method of using HCE 3.0 to identify the optimal signal/noise balance

in Affymetrix gene chip data analyses. HCE 3.0's interactive features help researchers to find the optimal combination of three variables (probe set signal algorithms, noise filtering methods, and clustering linkage methods) to maximize the effect of the desired biological variable on data interpretation [8]. HCE 3.0 was also used to analyze *in vivo* murine muscle regeneration expression profiling data using Affymetrix U74Av2 (12,488 probe sets) chips measured in 27 time points. HCE 3.0 's visual analysis techniques and dynamic query controls played an important role in finding 12 novel downstream targets that are biologically relevant during myoblast differentiation [9]. In section 3 and 4, we will use this data set to demonstrate how HCE 3.0 combines users' domain knowledge with other views to facilitate insight about the clustering result and the data set.

3. Combining users' domain knowledge: Parallel coordinates view

Many microarray experiments measure gene expression over time [5][9]. Researchers would like to group genes with similar expression profiles or find interesting time-varying patterns in the data set by performing cluster analysis. Another way to identify genes with profiles similar to known genes is to directly search for the genes by specifying the expected pattern of a known gene. When researchers have some domain knowledge such as the expected pattern of a previously characterized gene, researchers can try to find genes similar to the expected pattern. Since it is not easy to specify the expected pattern at a single try, they have to conduct a series of searches for the expression profiles similar to the expected pattern. Therefore, they need an interactive visual analysis tool that allows easy modification of the expected pattern and rapid update of the search result.

Clustering and direct profile search can complement each other. Since there is no perfect clustering algorithm right for all data sets and applications, direct profile search could be used to validate the clustering result by projecting the search result onto the clustering result view. Conversely, a clustering result could be used to validate the profile search by projecting the cluster result on the profile view. Therefore, coordination between a clustering result and a direct search result make the identification process more valid and effective.

'Profile Search' in the Spotfire DecisionSite (www.spotfire.com) calculates the similarity to a search pattern (so called 'master profile') for all genes in the data set and adds the result as a new column to the data set. The built-in profile editor makes it possible to edit the search pattern, but the editor view is separate from the profile chart view where all matching profiles are shown, so users need to switch between two views to try a series of queries. The modification of master profile in the profile editor view is interactive, but search results are not updated dynamically as the master profile changes.

TimeSearcher [7] supports interactive querying and exploration of time-series data. Users can specify interactive timeboxes over the time-varying patterns, and get back the profiles that pass though all the timeboxes. Users can drag and drop an item from the data set into the query window to create a query with a separate timebox for each time point over the item in the data set. Each timebox at each time point can be modified to change the query.

HCE 3.0 reproduces Spotfire's and TimeSearcher's basic functions with a novel interface, the parallel coordinates view powered by a direct-manipulation search, that

allows for rapid creation and modification of desired profiles using novel visual metaphors. Key design concepts are:

- interactive specification of a search pattern on the information space : Users can submit their queries simply by mouse drags over the search space rather than using a separate query specification window.
- dynamic query control : Users get the query results instantaneously as they change the search pattern, similarity function, or similarity threshold.
- sequential query refinement : Users can keep the current query results as a new narrowed search space for subsequent queries. This enables users to refine their query results, which follows the process of general problem solving.

The parallel coordinates view consists of three parts (Figure 2): the information space where input profiles are drawn and queries are specified, the range slider to specify similarity thresholds, and a set of controls to specify query parameters. Users specify a search pattern by simple mouse drags. As they drag the mouse over the information space, the intersection points of mouse cursor and vertical time lines define control points. A search pattern is a set of line segments connecting the contiguous control points specified. Users choose a search method and a similarity measure on the control panel. They can change the current search pattern by moving a control point (a rectangular point on the search pattern), by moving a line segment vertically or horizontally, or by adding or removing control points. All of these modifications are done by mouse clicks or drags, and the results are updated instantaneously. This integration of the space where the data is shown and the space where the search pattern is composed reduces users' cognitive load by removing the overhead of context switching between two different spaces.

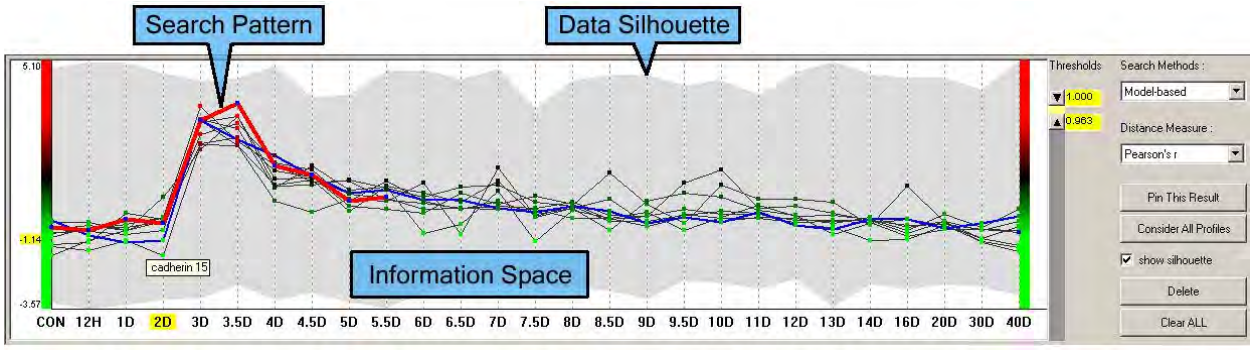

Figure 2. Parallel coordinates view: Layout of the parallel coordinates view and an example of model-based query on the mouse muscle regeneration data. The data silhouette (the gray shadow) represents the coverage of all expression profiles. The red bold line is a search pattern specified by users' mouse drags. Thin regular solid lines are the result of the current query that satisfies the given similarity threshold (more than 96.3% similar to the search pattern). The data set shown is a temporal gene expression profile on the mouse muscle regeneration [9].

Incremental query processing enables rapid updates (within 100 ms) so that dynamic query control is possible for most microarray data sets. The easy and fast search for interesting patterns enables researchers to attempt multiple queries in a short period of time to get important insights into the underlying data set.

In the parallel coordinates view, users can submit a new query over the current query result. If users click "Pin This Result" button after submitting a query, the query result becomes a new narrowed search space (Figure 2). We call this "pinning." Pinning enables sequential query refinement, which makes it easy to find target patterns without losing the focus of the current analysis process. If users click on a cluster in the dendrogram view, all items in the cluster are shown in the parallel coordinates view. By pinning this result, users can limit the search to the cluster to isolate more specific patterns in the cluster.

Genes included in the search result are highlighted in the dendrogram view. Conversely, if users click on a cluster in the dendrogram view, profiles of the genes in the cluster are shown in the parallel coordinates view so that users can see the patterns of genes in a different view other than color mosaic. Through the coordination between the parallel coordinates view and the dendrogram view, users can easily see the representative patterns of clusters and compare patterns between clusters. Since queries done in the parallel coordinates view identify genes with a similar profile, the search results should be consistent with clustering results, if the same similarity function is used. In this regard, the parallel coordinates view helps researchers to validate the clustering results by applying their domain knowledge through direct-manipulation searches.

In the parallel coordinates view, users can run a text search (called search-by-name query) by typing in a text string to find items whose name or description contains the string. Moreover, two different types of direct-manipulation queries are possible in the parallel coordinates view: model-based queries and ceiling-and-floor queries.

Model-based queries: Users can specify a model pattern (or a search pattern) simply by mouse drags as shown in Figure 2, and select a distance/similarity measure among 3 different ones and assign the similarity/distance threshold values. All profiles satisfying the similarity/distance threshold range will be rapidly shown in the information space. The three different measures are 'Pearson correlation coefficient', 'Euclidean distance', and 'absolute distance from each control point'. The first measure is useful when the up-down trends of profiles are more important than the magnitudes, while the second and the third measures are useful when the actual magnitudes are more important. When users know the name of a biologically relevant gene, they can perform a text-based search first by entering a name or a description of the gene (Figure 4). Then they can choose one of the matching genes and make them a model pattern by right-clicking on the pattern and selecting "Make it a model pattern." They can adjust or delete some control points depending on their domain knowledge. Finally, they adjust the similarity thresholds to get the satisfying results and project them onto other views including the dendrogram view.

Ceiling-and-Floor queries: Ceilings and floors are novel visual metaphors to specify satisfactory value ranges using direct manipulation. A ceiling imposes upper bounds and a floor imposes lower bounds on the corresponding time points. Users can define ceilings and floors on the information space so that only the profiles between ceilings and floors are shown as a result (Figure 3). Users can specify a ceiling by dragging with the left mouse button depressed, and a floor by dragging with the right mouse button depressed. They can change ceilings and floors with mouse actions in the same way as they did for

changing search patterns in model-based queries. This type of query is useful when users know the up-down patterns and the appropriate value ranges at the corresponding time points of the target profiles. Compared to model-based queries, ceiling-and-floor queries allow users to specify separate bounds for each control point.

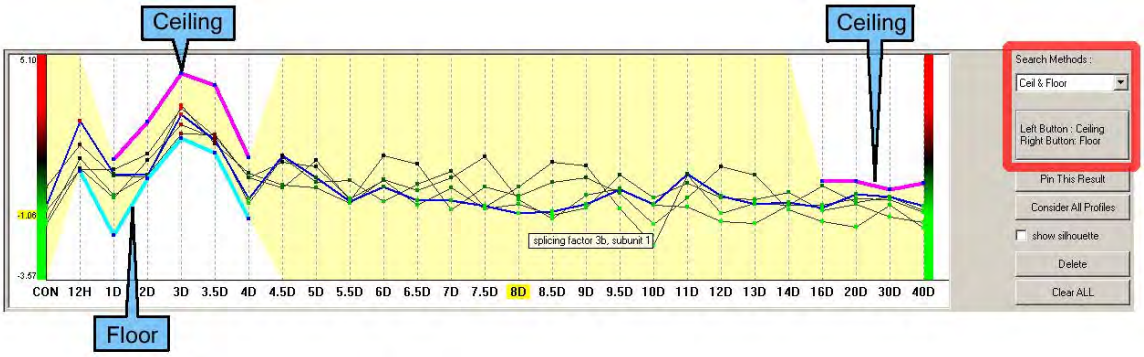

Figure 3. An example of the Ceiling-and-Floor query. Bold line segments above the profiles define ceilings, and bold line segments below profiles define floors. Profiles below ceilings and above floors at the time points where ceilings or floors are defined are shown as a result. Users can move a line segment or a control point of ceilings or floors to modify current query. The highlighted region gives users informative visual feedbacks of the current query. The data set shown is a temporal gene expression profile on the mouse muscle regeneration [9].

Coordination example: Researchers generated *in vivo* murine muscle regeneration expression profiling data using Affymetrix U74Av2 (12,488 probe sets) chips. They measured expression levels at 27 time points to find genes that are biologically relevant to the muscle regeneration process. They already have domain knowledge that *MyoD* is one of genes that are the most relevant to muscle regeneration. They run the hierarchical clustering with the data set, and identify a relevant cluster that has a peak on 3 day (Figure 4). In the parallel coordinates view, they search *MyoD* using search-by-name query, then make it a model pattern to perform a model-based query. They adjust the similarity thresholds to get the search result that mostly overlaps with the relevant 3 day cluster (Figure 4). Finally, they confirm through other biological experiments that 2 genes (*Cdh15* and *Stam*) in the overlapped result set are novel downstream targets of *MyoD*.

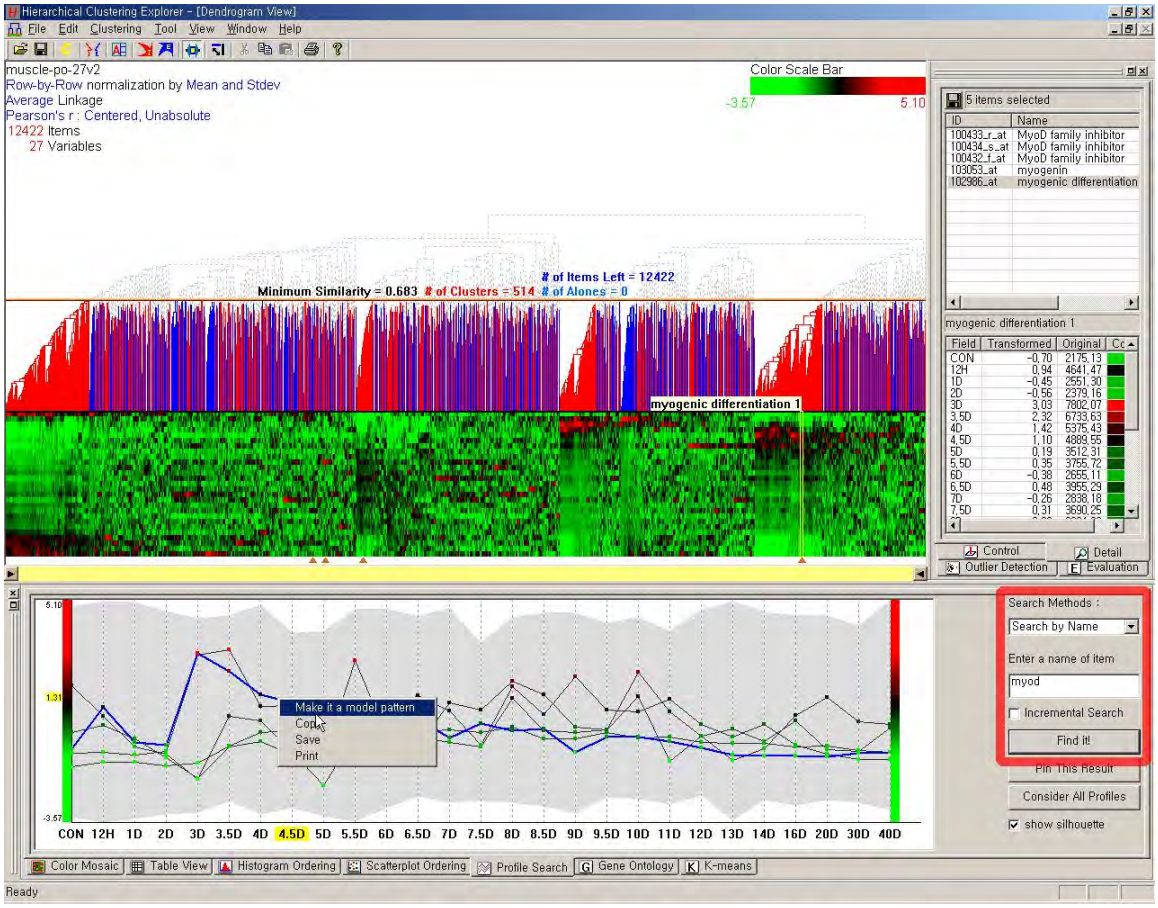

Figure 4(a). Run a search-by-name query with '*MyoD*' to find 5 genes whose name contains *MyoD*, and the 5 genes are projected onto the current clustering result visualization shown by triangles under the color mosaic. Select a gene (*myogenic differentiation 1*) and make it a model pattern for next query.

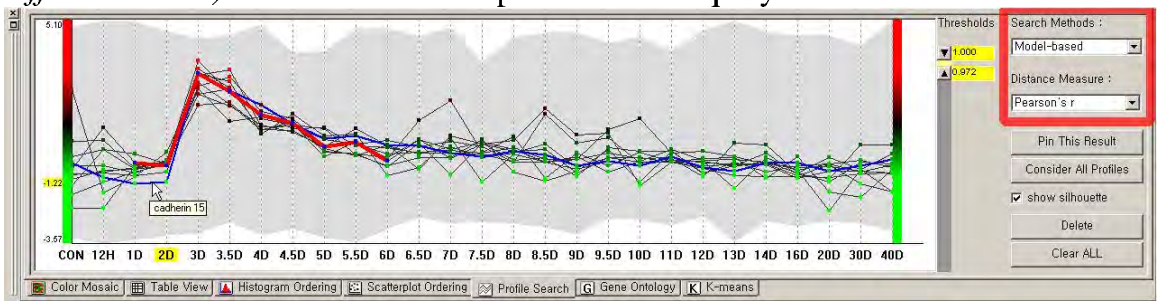

Figure 4(b). Modify the model pattern to emphasize 3 day peak (notice the bold red line), and run a model base query to find a small set of candidate genes. The updated search result will be highlighted in the dendrogram view and the gene ontology browser (see section 4).

Figure 4. An example of coordination with the parallel coordinates view

4. Combining domain knowledge: Tabular and hierarchy viewers

Interactive visualization techniques combined with cluster analysis help researchers discover meaningful groups in the data set. A direct-manipulation search coordinated with clustering result visualization facilitates insight about clustering result and the data

set. Further improvement is possible if there is another well-understood and meaningful knowledge structure for the same data set. For example, when marketers perform a cluster analysis on the customer transaction data, they discover customer groups based on purchasing patterns. If they have another knowledge structure on the data such as the customer preferences or demographic information, they can acquire more insight into the clustering results by projecting the additional information onto the clustering result. In this market analysis example, if a geographic hierarchy of states, counties, and cities were available, it might be possible to discover that purchasers of expensive toys reside in large southern cities. They are likely to be older grandparents in retirement communities.

Coordination between clustering results and external domain knowledge, such as the Gene Ontology, is also being added to commercial software tools, such as Spotfire DecisionSite and CoMotion(www.mayaviz.com). We expand on this important idea by allowing rapid multiple selection in secondary databases through tabular and hierarchical views. The paper continues with the genomic data case study.

Tabular View

In recent decades, biological knowledge has been accumulated in public genomic databases (GenBank, LocusLink, FlyBase, MGI, and so on) and it will increase rapidly in the future [4]. These databases are useful sources of external domain knowledge with which biologists gain insights into their data sets and clustering results. Biologists frequently utilize those databases to obtain information about genomic instances that they are interested in. However, those databases are so diverse that researchers have difficulties in identifying relevant information from the databases and combining them.

HCE 3.0 implements a tabular view (Figure 5) as a hub of database annotations where users can see annotations extracted from those databases for items in the data set. Each row represents an item and each column represents an annotation from an external knowledge source. The tabular view is interactively coordinated with other views in HCE 3.0 as shown in Figure 8. If users select a group of items in other views, rows of the selected items are highlighted in the tabular view. By carefully looking at the annotations for the selected item in the table view and looking them up in the corresponding databases, users can gain more insight into the items by utilizing the domain knowledge from the databases. Conversely, if users select a bunch of rows in the tabular view, the selected items are also highlighted in other views. Researchers can do annotation either manually or by using annotation files provided by gene chip makers. For example, Affymetrix provides annotation files for all their GeneChips, and users can easily import the annotation file and combine it with the data set.

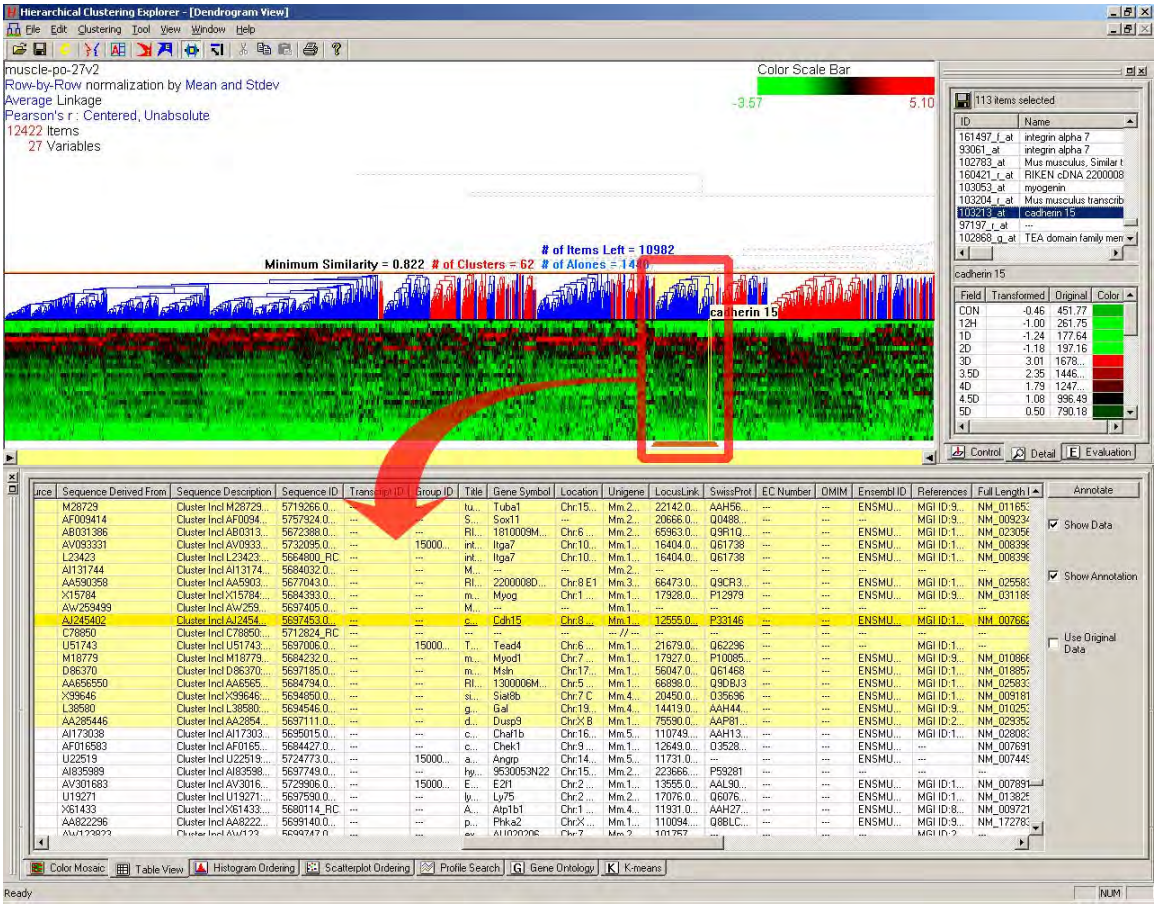

Figure 5. Tabular view: Each row has annotations for a gene. Each column represents an annotation from an external database. All of 12422 genes are in the tabular view, and there are 28 annotation columns. When users select a cluster of 113 genes in the dendrogram view, the annotation information for those genes is highlighted in the tabular view. The Affymetrix U74Av2 chip annotation file downloaded from www.affymetrix.com was imported and combined with the data set. The data set shown is a temporal gene expression profile on the mouse muscle regeneration [9].

Hierarchy View: Gene Ontology Browser

One of the major reasons that biologists cannot efficiently utilize the abundant knowledge in public genomic databases is the lack of a shared controlled vocabulary. The Gene Ontology (GO) project [6] is a collaborative effort of biologists to build consistent descriptions of gene products in different databases. The GO collaborators have been developing three ontologies - structured, controlled vocabularies with which gene products are described in terms of their associated biological processes, molecular functions, and cellular components in a species-independent manner.

The good news is that Gene Ontology (GO) annotation is a widely accepted, well-understood and meaningful knowledge structure for gene expression data. GO annotations of genes in a cluster or a direct manipulation search result might reveal a clue about why the genes are grouped together. With the GO annotation, researchers can easily recognize the biological process, molecular function, and cellular component that genes in a cluster are associated with. Furthermore, it is possible to test a hypothesis that

an unknown gene might have the same or similar biological role with the known genes in the same cluster. Interactive coordination with the GO annotation enables researchers to upgrade their insights by combining generally accepted knowledge from other researchers.

HCE 3.0 integrates the three ontologies – molecular function, biological process, and cellular component into the process of understanding clusters and patterns in gene express profile data. The ontologies are shown in a hierarchical structure as in Figure 6. The gene ontology hierarchy is a directed acyclic graph (DAG), but we use a tree structure to show the hierarchy since the tree structure is easier for users to understand and easier for developers to implement than a DAG. Thus, a gene ontology term may appear several times in different branches, but the path from the root to a node is unique.

Users can download the latest gene ontologies from the Gene Ontology Consortium's ftp server ('Get Latest Ontology' button), and browse the ontology hierarchy on its own ('Load Ontology' button). Coordination between the gene ontology browser and other views in HCE 3.0 is bi-directional.

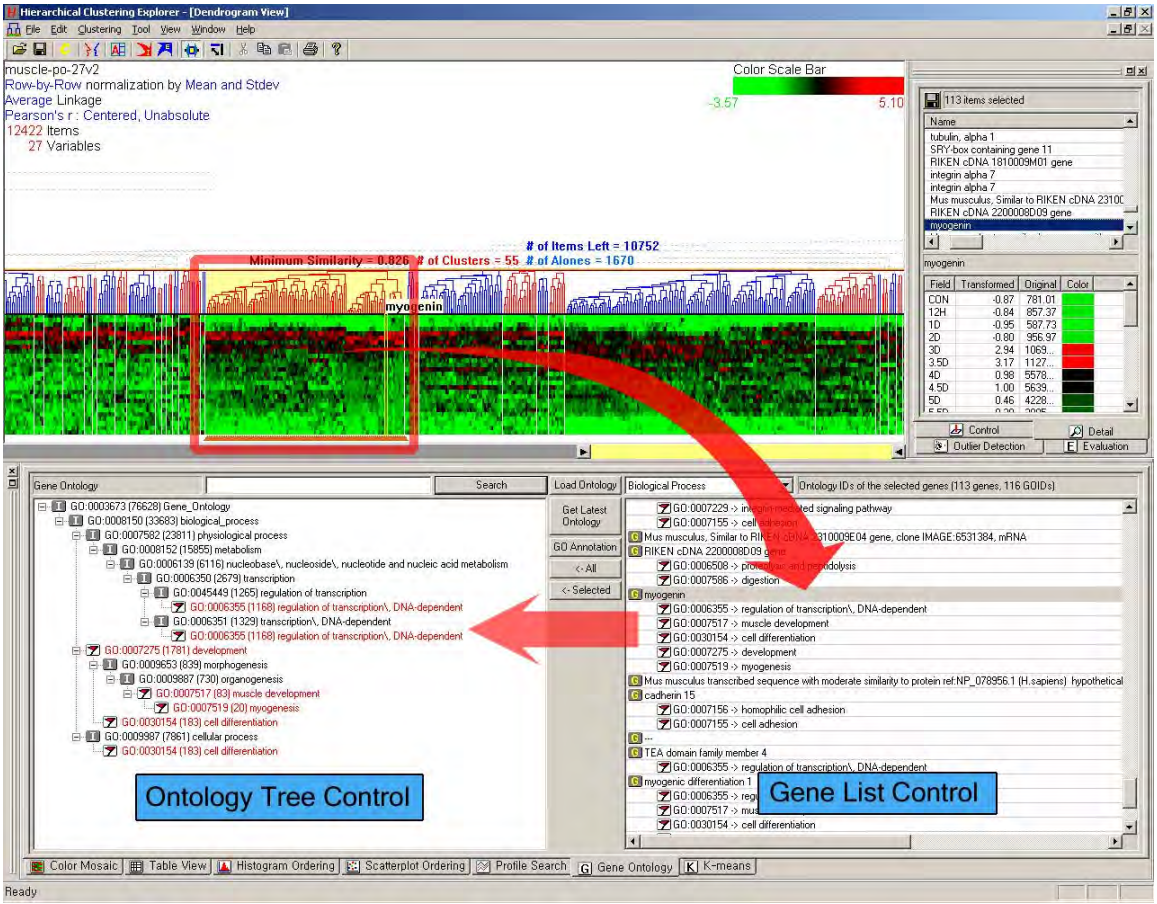

Figure 6. HCE 3.0 with gene ontology browser on. Users can select a cluster in the dendrogram view (at the top left corner), which is highlighted with a rectangle. 113 genes in the selected cluster are shown in the gene list control at the bottom right corner. All paths to the selected GO terms (associated with *myogenin*) are shown with a flag-shape icon in the ontology tree control at the bottom left corner. 'I' represents 'IS-A' relationship and 'P' represents 'PART-OF' relationship. The data set shown is *in vivo*

murine muscle regeneration expression profiling data using Affymetrix U74Av2 (12,488 probe sets) chips measured in 27 time points.

Coordination from other views to the Gene Ontology Browser: Selection of genes in other views such as a click on a cluster in the dendrogram view, a direct-manipulation search in the parallel coordinates view, and a rubber-band selection in a scatterplot populate the gene list control with the selected genes and their GO identifiers as shown in Figure 6 (bottom right corner). Gene names are preceded by the 'G'-shape icon and GO identifiers are preceded by a flag-shape icon. GO identifiers are listed below the gene name with an indentation. If users select a GO identifier in the gene list control, all possible paths from the root to the selected GO identifier in the entire GO hierarchy are shown at the ontology tree control (in the bottom left corner of Figure 6). To reduce clutter, irrelevant paths are hidden. If users select a gene in the gene list control as in Figure 6, paths for all GO identifiers of the gene are shown in the ontology tree control. By taking a look at GO term names shown in the ontology tree control, users can see the detail of biological functions related to the gene described using a shared controlled vocabulary. Clicking on the '← All' button or '← Selected' shows all paths from the root to GO identifiers of all or selected genes in the gene list control. By carefully investigating the shared paths in the ontology tree control, users can learn which molecular function, biological process, or cellular component is related to the genes in the cluster. For example, if all genes in a cluster are mapped to GO nodes below *physiological process* in the biological process ontology, genes in the cluster are likely to be involved in a physiological process.

Coordination from the Gene Ontology Browser to other views: When a gene expression profile data is loaded into HCE 3.0, each gene is mapped to its associated gene ontology identifiers. Each item in the gene ontology tree control shows the number of genes mapped to the item or its descendants within parentheses following the gene ontology identifier. Scrutinizing the numbers next to GO identifiers, researchers can have some idea about which known gene ontology terms better describe the gene expression profile data. If users *right-click* on an item (or, a GO term), all genes mapped to the item or its descendants are highlighted in all other views including the dendrogram view and they are listed in the gene list control (Figure 7). If users want more information about a GO identifier, they can *double-click* on it and HCE 3.0 will launch a web browser and open up a web page for the identifier at *godatabase.org* where users can also find all associated genes across available public data sources (FlyBase, MGI, SRS, etc.).

Coordination example: Researchers annotate their 27 time point murine muscle data set with GO identifiers using the annotation file downloaded from Affymetrix website (www.affymetrix.com). They click on the 3 day cluster in the dendrogram view, or perform a model-based query in the parallel coordinates view and check the GO annotations of the genes in the result to see if there are any shared ontology terms. Conversely, they can browse the ontology tree control and perform a text-based search for GO:0007519 (*myogenesis*) that is one of the most biologically relevant to their experiment. By right-clicking on the GO term, they see all genes that are mapped to *myogenesis* and its descendants are highlighted in the dendrogram view, and then they

realize that many of the genes are in the 3 day cluster. All genes in the cluster actually become candidate genes of novel downstream targets of *MyoD*, and deserve further biological experiments. The coordination with GO would produce more meaningful insights as GO becomes more comprehensive and as more genes are annotated with GO terms.

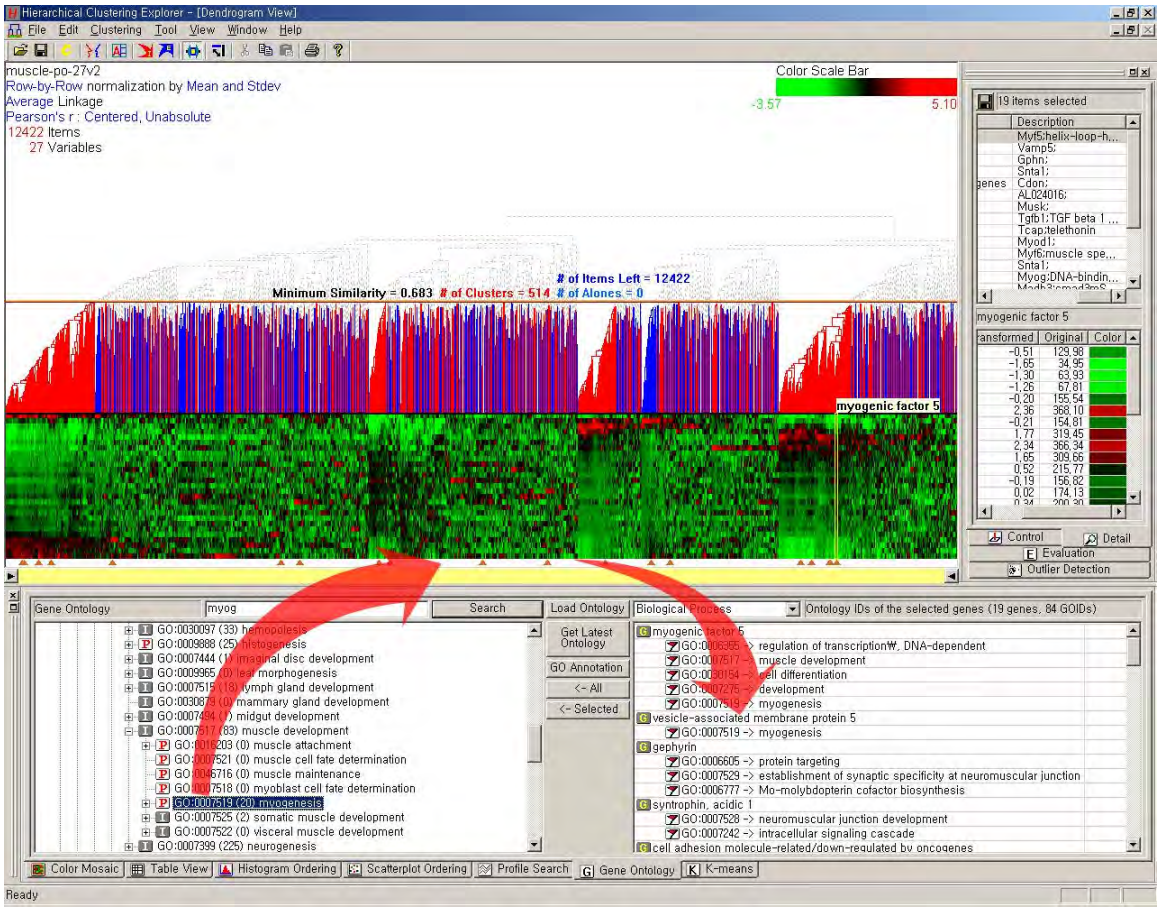

Figure 7(a). Users right-click on a GO identifier at the ontology tree control to highlight all genes mapped to the identifier or its descendants in the dendrogram view. The selected genes are also listed in the gene list control.

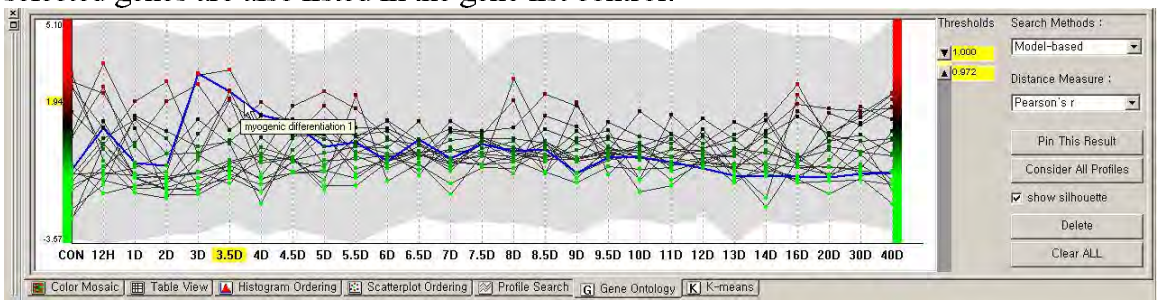

Figure 7(b). The selected genes are also shown in the parallel coordinates view to enable users to check the result in a different view.

Figure 7. An example of coordination with the Gene Ontology Browser

5. Implementation

HCE 3.0 was implemented as a stand-alone application using Microsoft Visual C++ 6.0. The Microsoft Foundation Class (MFC) library was statically linked. HCE 3.0 runs on personal computers running Windows (at least Window 95) without special hardware or external library support. HCE 3.0 is freely available at http://www.cs.umd.edu/hcil/hce/ for academic or research purposes.

Figure 8 shows four tightly coupled components of HCE and linkages between them. Updates by each linkage in Figure 8 are instantaneous (or, it takes less than 100ms) for most microarray data sets.

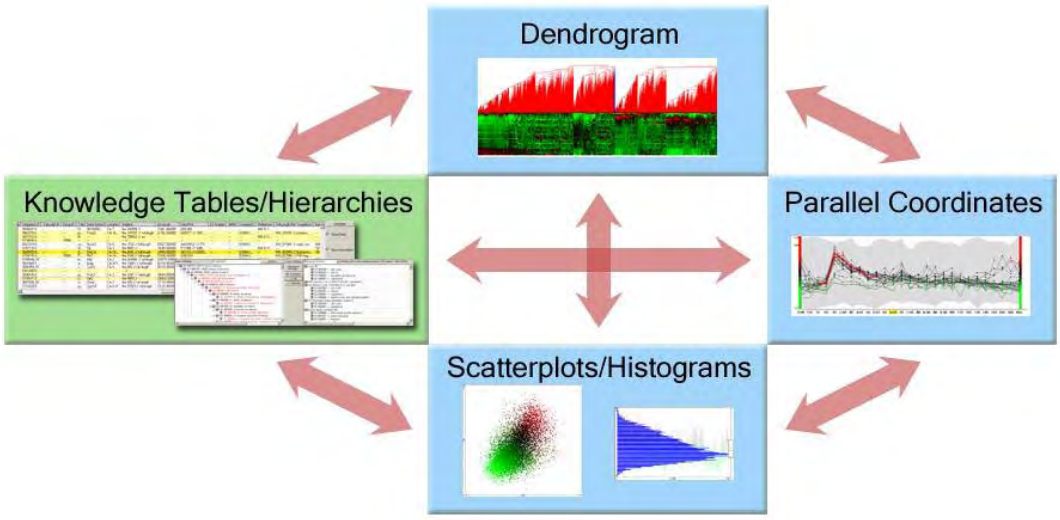

Figure 8. Diagram of interactions between components of HCE 3.0. All interactions are bi-directional. This paper describes coordination between the dendrogram view, parallel coordinates view, and knowledge tables/hierarchies view. Knowledge tables/hierarchies incorporate external domain knowledge while others show the internal data using different visual representations.

To achieve rapid responses to users' actions, hash and map data structures were used because they enable constant time lookup of items, with only a modest storage overhead. Incremental data structures were used to support rapid query update in the parallel coordinate view by maintaining active index sets for intermediate query results.

Microarray experiment data set can be imported to HCE 3.0 from a tab-delimited text or an Excel spreadsheet. The latest gene ontology annotation data is automatically downloaded from the Gene Ontology Consortium's ftp server. The current annotation file with GO annotations for most Affymetrix chips is downloadable from www.affymetirx.com and it can be automatically attached to the input data.

6. Conclusion

Cluster analysis has been the focus of numerous research projects conducted in various fields. It reveals the underlying structure of an input data set, interesting unusual patterns, and potential outliers. Understanding the clustering result has been a tedious process of checking items one by one. With HCE 3.0, we believe users can quickly apply their own

or external domain knowledge to interpret a cluster by visual display in coordinated views.

This paper presented two coordinated views to incorporate users' domain knowledge with visual analysis of the data set and clustering results. First, when users know an approximate pattern of a candidate group of interest, they can use the parallel coordinates view to quickly compose the search pattern according to their domain knowledge and run a direct manipulation search. Second, when there is a well-understood and meaningful tabular or hierarchical information for their data set, they can utilize other researchers' knowledge to make interpretations based on the clustering result. Well-designed interactive coordination among visual displays helps users to evaluate and understand the clustering results as well as the data set by visually facilitating human intuition.

This work is a part of our continuing effort to give users more controls over data analysis processes and to enable more interactions with analysis results through interactive visual techniques. These efforts are designed to help users perform exploratory data analysis, establish meaningful hypotheses, and verify results. In this paper, we show how those visualization methods can help molecular biologists analyze and understand multidimensional gene expression profile data. Empirical validation on standard tasks, more case studies with biological researchers, and feedback from users will help refine this and similar software tools.

Acknowledgements

This work was supported by N01 NS-1-2339 from the NIH.

References

1. A. Inselberg and T. Avidan, "Classification and visualization for high-dimensional data," *Proc. 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2000, pp. 370-374.

2. E. Kandogan, "Visualizing multi-dimensional clusters, trends, and outliers using star coordinates," *Proc. 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001, pp. 107-116.

3. J. Seo and B. Shneiderman, "Interactively exploring hierarchical clustering results," *IEEE Computer*, Vol. 35, No. 7, 2002, pp. 80-86.

4. A.D. Baxevanis, "The Molecular Biology Database Collection: 2003 update," *Nucleic Acids Research*, 31, 2003, pp. 1-12.

5. A. Butte, "The use and analysis of microarray data," *Nature Reviews Drug Discovery,* Vol. 1 No. 12, 2002, pp. 951-960.

6. Gene Ontology Consortium, "Gene Ontology: tool for the unification of biology", *Nature Genet*, 25, 2000, pp. 25-29.

7. H. Hochheiser and B. Shneiderman, "Visual specification of queries for finding patterns in time-series data," *Proceedings of Discovery Science*, Springer, Berlin, 2001, pp. 441-446.

8. J. Seo, M. Bakay, P. Zhao, Y. Chen, P. Clarkson, B. Shneiderman, and E.P. Hoffman, "Interactive Color Mosaic and Dendrogram Displays for Signal/Noise Optimization in Microarray Data Analysis," *Proc. IEEE International Conference on Multimedia and Expo*, 2003, pp. III-461~III-464.

9. P. Zhao, J. Seo, Z. Wang, Y. Wang, B. Shneiderman, and E.P. Hoffman, "In vivo filtering of in vitro MyoD target data: An approach for identification of biologically relevant novel downstream targets of transcription factors," *Comptes Rendus Biologies*, Vol. 326, Issues 10-11, October-November 2003, pp 1049-1065.