Contents

1 Statement of Work

1.1 Summary of Proposed Work

Goal: We propose to enhance and further develop a tool called DEVise for *retrieval, integration and visual analysis of large datasets from one or more data sources*, including tabular data sets, time-series and imagery. DEVise is a public-domain tool for interactive visual exploration of large datasets, and BIRCH is a tool for clustering multidimensional data that is integrated with DEVise. We will leverage existing tools and funding from NASA and ORD by building upon these tools. Our proposal is focused upon specific applications in space sciences, but the underlying research has wide applicability in other domains as well; in addition to space science applications, we are currently applying these tools in areas such as analysis of medical data from the UW Clinics, financial applications using data from a variety of sources including Compustat, CRSP and ISSM, and sales information from a catalog merchant.

Research Topics: As larger and larger datasets are collected and generated by scientists, and the number and diversity of these datasets increases, effective tools for data retrieval, integration and analysis become essential. Given that the Web is quickly becoming the most widely used means of connecting users to datasets, tools that can efficiently and conveniently access local data together with remote data over the Web are especially timely and useful. From a research standpoint, several issues require attention in developing such tools:

1. *Dataset Size:* Tools should be able to deal with datasets larger than main memory by using buffering to bring in portions of data as needed from secondary storage devices. While this is standard for database query engines, it is not so for visualization tools. Further, as the amount of graphical data (i.e., data describing the visual presentation) increases, efficient indexing techniques are required for ensuring interactive response times.
2. *Integrated Presentation of Data and Metadata:* As datasets increase in number, diversity and size, *metadata* becomes an essential tool for managing this complexity. We use the term 'metadata' in a general sense to mean *a summary of a subset of data*. A powerful paradigm is to allow users to explore data and (various kinds of) metadata in a uniform manner, and to *drill down* to the necessary level of detail interactively. How to achieve this, especially in the context of a visual querying environment, is an important focus of this proposal.
3. *Diversity, Distribution and Autonomy of Data Sources:* The Web can be thought of as a gigantic distributed database, and queries over multiple data from multiple sites can be thought of as distributed database queries, but there are important differences: typically, there is little or no local data processing capability, the number and diversity of data sources is enormous, and there is complete autonomy across sites (one consequence of which is that, in contrast to a distributed database, there is no such thing as a global system catalog).
4. *Integration of Database-style Querying and Visualization:* End users often require visual presentations of data. Allowing users to define visual presentations in a flexible manner, and to pose complex queries in terms of these visual presentations, greatly facilitates data retrieval and analysis for users who are not database experts.

Expected Results:

1. An enhanced version of the DEVise system that is capable of supporting data integration and analysis tasks, with emphasis on visual exploration and queries over the Web. In contrast to other related tools, it will allow database-style queries to be combined easily

with visual data exploration, support uniform querying of data and metadata, and implement queries over the Web efficiently.

2. An environment for analyzing data at multiple levels of detail, within the context of the DEVise tool. We believe that understanding how to use metadata and summary data effectively in exploring data is central to interactive analysis of large datasets. Various tools can be used to produce a high-level summary of data, and the visualization tool must support integrated viewing of the summary data, or *metadata*, as well as the original data. This requires the ability to inform the visualization tool about the mapping from a summary object to the data records that correspond to it.

   Specifically, we will develop an enhanced version of the BIRCH clustering package (which produces summaries that are 'clusters' in the data) integrated fully with DEVise, to support interactive classification tasks. More generally, we will provide interfaces to enable other kinds of tools to generate summaries and pass to DEVise the mapping from summary objects to data records.

Project Team: This project involves faculty members at the University of Wisconsin-Madison, graduate students, and a staff programmer. Prof. Livny has led the development of the DeNet simulation package and the Condor system for workstation cluster management system. Prof. Ramakrishnan has led the development of the Coral deductive system. These are research software packages that are considered to be state-of-the-art, and have been used widely in universities and research labs in the US and abroad. Profs. Livny and Ramakrishnan have collaborated in the past, with NASA support, to develop the current versions of the DEVise and BIRCH tools and a system (called SEQ) for querying sequence data, and have also combined Coral and Condor (using ORD support) to provide query evaluation capabilities utilizing idle resources. Prof. Norman has worked with measurements and integrative modeling in the area of environmental biophysics. He has developed a soil- plant-atmosphere model, named Cupid, that has been applied to problems in remote sensing, irrigation scheduling, integrated pest management and water and carbon budgets of ecosystems. Prof. Norman has been the domain scientist driving the development and use of DEVise and BIRCH.

1.2 Introduction

Many large datasets are being generated by scientists, in large projects such as NASA's EOS and Mission to Planet Earth, as well as numerous smaller projects. Datasets from a variety of sources are becoming available through the World-Wide Web. Ease of access to these datasets and effective analysis tools are crucial to the successful exploitation of this data for scientific research. The following trends motivate our work, and we aim to develop a data integration and analysis tool that addresses the attendant problems:

1. Increasing size of datasets, which implies that tools must be able to deal with data spread over secondary storage devices. Further, tools must allow users to examine data at several levels of detail, ranging from individual observations to *metadata* that summarizes a large volume of related data. For example, browsing the data at the level of metadata or summary descriptions may reveal interesting high-level features, and the user may then want to examine relevant portions of the data (visually or analytically) in more detail.

2. Increasing variety and number of data sources, which means that there must be a convenient way to integrate and analyse data from multiple remote sources, and to combine this information with

local data.

3. The need for visual exploration of data to complement analytical tools. This is expecially important when data is noisy or when the user is looking for loosely defined patterns or trends in the data.

4. The Web as a medium for connecting researchers to data sources. This is a convenient means of sharing data between researchers in different locations, and of disseminating data from archival repositories. Ideally, users should be able to query data from several Web sites as if all the data were stored locally; currently available tools cannot do this efficiently when the datasets involved are large.

This proposal can be seen as a logical continuation of the work we have carried out under our current AISRP NASA grant, in which we have developed the SEQ system for analytical queries over sequences and the current version of the DEVise tool for visual queries. We intend to expand the scope of our work to include more general datasets and queries, and to emphasize two new issues:

1. Analytical data transformation and querying capabilities, with an emphasis on efficient access to data sources through the Web.

2. Integrated querying and visualization of data and *metadata*.

These two changes to our research direction are based largely on our experience with the current tools and users' feedback. Originally, DEVise was intended to address just visual exploration. Computation and transformation of the original dataset were carried out using the CORAL system for deductive queries and the SEQ system for queries over sequences. While this paradigm of loosely coupling the data transformation tools with the visualization tool does provide a powerful environment (and continues to be important to users), users clearly wanted to be able to combine data transformation and visualization seamlessly within a single tool. This experience has led us in the direction of providing more data transformation and querying capabilities within a single tool, and this motivates many of the enhancements that we are currently making to DEVise and that we intend to carry out as part of this proposal.

The second issue, namely integrated access to data and metadata, arose out of user feedback on using DEVise in conjunction with the BIRCH clustering tool. The clusters produced by BIRCH can be seen as a summary of the original data, or as a form of *metadata*, providing a compact, high-level description. In our interaction with users of BIRCH and DEVise, it has become very clear that users want to be able to cluster a dataset with a particular choice of clustering parameters (e.g., weights for the different dimensions, choice of distance metric), see the clusters in a visual presentation, then select all records in the *original data* corresponding to one or more clusters and process these records further. This kind of summary is by no means the only useful one, but it is a very good canonical example. Other examples of summaries include statistical measures, highly compressed versions of images, moving averages over a time-series (with the window size controlling the degree of summarization), etc. Clearly, we cannot expect to provide a comprehensive set of tools for generating different kinds of summaries. Rather, we must provide a convenient environment for generating summaries using an appropriate tool, and analysing both the summaries and the original data together in a natural manner. The key here is to enable the visualization tool to understand the connection between a summary and the original data (i.e., which original data records contributed to a summary record, so that the user can go from an interesting portion of the summary to the corresponding portion of the original data).

As concrete applications of our tool, we envisage the following:

- **Quality Assurance:** Is the data noisy? Is there a lot of missing data? Is the loss/noisiness concentrated in certain 'regions' of the data (identified by value ranges for some of the attributes)? Such 'data cleaning' is an ubiquitous and time-consuming task for many scientists.

- **Queries Over the Data:** Scientists often need to identify parts of the data that satisfy certain criteria. For example, to test a climate model, a scientist may want to run it on datasets that were collected under certain atmospheric conditions. Sometimes, determining which part of a dataset satisfies such criteria involves complex analysis; often, however, the conditions to be checked can be expressed as a database-style query, e.g., "Find days when the temperature was higher than some value," or, "Find days when the average weekly temperature for the preceding week was higher than some value."

- **Concept Identification and Classification:** Can we identify instances of domain-specific concepts in the data? For example, given several terrain photographs collected by satellites, can we classify the regions in each photograph into *leaves*, *sunlit regions*, *branches*, etc.? (This is an example of an application that we can already handle, although with more manual intervention than we would like to have ultimately, with the current versions of the DEVise and BIRCH tools.)

- **Hypothesis Verification:** If we believe that there is a causal relationship between ozone levels in the atmosphere and the incidence of skin cancer, it would be helpful to be able to quickly browse relevant datasets to see if they might support such a hypothesis. Then, additional tools can be brought to bear upon such promising datasets.

- **Exploratory Data Analysis:** Can we spot interesting patterns or trends in the data, especially when the dataset is very large? For example, it might be the case that whenever the use of pesticides in an area goes above a certain threshold, there is a corresponding increase in the level of certain contaminants in local rivers. If we do not know this *a priori*, but would be interested in knowing, can tools be developed to assist us in discovering such correlations? Again, a combination of database-style ad-hoc querying and visual exploration could suggest such patterns, and additional tools could then be used to validate them.

  Such exploratory analysis is carried out routinely in many different application domains, but software environments supporting this activity are in their infancy. The problems that must be addressed and the techniques that can be applied have been studied in Statistics, Artificial Intelligence, and in Databases.

1.3 Software that We Propose to Enhance and Build Upon

Our overall goal is to develop a tool to support retrieval, integration and analysis of data from one or more sources over the Web. The tool must be capable of dealing with datasets whose size is larger than available main memory on a typical desktop machine, and must support browsing of metadata (typically generated using other tools) and data in a uniform manner. There are many aspects to such a tool, and we intend to enhance existing tools and software, rather than to develop a new tool from scratch. By doing so, we significantly leverage existing and prior support from NSF, NASA, ORD and the Packard Foundation. In this section, we describe the tools that we intend to enhance, and related software that we intend to build upon. Details of our enhancements and our research objectives are presented in Section 1.4.

DEVise: A Tool for Visual Data Analysis and Exploration

Visual exploration of large, multidimensional data sets is becoming increasingly important for business and scientific communities alike. Data streams are being continuously produced by remote sensors, program trace generators, simulators, and the stock market, to name a few. Interpreting such data in text form is a virtually impossible task due to the quantity of data. Analyzing the data statistically may be impossible if what should be looked for is not fully understood. A visual representation, on the other hand, can often be used to find interesting trends and patterns that cannot be expressed in exact statistical terms.

Large and complex data sets present a challenging task for visualization systems. The size of data streams can easily reach tens or hundreds of megabytes and each record in the stream may contain hundreds of data attributes. It is important for visualization systems to avoid overwhelming the main memory or virtual memory of a workstation, and to allow the user to navigate through the data and metadata without being buried in it. Standard & Poor's CompuStat database, for instance, contains financial data of roughly 5,000 companies. The total size of the database is over 1 gigabyte and each data record has over 350 attributes. The Institute for the Study of Security Markets (ISSM) distributes an annual database containing trade and quote transactions of 5,000 stocks. Each stock is represented as a separate time series. The single largest time series in the 1992 ISSM database is for the IBM stock which recorded almost a million trade and quote transactions, totalling over 20 megabytes in size.

We have developed a data visualization and exploration system called DEVise that attempts to solve both the data size and metadata complexity issues. The system uses off-the-shelf hardware and has minimal main memory requirements. It can handle data that is large in volume, complex in structure (multidimensional and/or hierarchical), and that may be imported from different sources such as database servers, external programs, and World Wide Web resources. Commercial and scientific databases can also be linked to DEVise to allow the user to visualize and analyze related information from heterogeneous sources.

As an example of the use of DEVise, consider the following scenario, which is based on our collaboration with a Soil Sciences user-group. Extensive manual effort is currently required to extract useful data from the FIS or BORIS data bases (maintained by NASA). For example, to compare a prediction of infrared temperature from a soil-plant-atmosphere model (named Cupid) with aircraft and satellite observations, input data (soil, vegetation and weather data) required by the model must be obtained from the data base subject to the constraint that the site was cloud free at the time of satellite and aircraft observation. Furthermore, when a cloud-free period is obtained all the model input data must be available (weather data is collected on half hour intervals, surface soil moisture measured daily, leaf area and deep soil moisture measured weekly with the time of observation varying from station to station) from the previous day of rain to the time of coincidence of clear sky and satellite overpass either by direct measurement or interpolation in time or space. Extracting the appropriate data from the database is itself time consuming, but it does not require as much time as just finding an appropriate time sequence that meets the criteria, which can take months. We note that SQL is wholly inadequate for tasks such as this.

Missing or erroneous data are among the major complicating factors that make this selection process difficult. After the inputs required by the model have been extracted, the model is run; then model predictions of infrared temperatures are compared with measurements from aircraft or satellites, which also need to be extracted from the data system. Typically such comparisons involve time sequences of predictions and measurements. For example, predicted and measured infrared temperature would be plotted as a function of time; they may agree at some times and disagree at other times. On other occasions the two curves may just be shifted in time by some fixed phase. Alternatively they may agree at high wind speeds and disagree at low wind speeds. Clearly, the ability to visually browse such datasets is invaluable.

Defining Visual Queries in DEVise

In essence, we can think of *visual queries* in DEVise as definitions of views that integrate the source data, which can come from a variety of sources including text files, sequence databases, and the World-Wide Web. Using a point-and-click interface, a user can define a 'visual query' that determines how the source data is to be interpreted and displayed on the screen.

In DEVise, a *data stream* consists of binary or ASCII records where each record contains one or more *attributes*. A data stream is combined with a *schema* to produce *TData*. The schema describes the data attributes in the data stream. Each TData record is mapped to a *GData* record where each GData record consists of graphical attributes. This *mapping* can be defined by the user, and provides a very high degree of flexibility in defining visual queries, as the examples in the appendix illustrate.

Example Applications of DEVise

We now present some example applications of DEVise, to illustrate the power of the current tool and its scope.

The first example, shown in Figure DEVise-1, illustrates the combined use of DEVise and BIRCH and is discussed in Section 1.3.

The second example, shown in Figure DEVise-2, illustrates the use of DEVise to view data from multiple sources in a uniform manner. The sugar futures information is retrieved (in real-time, in response to user actions like *scrolling* and *zooming*) over the Web. The S&P information is stored in a local file. The information on DEC is taken from two vendor databases.

The third example, shown in Figure DEVise-3, illustrates the flexibility of DEVise. The data that is visualized is a collection of traces that reflects the execution of an experiment controlled by the CONDOR idle resource utilization software. The two pairs of graphs correspond to two different runs, with different scheduling policies (explained below). The experiment consists of several tasks, each of which must be executed in a round. Rounds of execution continue until the computation converges in a certain manner, and a new round cannot be started until all tasks in the previous round are complete. Clearly, the scheduling policy used for tasks within a round—when a machine becomes available, should we schedule the longest remaining task or the shortest, for example—affects the length of the computation. The visualization brings this out graphically. The X-axis corresponds to time, and the Y-axis corresponds to machines. A given location is assigned a block of a certain color, denoting the task being executed by the corresponding machine at that time. The red graph below shows the cumulative idle time. The 'rounds' of computation are clearly reflected in both graphs. It is worth noting that the datasets in these examples are all considerably larger than the memory available to DEVise—buffering is used to bring in just the portion of the data needed to refresh the display.

BIRCH: A Tool for Clustering Multidimensional Datasets

Finding natural 'clusters' in a multidimensional dataset is a well-studied problem in Statistics, and has been applied in a variety of domains for concept identification and classification. It can be regarded as a variant of the problem of computing a *density function*; this has applicability for constructing histograms of data, for example. Clearly, clustering is a powerful technique for exploratory analysis.

We have developed a clustering algorithm called BIRCH [?] that is capable of clustering very large datasets while using very little memory. It incrementally and dynamically clusters incoming multidimensional metric data points, and can typically find a good clustering with a single scan of the data, and improve the quality further with a few additional scans. A novel feature is that BIRCH exploits its knowledge of the portion of the input dataset that it has already clustered to find-tune the clustering of the rest of the dataset. Thus, it adjusts dynamically to the input dataset to try and produce the best quality clustering with the available resources (i.e., available memory and time constraints). The low memory requirements of Birch enable us to cluster on different subsets of the data attributes (i.e., dimensions) in parallel, and to collect useful statistical information on attributes that do not take part in the clustering.

Multidimensional Constraint Queries

Constraints offer a very powerful way of formulating queries over multidimensional point datasets, and can be used in a variety of applications, including business and scientific analysis. The idea is that a desired region (in the multidimensional space) can be specified as a conjunction of polynomial constraints. As an example, a convex polygon can be represented as a *conjunction of two-dimensional linear constraints.* Each constraint identifies a half-plane; a conjunction of constraints can therefore be used to identify any convex region, as illustrated in Figure 1. Note that the constraints did not have to form a closed region.

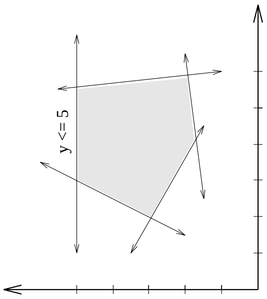

Figure 1: A Convex Polygon as a Conjunction of Constraints

The R^*-Tree index is a popular multidimensional index structure and can be used to store multidimensional point datasets (as well as non-point data). It is used in several extensible and GIS-oriented database systems. Boxes that are orthogonal to the axes are the standard approximation for data objects and queries in such indexes. However, a box is often a poor approximation, especially for queries and data that are irregularly shaped and not suitably oriented with respect to the axes.

An alternative is to use a conjunction of constraints to describe the bounding envelope of data objects and queries. Storing conjunctions of linear constraints can be expensive in terms of space; we do not consider this approach. We do propose, however, that queries be approximated by such conjunctions; the only overhead is a more complex overlap test, which we demonstrate can be carried out very efficiently [?].

To illustrate the kinds of queries that can be supported with such index structures, let us consider a portion of the Compustat stock and business data set with attributes *year number, earnings per share, stock low, stock high,* and *stock value at year close.* One example query using constraints is: *Retrieve all tuples that have a price-earnings (P/E) ratio less than some constant.* Investors frequently use the P/E ratio as a measure of how undervalued or overvalued a company is. In terms of constraints, the above query can be expressed as: $price_{year-end} = r * earnings \leq 0$ (where r is the ratio value). As another example, consider: *Retrieve all tuples that correspond to companies whose value didn't vary by more than some percent of the price at year's end.* In terms of constraints: $price_{high} - price_{low} - p * price_{year-end} \leq 0$ (where p is the percentage above). This measure is indicative of a stock's *volatility.*

It is easy to identify similar examples over virtually any multidimensional dataset of interest. For example, if we have an environmental dataset with attributes *time, latitude, longitude, height, radiation-level1, radiation-level2,* we can ask queries like the following: "Find observations such that *radiation-level1* is greater than *radiation-level2.*" "Find observations such that $radiation-level1 * radiation-level2 < 1$."

We have developed an extension to the R^*-Tree index that can support multidimensional constraint queries very efficiently, and implemented a package based upon this extension. We intend to use this both within the visual query processing engine of DEVise (to index graphical data) and to provide constraint query capabilities in the data transformation engine.

1.4 Research Goals

In this section, we motivate and describe how we propose to enhance and apply the tools and software mentioned above.

Visual Queries and DEVise

Visual browsing or **visual querying** of the data can be an extremely powerful way to explore large datasets. Several properties of a visual browser are important for the applications that we target:

1. In addition to providing some convenient way of using common visual presentation 'templates' such as scatter plots or bar graphs, the system should provide a flexible way to construct a broad class of new templates so that each dataset can be viewed in the most natural manner for any given application. To view data, the user must describe how it is to be visually presented; this includes a choice of existing templates or construction of new templates, and a specification of how the data records are to be mapped into a visual presentation based on these templates. We refer to the combination of templates and mappings used to describe a visual presentation as a *visual query.*

2. The definition of a 'visual query' should be cleanly separated from the actual dataset, so that it can be applied to other similar datasets, and the query definition facilities should be very flexible, in order to allow users to 'see' their data in as many different ways as possible.

3. The browser should be capable of simultaneously presenting multiple views of one or more datasets, in order to allow effective browsing of datasets with complex inter-relationships. Users should be able to define *links* between different views so that all related views change automatically in natural ways when one view is manipulated through a visual operation (e.g., a cursor operation, zooming, scrolling, etc.).

4. The browser should be capable of dealing with very large datasets, even if the datasets are several times larger than available memory sizes.

5. The browser should be capable of presenting the data at multiple levels of detail, in order to support effective interactive browsing of very large datasets. Typically, users want to see the data at a coarse level of detail, and then be able to look at interesting portions in more detail. The familiar operation of *zooming* is just one instance of this paradigm; more generally, there are several levels of summary data, or metadata, and users want to interleave browsing of the metadata and browsing of relevant portions of the actual data as needed.

Of the currently available tools for visual data exploration, we believe that DEVise [?] comes the closest to meeting these goals. DEVise was developed by us with prior NASA support and is currently in use by several groups (e.g., the business school, scientists in soil science, the NMR data center, the UW clinics, a catalog merchant; under consideration by IBM for use with their Data Mining product, and by LBL Laboratories).

Extending DEVise

There are several areas in which DEVise needs to be enhanced. We intend to work on the following tasks, which are directly related to the visualization capabilities of the tool or the overall system architecture:

1. Separate query definitions from details of the underlying data sets. This will make DEVise a powerful tool for *collaborative* computing: a user can navigate the data, and other connected users will see the same data display simultaneously, including all the changes performed by the original user. Since *each* user has the ability to do such navigation, in effect each user sees an *active* data presentation.

2. Improve various implementation-level aspects of DEVise, e.g., 3D symbols, color management, network connections, etc.

3. Build 'drivers' for a variety of data sources so that DEVise can access these data sources. (We have drivers for text files, SEQ data streams, Web URLs, Compustat data, etc., but not for NetCDF, for example).

Extensions related to other aspects such as integrated data and metadata browsing and ad-hoc querying are described in Section 1.4 and **??**.

Integrated Access to Data and Metadata

Visualization of very large datasets presents several challenges. A basic problem is that the dataset size may exceed the amount of main memory available. This problem has been addressed in DEVise by incorporating a *buffer manager*, which brings in data records as needed, if necessary making space in the buffer pool (an area of main memory allocated for this purpose) by discarding records that are not currently needed. Even with intelligent buffer management policies, interactive response times cannot be achieved for very large datasets, and too much information is lost by compressing a very large volume of data onto a single screen. A powerful paradigm for addressing this fundamental problem is to let users create summaries of data (which are typically much smaller than the original dataset) and to browse the summaries, or metadata, to get an overview of the entire dataset. Subsequently, users can look at interesting portions of the data in more detail; in our experience, users in fact want to be able to interleave the browsing of data and metadata.

As a concrete example, we have demonstrated the feasibility of integrating the BIRCH clustering tool with DEVise [?]. To illustrate the combined use of BIRCH and Devise, we present an actual application involving the filtering of forestry images. We were given two similar images of trees with a partly cloudy sky as the background, taken at two different wavelengths. One is in the near-infrared band (NIR), and the other is in visible wavelength band (VIS). Each image contains 512x1024 pixels, and each pixel has a pair of brightness values corresponding to NIR and VIS. Soil scientists receive hundreds of such image pairs and try to first filter the trees from the background, and then filter the trees into sunlit leaves, shadows and branches for statistical analysis.

We applied BIRCH to the (NIR,VIS) value pairs for all pixels in an image (512x1024 2-d tuples) by using 400 kbytes of memory (about 5% of the dataset size) and 80 kbytes of disk space (about 20% of the memory size), and weighting NIR and VIS values equally. We obtained 5 clusters that correspond to (1) very bright part of sky, (2) ordinary part of sky, (3) clouds, (4) sunlit leaves (5) tree branches and shadows on the trees. This step took 284 seconds.

However the branches and shadows were too similar to be distinguished from each other, although we could separate them from the other cluster categories. So we extracted the part of the data corresponding to (5) (146707 2-d tuples) and used BIRCH again. But this time, (1) NIR was weighted 10 times heavier than VIS because we observed that branches and shadows were easier to tell apart from the NIR image than from the VIS image; (2) BIRCH was able to draw finer distinctions because it processed a smaller dataset with the same amount of memory. The two clusters corresponding to branches and shadows were obtained in 71 seconds.

The result of clustering using BIRCH is shown in Figure DEVise-1 (in the appendix). Visually, we can see that it is a satisfactory filtering of the original image according to the user's intention. While this was a very successful effort in terms of the clustering results that we obtained, the interaction between BIRCH (to do clustering) and DEVise (to display the results) was tedious.

The important point to be noted in this example of using BIRCH and DEVise together is that the clusters produced by BIRCH can be seen as a summary of the original data: these clusters can thus be regarded as a form of *metadata*, providing a compact, high-level description. This kind of summary is by no means the only useful one, but it is a very good canonical example. In our interaction with users of BIRCH and DEVise, it has become very clear that users want to be able to cluster a dataset with a particular choice of clustering parameters (e.g., weights for the different dimensions, choice of distance metric), see the clusters in a visual presentation, then select all records in the *original data* corresponding to one or more clusters and process these records further. In other words, they use the clusters produced by BIRCH to obtain a high-level overview of the data, and narrow the scope of subsequent detailed analysis of interesting portions of the data.

Extending DEVise to Handle Integrated Data/Metadata Browsing

The clusters produced by BIRCH are only one example of a summary description of data. Other examples of summaries include:

1. Statistical measures over subsets of the data. Indeed, such summaries are so useful that some support is built directly into the current version of DEVise.

2. Compressed versions of images.

3. Moving averages over a time-series, with the window size controlling the degree of summarization. Such a summary could be generated using the SEQ tool, for instance.

4. The result of an SQL Groupby query, which summarizes all records with the same value in the grouped field into a single record (using statistical functions like *sum* or *average*).

5. Patterns extracted by data mining techniques for finding association rules [] or sequential patterns [].

As an example of integrated data/metadata browsing using some of these other kinds of summaries for metadata, we can obtain a monthly average of a series of daily temperature readings, using the SEQ