Data Visualization for Supporting Query-Based Data Mining

Kentarou Kichiyoshi, Hidehiko Iwasa, Haruo Takemura and Naokazu Yokoya

Graduate School of Information Science, Nara Institute of Science and Technology
8916-5 Takayama, Ikoma, Nara, 630-0101 Japan
{kentar-k, iwasa, takemura, yokoya}@is.aist-nara.ac.jp

ABSTRACT

This paper proposes a methodology for supporting the process of a query-based data mining by using visualization techniques. The query-based data mining is one of the important tasks of Knowledge Discovery in Databases (KDD). In the process of a query-based data mining, users hypothesize about patterns in a database and make a query to confirm the hypothesis. The proposed method supports the two aspects of the process, i.e., proposing an initial hypothesis as a query and modifying the hypothesis based on the query result. In the method, an instance in a database which has several attributes with numerical or nominal values is visualized as a color bar with several color parts which correspond to attribute values. Values of a function which evaluates the utility of a hypothesis are also visualized by using colors. These visualization technique helps users find an initial hypothesis and modify the hypothesis in order to increase the usefulness of it interactively. Experimental results show that the proposed method really helps a user find interesting rules in real world databases.

1 INTRODUCTION

An informal definition of Knowledge Discovery in Databases (KDD) is to find useful and interesting patterns in data. Data mining is one of the tasks of KDD and is defined as a method to find a part of data which has interesting common features and to acquire the description of the characteristics of the data [5, 6, 8, 19, 25].

Most of data mining methods that have been proposed to achieve the task try to find interesting patterns in databases automatically. In these methods, several functions that evaluate the usefulness score of patterns are employed and the results of the methods strongly depend on such evaluation functions.

To design appropriate evaluation functions is not an easy task. In [1], two functions called *support* and *confidence* are used to find association rules. *Support* represents the wideness of a rule; the percentage of instances that satisfy the rule in a database. *Confidence* indicates the strength of a rule; the percentage of instances that satisfy the rule within the instances that satisfy the condition part of the association rule. Since these functions are not domain specific, they are widely used in many data mining tools [4, 9, 10, 11, 12, 18, 21, 22, 24].

However, these general purpose evaluation functions are not always useful in the scene of the data mining. Data mining systems using these functions often discover trivial patterns in databases that are not interesting for users of the system, simply because users of the database have already known these rules that widely and strongly appear, especially when users are specialist of the contents of the database.

To avoid the problem above, query-based data mining systems have been proposed[2]. Users of a query-based data mining system give a query to the database system as a hypothesis of useful patterns in the database. The database system returns a data set which satisfies the given query conditions to users, and users confirm that how strongly and widely the hypothetical pattern exists in the database. If the strength or generality of an initial hypothesis is not sufficient, users modify the query and give it to the database system again.

In the query-based data mining, it is important to support users to have a good inspiration of initial queries and to modify a query adequately so as to increase the utility of the hypothesis based on the replied data set. To make an initial query as a candidate of a useful pattern, users have to know which part of the given data is dense, or which of attributes are correlate closely with each other. If users can grasp the distribution of data at a glance, users can find interesting patterns with taking into account the background knowledge such as the meaning of attributes. However, most of user interfaces of existing database systems are character-based, it is almost impossible for users to grasp the distribution of data at a glance.

This paper proposes a methodology for supporting the process of the query-based data mining[2] by using visualization techniques[3, 13, 17, 20, 26]. In the method, an instance in a database is represented as a bar with multiple colors and each color represents the value of a corresponding attribute defined in a database. Users can grasp the distribution of data at a glance by using this visualization method. Modifications of the generated hypothesis are carried out in two ways by changing ranges of attribute values in the query and by adding a new attribute to the query. Therefore, it becomes important to grasp the distribution of data in a multi-dimensional data space. The color bar representation is also used to help users understand how the utility score of the descrip-

0-7803-5731-0/99/$10.00 ©1999 IEEE

tion changes by relaxing or strengthening conditions which represent the present query. Histograms of attribute values are also used to find attributes which have strong correlation with other attributes included in the query.

This paper is structured as follows. In the next section, details of the visualization techniques are described. In Section 3, demonstration of the proposed method with a real world data set is shown and the usefulness of the proposed method is confirmed. Section 4 will conclude the paper with discussions about current problems of the system and future works.

2 VISUALIZATION METHODS FOR SUPPORTING QUERY-BASED DATA MINING

2.1 Query Representation

In this paper, we treat databases which have predefined attributes that take nominal or numerical values. For nominal attributes, possible values for the attribute are also given. To make the system simple, we employ a simple query representation, so called range query[7]. In the range query, users of database system specify the possible range for a numerical attributes and specify the value for a nominal attribute. If a user want to get a set of data which have values between 100 and 150 for an attribute A_1, the user gives an query $(100 \leq A_1 \leq 150)$ to the database. Users can make complex range queries by using AND and OR operators and nested queries are permitted such as $(GENDER = MALE) AND ((160 \leq HEIGHT \leq 180) OR (18 \leq AGE))$.

2.2 Visualization of Database for Finding Initial Queries

In the first stage of query-base data mining, users want to have an inspiration of useful patterns in a database. To have a good inspiration, it is important to understand distributions of data. Since users of database know the meaning of attributes, they can easily find candidates of useful patterns if they can grasp correlations among values of attributes at a glance.

To help users grasp distributions of values, values of attributes are transformed into colors in our visualization method. The Munsell color system[23] is employed for the transformation. Numerical values are normalized and mapped to the corresponding colors on the color wheel from *purple*(value=0) to *blue* (value=1) as shown in Fig. 1. In the case of nominal attributes, the color wheel is divided into arcs and colors at the border between arcs are assigned to each value. After the transformation, an instance in the database is visualized as a color bar with multiple colors that represent the values of attributes. Figure 2 illustrates the transformation of a database which has four attributes. In this case, an instance of the database becomes a columnar bar with four colors.

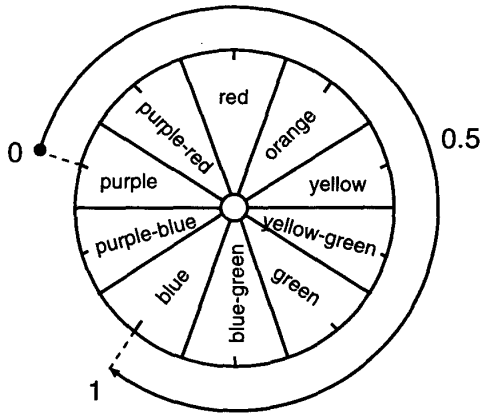


Figure 1: Transformation of values into colors using Munsell color system.

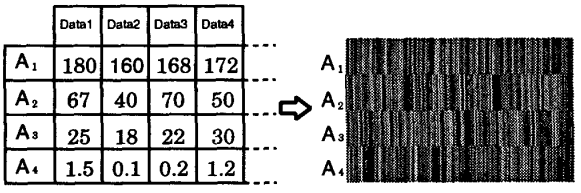


Figure 2: Visualization of a database with transforming numerical values into colors.

With this visualized database, users can grasp the distribution of values interactively by sorting the values of each attribute. In Fig. 3, data are sorted based on the values of the attribute A_1. As can be seen, there exists a region with similar colors in the middle part of the attribute A_2, i.e., two attributes A_1 and A_2 may have correlation in the region. If the range of A_1 where the colors of A_2 are similar is $[a_{1,1}, a_{1,2}]$, the query can be specified by $(a_{1,1} \leq A_1 \leq a_{1,2}) AND (a_{2,1} \leq A2 \leq a_{2,2})$, where $a_{2,1}$ and $a_{2,2}$ are the maximum and minimum values for attribute A_2 in the range of data.

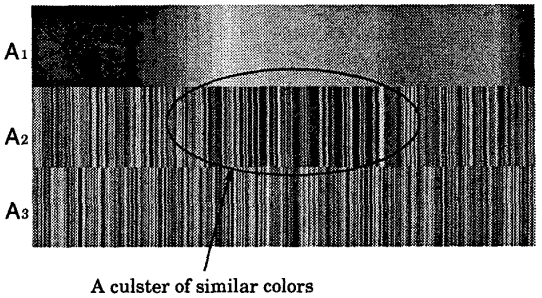


Figure 3: Finding a similar color region with the visualized database.

2.3 Finding Additional Attributes with Histograms of Attribute Values

Another promising way of finding attributes which have correlations with other attributes is to investigate distributions of values directly using histograms of values. Users can grasp the distribution of values of an attribute. Especially, histograms are useful as a supplemental tool of visualized databases. When a user find a color cluster in the visualized database, the density of instances can be confirmed using the histogram of them.

In our method, two histograms of an attribute are superimposed, one is a histogram of all data and the ohter is that of data in the specified range. If there exists a mountain in the first histogram and the second histogram is flat, user can confirm that a cluster of instances exists in the specified region of the attribute and can add the attribute to the current query.

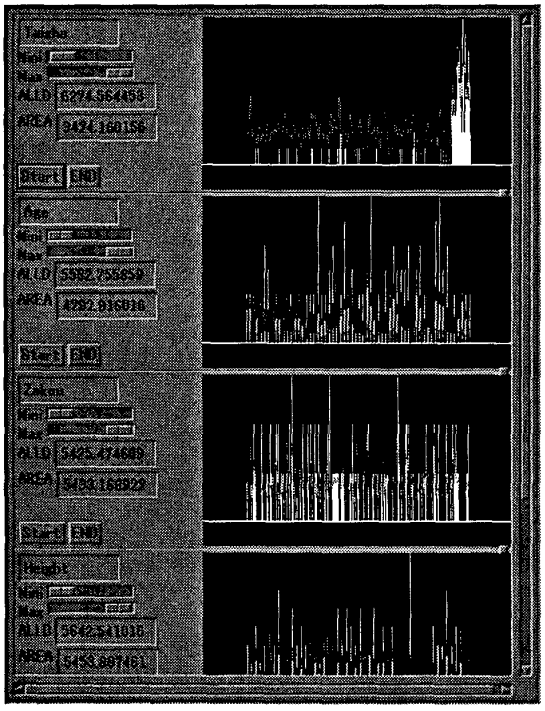


Figure 4: Superimposed two histograms of several attributes.

2.4 Visualization of Values of Evaluation Function for Modifying Query-Ranges

After users find an initial query by browsing the visualized database and histograms, the next step is to refine the query. The region of the initial query is roughly determined by directly specifying the area which has similar colors. Therefore, the range of specified area may be too wide or too narrow to appropriately capture the group of data which have similar values in the attributes. These mis-specifications of ranges decrease the strength or generality of the pattern. To increase the strength and generality of the patterns represented by a query, users have to narrow the range of the query as much as possible without decreasing the number of instances included in the query.

In order to navigate users to refine a query to satisfy the requirement, we employ two functions called *distance* and *relevance*[14, 15, 16].

Distance is a simple distance function between a specified range $[a1, a2]$ and a value x of an attribute defined as follows:

$$distance(x, [a_1, a_2]) = \begin{cases} x - a_2 & (x > a_2) \\ 0 & (a_1 \leq x \leq a_2) \\ x - a_1 & (x < a_1) \end{cases}$$

Relevance is a function that represents the distance between an instance and a query and is calculated by using *dinsance* values of attributes that appears in the query. Two *distance* values are integrated into *relevance* in the following manner according to the connection operators:

$$relevance(d1, d2) = \begin{cases} d1 + d2 & \text{in the case of AND} \\ d1 \times d2 & \text{in the case of OR} \end{cases}$$

In order to support users to change ranges of attributes to increase the strength and generality of the query, values of *distance* and *relevance* are transformed into colors in the same manner as values of data are transformed. Values of *distance* and *relevance* are normalized to $[-1, 1]$. In the Munsell color wheel, *purple*, *yellow* and *blue* are assigned to -1, 0 and 1, respectively. Figure 5 shows an example of visualized *relevance* and *distance* values with a query $(a_{1,1} \leq A_1 \leq a_{1,2}) AND (a_{2,1} \leq A_2 \leq a_{2,2})$.

With the visualized *relevance* and *distance* values, users can understand how the strength and generality of the query change by modifying the ranges. In Fig. 5, instances are sorted based on *relevance* values. There exist instances whose values for A_1 are in the range of the query (*yellow*) but values for A_2 are not. The color of *distance* values of these instances in A_2 are yellow-green, i.e., the distance between the specified range and the attribute value of each instance is short. This means that the generality of the query, i.e., the number of instances in the query ranges, will increase by slightly moving the upper limit of A_2 to widen the range. On the other hand, if there exist instances whose *disntance* colors of A_1 are *yellow* and colors of A_2 are not close to *yellow*, the range for A_1 may too wide. Users have to investigate the appropriate range for A_1 by sorting data based on *distance* values of A_1.

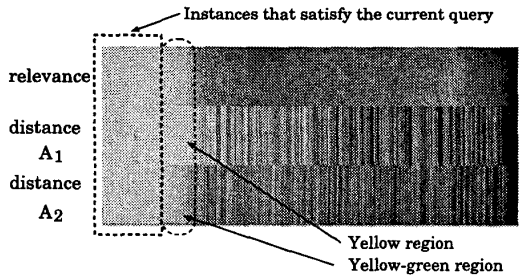


Figure 5: Visualization of distance and relevance values.

Table 1: Eight attributes in *boston* dataset.

Attribute name	Meaning
CRIM	per capita crime rate by town
ZN	proportion of residential land zoned for lots over 25,000 sq.ft.
INDUS	proportion of non-retail business acres per town
NOX	nitric oxides concentration (parts per 10 million)
RM	average number of rooms per dwelling
AGE	proportion of owner-occupied units built prior to 1940
PTRATIO	pupil-teacher ratio by town
LSTAT	% lower status of the population

3 DEMONSTRATION OF QUERY-BASED DATA MINING WITH THE PROPOSED VISUALIZATION METHODS

3.1 Data specification

To demonstrate the proposed visualization methods in the process of the query-based data mining, a real world dataset called *boston* which contains information collected by the U.S Census Service concerning housing in the area of Boston Mass. It was obtained from the StatLib archive (http://lib.stat.cmu.edu/datasets/boston). The dataset includes 506 instances and has 14 attributes. We use eight of them as shown in Table 1 to reduce the size of figures in this paper. The following demonstrates the proposed method with a sequence of example queries.

3.2 Demonstration of Query-Based Data Mining with *boston* dataset

In the demonstration, it is assumed that a user want to know the property of exclusive residential districts. Therefore, the user firstly sets the range for the attribute **ZN**. In our visualization method, the user can set the range by sorting instances with **ZN** values and select a region in which instances have relatively high **ZN** values as shown in Fig. 6. In this case, the user

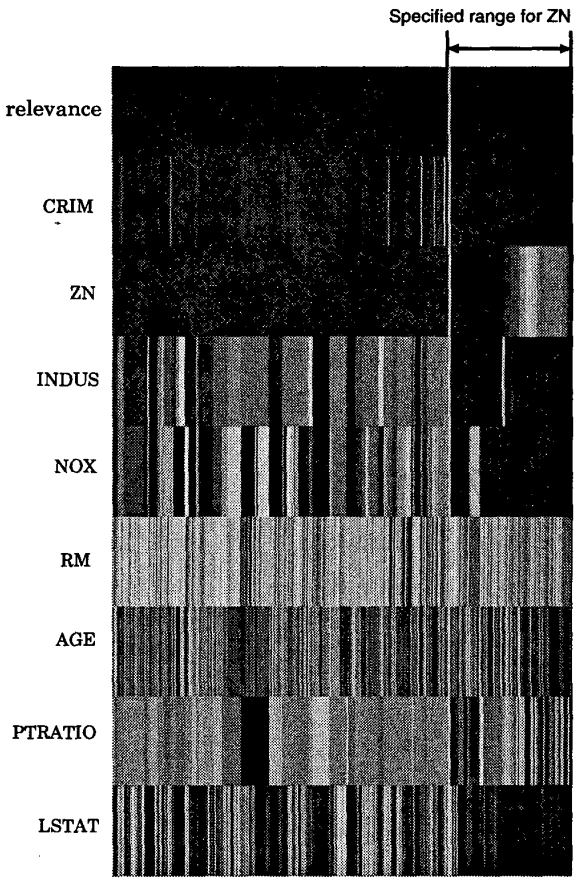


Figure 6: Visualized database with eight attributes sorted by **ZN**.

set the range as $(12.2 \leq ZN \leq 100)$.

In regions of other attributes corresponding to $(12 \leq ZN \leq 100)$, the user can find clusters of colors in attributes **CRIM**, **INDUS**, **NOX** and **LSTAT**. The user can easily set the ranges for these attributes by using their histograms. On the other hand, in attributes **RM**, **AGE** and **PTRATIO**, the user can not find any color clusters. Figure 7 shows histograms of these attributes. From these histograms, the user can confirm that the distribution of values in then specified range is not different from that of all instances. In this case, user can make a query: $(12.2 \leq ZN \leq 100)\ AND (0.0 \leq CRIM \leq 0.006)\ AND (0.46 \leq INDUS \leq 6.88) AND (0.39 \leq NOX \leq 0.47)\ AND (1.87 \leq LSTAT \leq 9.40)$. The percentage of instances that satisfy the query within all instances, i.e., *support* of the query, are 18, and the percentage of instances that satisfy the query within instances that are included in the range $(12.2 \leq ZN \leq 100)$, i.e., *confidence* of the query, are 69.

To increase the values of *support* and *confidence*

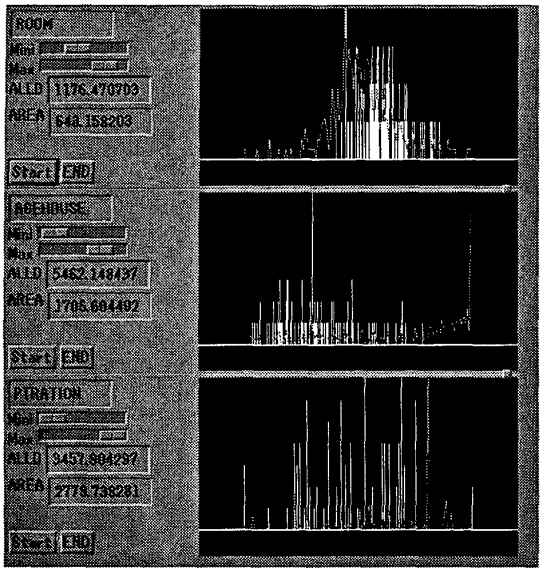


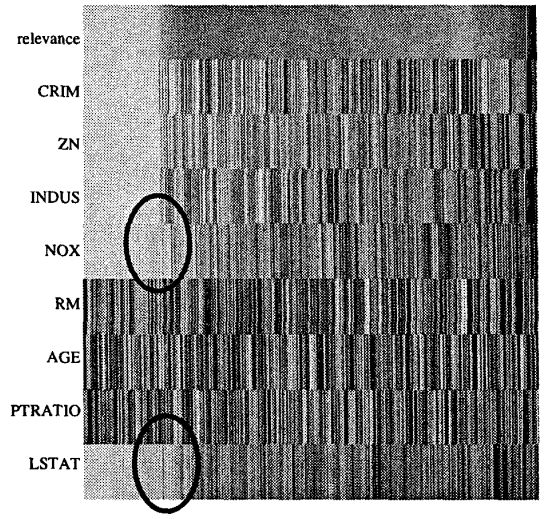
Figure 7: Histograms of **RM**, **AGE** and **PTRATIO**.

without losing the strength of the pattern, the user modified the range of the query by using visualized *relevance* and *distance* values. The user can find two *yellow-green* regions easily as shown in Fig.8(a). Figure 8(b) illustrates the way of widening the range of **NOX** as the range includes the yellow-green region.

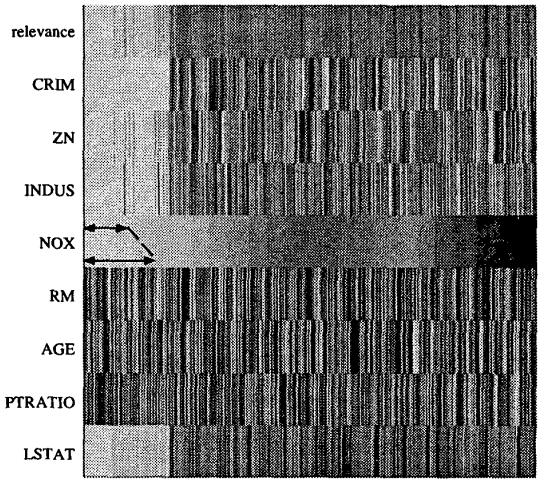
After the refinement, the query became $(12.2 \leq ZN \leq 100) AND (0.0 \leq CRIM \leq 0.006) AND (0.46 \leq INDUS \leq 6.88) AND (0.39 \leq NOX \leq 0.49) AND (1.87 \leq LSTAT \leq 9.62)$. As a result, *support* and *confidence* became 20 and 70, respectively. As demonstrated above, the user can find a strong pattern in the database only with his/her perceptual capability.

4 CONCLUSION

Data mining in very large databases is one of the most important challenges in the research area of databases. The task is to efficiently find interesting data sets, i.e., clusters of similar data or correlations between several parameters. Our approach to support the data mining process enhances the capability of traditional database querying by visualizing database itself and giving users visual feedbacks of queries. Since our method is independent of any specific domain area and requires no knowledge of statistics such as cluster analysis, users with perceptual capabilities and general knowledge are responsible for doing the analysis and interpretation. As we demonstrated the proposed method in Section 3, users of the system can explore databases by incrementally refining queries guided by the visualized database and visual feedbacks of previous queries. We will improve the method by extending the capability of the query language. The visualization techniques for



(a) Visualized values of *relevance* and *distance* sorted by values of *relevance* .



(b) Modification of the range of NOX

Figure 8: Modification of ranges based on visualized *relevance* and *distance* values.

displaying a numbers of attributes and data should be further investigated.

REFERENCES

[1] R. Agrawal, H. Mannila, R. Srikant, and H. Toivonen. Fast discovery of association rules. In *Advances in Knowledge Discovery and Data Mining*, chapter 12, pp. 307–328. AAAI/MIT Press, 1996.

[2] T. M. Anwar, H. W. beck, and S. B. Navathe. Knowledge mining by imprecise querying: A classification-based approach. In *Proceedings 8th International Conference on Data Engineering*, pp. 622–630, 1992.

[3] B. G. Becker. Using mineset for knowledge discovery. *IEEE Computer Graphics and Applications*, pp. 75–78, 1997.

[4] S. Brin, R. Motwani, J. D. Ullman, and S. Tsur. Beyond market baskets: Generalizing association rules to correlations. In *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data*, pp. 265–276, 1997.

[5] P. Cheeseman and J. Stutz. Bayesian classification (autoclass) : Theory and results. In *Advances in Knowledge Discovery and Data Mining*, chapter 6, pp. 153–180. AAAI/MIT Press, 1996.

[6] M. Chen, J. Han, and P. S. Yu. Data mining: An overview from a database perspective. *IEEE Transactions on Knowledge and Data Engeneering*, Vol. 8, No. 6, pp. 866–881, 1996.

[7] C. Faloutsos and K. I. Lin. FastMap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets. In *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, pp. 163–174, 1995.

[8] U. M. Fayyad, G. P-Shapiro, and P. Smith. From data mining to knowledge discovery: An overview. In *Advances in Knowledge Discovery and Data Mining*, chapter 1, pp. 1–34. AAAI/MIT Press, 1996.

[9] T. Fukuda, Y. Morimoto, and S. Morishita. Constructing efficient decision trees by using optimized numeric association rules. In *Proceeding of the 22nd VLDB Conference*, pp. 146–155, 1996.

[10] T. Fukuda, Y. Morimoto, S. Morishita, and T. Tokuyama. Data mining using two-dimensional optimized association rules: Scheme, algorithms and visualization. In *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, pp. 13–23, 1996.

[11] Takeshi Fukuda, Yasuhiko Morimoto, Shinichi Morishita, and Takeshi Tokuyama. Mining optimized association rules for numeric attributes. In *Proceedings of the 15th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pp. 182–191, 1996.

[12] J. Han and Y. Fu. Discovery of multiple-level association rules from large database. In *Proceedings of the 21st VLDB Conference*, pp. 420–431, 1995.

[13] Y. Iizuka, H. Shiohara, T. Iizukam, and S. Isobe. Automatic visualization method for visual data mining. In *Research and Development in Knowledge Discovery and Data Mining*, pp. 174–185. Springer, 1998.

[14] D. A. Keim and H.-P. Kriegel. VisDB: Database exploration using multidimensional visualization. *IEEE Computer Graphics and Applications*, pp. 40–49, 1994.

[15] D. A. Keim and H.-P. Kriegel. Visualization techniques for mining large databases: A comparison. *IEEE Transactions on Knowledge and Data Engeneering*, Vol. 8, No. 6, pp. 923–938, 1996.

[16] D. A. Keim, H.-P. Kriegel, and M. Ankerst. Recursive pattern: A technique for visualizing very large amounts of data. In *Proceedings of Visualization '95*, pp. 279–286, 1995.

[17] D. A. Keim, H.-P. Kriegel, and T. Seidl. Supporting data mining of large databases by visual feedback queries. In *IEEE 10th International Conference on Data Engeneering*, pp. 302–313, 1994.

[18] W. Klosgen. Explora: A multipattern and multistrategy discovery assistant. In *Advances in Knowledge Discovery and Data Mining*, chapter 10, pp. 249–271. AAAI/MIT Press, 1996.

[19] W. Klosgen and J. M. Zytkow. Knowledge discovery in databases terminology. In *Advances in Knowledge Discovery and Data Mining*, chapter A, pp. 573–592. AAAI/MIT Press, 1996.

[20] H. Y. Lee and H. L. Ong. Visualization support for data mining. *IEEE Expert Intelligent Systems and Their Application*, Vol. 11, No. 5, pp. 69–75, 1996.

[21] B. Lent, A. Swami, and J. Widom. Clustering association rules. In *Proceedings of the 13rd International Conference on Data Engeneering*, pp. 220–231, 1997.

[22] Ramakrishnan Srikant Rekesh Agrawal. Mining sequential patterns. In *Proceedings of the 11st International Conference on Data Engeneering*, pp. 3–14, 1995.

[23] S. J. Sangwine and R. E. N. Horne. *The Colour Image Processing Handbook*. CHAPMAN&HALL, 1998.

[24] R. Srikant and R. Agrawal. Mining generalized association rules. In *Proceedings of the 21st VLDB Conference*, pp. 407–419, 1995.

[25] R. Uthurusamy. From data mining to knowledge discovery: Current challenges and future directions. In *Advances in Knowledge Discovery and Data Mining*, chapter 23, pp. 561–569. AAAI/MIT Press, 1996.

[26] P. C. Wong and R. D. Bergeron. 30 years of multidimensional multivariate visualization. In *Scientific Visualization: Overviews, Methodologies, and Techniques*. IEEE Computer Society Press, 1997.