Similarity Search and Data Mining: Database Techniques Supporting Next Decade's Applications

Christian Böhm
Unit for Database Systems, University for Health Informatics and Technology
Innrain 98, 6020 Innsbruck, Austria
Christian.Boehm@umit.at – www.ChrBoehm.de
Phone&Fax: +49-700-24726346 (+49-700-ChrBoehm)

Abstract

Similarity Search and Data Mining have become widespread problems of modern database applications involving complex objects such as Multimedia, CAD, Molecular Biology, Sequence Analysis, etc. Search problems in such databases are rarely based on exact matches but rather on some application specific notion of similarity. A common approach to grasp the intuitive idea of similarity by a formal means is to translate complex objects into multidimensional vectors by a feature transformation which allows retrieval of the most similar objects to a given query object (similarity search) but also to analyze the complete set of complex objects with respect to clusters, outliers, correlations etc. (data mining). In this contribution we identify several areas of applications where the classical feature approach is not sufficient. Example applications include Biometric Identification, Medical Imaging, Electronic Commerce and Share Price Analysis. We show that existing feature based similarity models fail due to different reasons, e.g. because they do not cope with the uncertainty which is inherent to their feature vectors (biometric identification) or because they do not integrate application specific methods into the similarity model (share price analysis, medical imaging). We survey the challenges and possible solutions to these problems to direct future research.

1. Introduction

Information is the master key to economic success and influence in the contemporary information society. *"Only who can apply the newest information for his product development, is able to survive in the global competition"* [Sch 95]. Crucial for the applicability of information is its quality and its fast availability. What is lacking most, however, is not the access to information resources but rather the facility to effectively and efficiently search for the required information. To cope with the *information overkill* will be the central competence of the next decade. Therefore, research on information systems will be one of the most important domains of computer science. To have connection with the worldwide leading group in this area will not only be of vital interest for academic scholars but even for the complete worldwide economy, in all branches of traditional and "new" economy.

If the structure of the information to be searched is sufficiently simple, such as in one-dimensional numerical attributes or character strings, search problems can be considered as solved. Database management systems (DBMS) provide index structures for the management of such data [BM 77, Com 79] which are well-understood and widely applied. Requirements of traditional applications such as accounting and billing are perfectly met by a commercial DBMS. Therefore, the information infrastructure of most enterprises is based on products such as Oracle or Informix.

Recently, an increasing number of applications has emerged processing large amounts of complex, application-specific data objects [Jag 91, GM 93, FBF+ 94, FRM 94, ALSS 95, KSF+ 96]. In application domains such as multimedia, medical imaging, molecular biology, computer aided design, marketing and purchasing assistance, etc., a high efficiency of query processing is crucial due to the immense and even increasing size of current databases. The search in such databases, called non-standard databases, is rarely based on an exact match of objects. Instead, the search is often based on some notion of similarity which is specific to the application.

For applications which do not only support transaction oriented search operations but also high-level decision making, it is necessary not only to search for objects which are similar to a given query object but rather to analyze the data set as a whole. Information which is interesting in the process of decision making are common patterns, classifications of data, knowledge about clusters (collections similar of objects), and, as the opposite of a cluster, exceptional data (outliers) which can be, for instance, indicators for the misuse of a system. This kind of information is commonly referred to as *knowledge* and the process of deriving such higher-level information from low-level transactional data is called *data mining* or (in the presence of a vast amount of data) *knowledge discovery in databases* (*KDD*). Because such applica-

tions on top of modern databases are also often based on some notion of similarity (or equivalently on the notion of *data density*) they also depend on similarity search. The difference to traditional similarity search applications is, however, that these applications do not only raise few, single similarity queries but rather a high number of such queries.

Our main focus is on efficiency. Many problems of similarity search and data mining have been basically solved, and algorithms have been proposed that produce useful results. A general problem, however, are the vast amounts of data of today's applications, and a processing time that makes the algorithms inoperative. Our general motivation is to change such algorithms, base them on powerful database primitive operations such that thy scale well to large data sets, potentially not fitting into the main memory. Thus, the database system becomes a powerful toolset to support the next decade's applications.

A typical approach to handle complex objects of modern applications such as CAD, multimedia, etc. is the feature transformation. This transformation extracts from the objects a number of characterizing properties and translates them into vectors of a multidimensional space. The specific property of a feature transformation is that the similarity of the objects corresponds to a small distance (in most cases measured according to the Euclidean norm) of the associated feature vectors. Therefore, similarity search systems and data mining algorithms rely on distance based queries (*similarity queries*) on the feature vectors (cf. figure 1).

The remainder of this paper is organized as follows: Section 2 describes previous work in the areas of similarity search and data mining. Section 3 is dedicated to indicate promising directions of future research, and section 4 summarizes and concludes the paper.

2. Previous Work

Our previous work primarily concentrates on indexing of feature transformed data from modern database applications such as CAD, multimedia, molecular biology, time sequences etc. and on advanced query processing techniques for applications of data analysis and data mining.

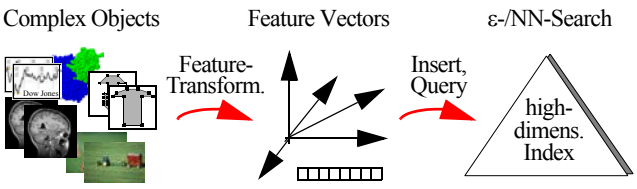

Figure 1. Basic Idea of Feature-based Similarity Search

2.1 Similarity Search

Index Structures

First we considered single similarity queries and their support by multidimensional index structures. Starting from existing approaches for high-dimensional spaces such as the X-tree [BKK 96], the TV-tree [LJF 95], or the SS-tree [WJ 96] our observation was that for sufficiently high dimensions, the complexity of similarity queries is still far away from being logarithmic. Moreover, simple query processing techniques based on the sequential scan [WSB 98] of the data set are often able to outperform approaches based on sophisticated index structures. The deterioration of performance for increasing dimensions is often called the *curse of dimensionality*. For previous approaches on indexing high-dimensional data spaces, cf. also our computing survey [BBK 02].

Our primary intention was to develop index structures that solve the problems of the existing approaches and outperform them clearly. This could be achieved with the pyramid technique [BBK 98b] for a limited kind of queries. Another important approach was the parallelization of index structures for high-dimensional data spaces. A declustering technique for distributed and parallel environments [BBB+ 97] has gained the best paper award 1997 of the ACM SIGMOD Int. Conf. on management of data.

A few techniques primarily concentrating on the efficient construction of multidimensional index structures are proposed in [BBK 98a] and [BK 99]. Problems of the integration of multidimensional index structures into object-relational environments were considered in [BBKM 99] and [BBKM 00] and with the focus on objects with a spatial extension which are more prevalent in geographical databases in [BKK 99].

Cost Modelling and Optimization

As it was not possible to develop index structures which are not subject to the curse of dimensionality and which yield a uniformly good performance over all dimensions, we analyzed index based query processing from a theoretical point of view and proposed a cost model [BBKK 97] which was later significantly extended [Böh 00]. The central notion of the cost model is the access probability of an index page for an arbitrary query. The mathematical concept to capture this probability was the Minkowski sum (cf. figure 2), a concept primarily used in robot motion planning which was introduced in this work for the first time to cost modelling. Together with techniques for estimating the extension of index pages and queries in the data space, this concept could be used to accurately estimate the disk access and CPU cost of similarity queries under Euclidean and maximum metrics in low-dimensional data spaces. Two extensions accounting for

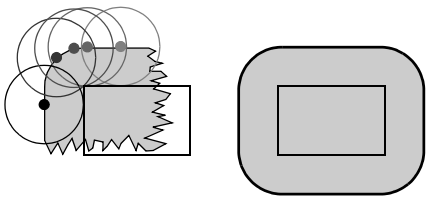

Figure 2. The Minkowski Sum

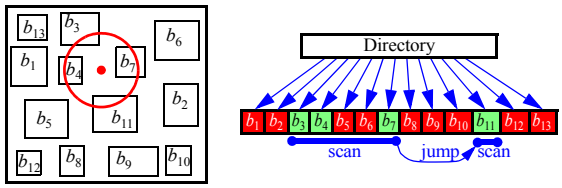

Figure 3. The Fast Index Scan of the IQ-Tree

boundary effects and *non-linear correlations* (described by fractal concepts) enabled our cost model also to accurately predict the performance of index structures constructed upon high-dimensional spaces.

We extensively used the cost model to optimize high-dimensional index structures. The most limiting drawback of previous index structures for similarity search was the possibility of being outperformed by the simplest query processing technique, the sequential scan, which considers all data objects. In [BK 00], we could identify the problem, that usual index structures access data in too small portions. This problem could be eliminated by a suitable optimization of the logical block size of the index structure. As the blocksize optimum is dependent on characteristics of the data set which are not known in advance and which are subject to changes over time, we proposed an index structure which adapts the blocksize dynamically during update operations. The result was an index structure which is always guaranteed to outperform both approaches, the classical (unoptimized) index and the sequential scan. For the extreme situation of a very high data space dimension (100d), the performance of the optimized index converges to the sequential scan. In the other extreme case of a quite low data space dimension (2d, 3d), the performance converges to that of a common index. For the very interesting case of dimensions in a middle range (5d-20d), the optimum is also in a middle range (between 64KBytes and 1MByte) and the resulting index outperforms both previous approaches by high factors.

A similar kind of optimization was proposed in [BBJ+ 00] where not the size of the stored blocks was optimized but the number and order of blocks which are processed in a single chained I/O operation (cf. figure 3). In the extreme cases, this *fast index scan* converges in a similar way as above either to classical processing of small units as in traditional indexes (if the data space dimension is small), or to the sequential scan if the data space dimension is large. The size of the units to be processed is also optimized by a probability model which is derived from our cost model [BBKK 97].

In our tree striping technique [BBK+ 00], our cost model was also applied. It was shown that for high-dimensional query processing it is not optimal to use a single, high-dimensional index structure. Instead, data vectors can be cut into smaller pieces which can be stored in separate indexes. The extreme case of storing each attribute in a single, one-dimensional index is called *inverted list approach*. We have shown, however, that the inverted list approach suffers from similar performance problems as high-dimensional indexes. But there exists often an optimum between the two extreme cases, inverted lists and single high-dimensional index which is exploited by our tree striping technique. Here, we construct a limited number of indexes of a mid-range dimensionality and, again, outperform the classical approaches. The optimum number and dimensionality can be determined by a modification of our cost model.

Our cost model also influenced other research groups in their work. Weber, Schek and Blott compared in a performance study [WSB 98] various query processing approaches and used our results to prove the general limitation of all indexing approaches for sufficiently high dimensional data spaces. As a consequence, they proposed a query processing technique based on the sequential scan which applies lossy data compression techniques to the feature vectors. In [BBJ+ 00], we extended their work such that a similar data compression technique can be integrated in index based query processing. As the choice of an optimal compression rate was out of the scope of [WSB 98] we applied a modification of our cost model for this purpose. Our results were also transferred into the research domain of metric index structures for which similar cost models have been devised [CPZ 98] and applied for optimization [TTSF 00]. Papadopoulos and Manolopoulos [PM 97] used our results for a novel declustering technique in parallel environments. Several papers such as [RGF 98] or [AGGR 98] apply the cost model in the area of data mining. Our cost model has been extensively cited by internationally recognized research groups.

2.2 Data Mining

Data mining algorithms in feature databases typically rely heavily on repeated iterative similarity queries. Therefore, all results described in section 2.1 can also be successfully applied to speed up such data mining algorithms. However, the fact that data mining algorithms analyze the whole data set and, therefore, raise a high number of similarity queries can be particularly taken into account. In [BBBK 00], we analyzed a high number of data mining algorithms, particularly the density based clustering algorithm DBSCAN [SEKX 98]

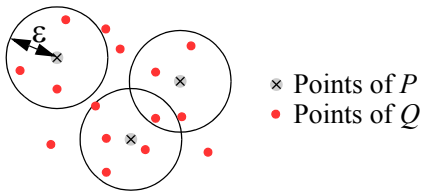

Figure 4. The Similarity Join

and the hierarchical cluster analysis method OPTICS [ABKS 99], which raise a similarity query for each object stored in the database. We showed that such algorithms can also be based on top of a database primitive called *similarity join*. The similarity join $P \bowtie_\varepsilon Q$ of two finite sets $P = \{p_1,..., p_n\}$ and $Q = \{q_1,..., q_m\}$ with a given distance parameter ε, is formally defined as the set (cf. figure 4)

$$P \bowtie_\varepsilon Q := \{(p_i,q_j) \mid \|p_i - q_j\| \le \varepsilon\}$$

The similarity join can also be expressed in a SQL like fashion:

SELECT * FROM *P*, *Q* **WHERE** ||*P*.vector – *Q*.vector|| ≤ ε.

We have shown that these algorithms, and a number of further data mining tasks such as outlier detection, simultaneous classification, and some other special data mining techniques can be transformed such that the result is guaranteed to be equivalent with the original algorithm. Our transformation, however, allows high performance improvements. It is possible to use the similarity join as a black box, i.e. the details of the actual algorithm which implements this basic operation, are not needed to be known to the implementor of the data mining algorithm. It is possible to select a suitable algorithm from a collection of different algorithms according to the performance requirements and the characteristics of the data set and depending which indexes are preconstructed.

Many algorithms for the similarity join are known, partly operating on index structures such as R-trees [BKS 93, HJR 97], or based on hashing [LR 96, PD 96] or sorting [SSA 97]. For a complete overview of previous and new techniques cf. our tutorial [Böh 01], and a survey which is currently in progress. Motivated by our applications, we have recently published original work in the area of similarity join algorithms. An algorithm based on a new kind of sort order, the epsilon-grid-order [BBKK 01], has been accepted at the ACM SIGMOD Int. Conf on Management of data. Several further articles regard cost modelling and optimization of the similarity join: In [BK 01a], we propose a cost model for the index based evaluation of the similarity join. The cost model accurately estimates the probability with which a pair of pages must be formed in the joining process (the *mating probability*). Starting from that point, we optimize the page capacity of an index for join processing. Our work reveals here a serious optimization conflict between CPU (where typically small pages with up to 20 points capacity are optimal) and disk I/O processing (where some 10,000 points are typically optimal). To solve this conflict, we propose a two-tier index architecture where large pages which are subject to I/O accommodate a secondary search structure which is optimized for the CPU operations.

In [BKK 01], we propose a generic technique for improving the CPU performance of similarity join algorithms. The underlying similarity join algorithm can be based on an index structure or on some hashing or sorting schema. The general idea is to avoid and/or accelerate the distance computations among feature vectors by *dimension sweeping*, an idea which is similar to the plane sweep paradigm of algorithms. As the order of dimensions which are subject to this sweeping technique is also optimized, we call our generic technique *optimal dimension sweeping*. In [BBKS 00], we proposed scheduling algorithms for a different kind of similarity join operation which is not based on a defined distance predicate, but combines each of the points from one data set with one or more of its nearest neighbors of the other data set.

3. Future Research Plan

Besides complementing and even strengthening our effort in the successful areas of database primitives for similarity search and data mining, we have identified several research directions into which we plan to extend our future work. This includes opening new, innovative application domains with new challenging research potential, a general framework for the development of similarity search systems, and database technology centered research.

3.1 New Data Mining Tasks

Due to a complex analysis of the complete data set data mining algorithms are often of a much higher computational complexity than traditional database applications. This has mainly prevented data mining tasks from being strongly integrated into the database environment. Our method of identifying very powerful database primitives such as the similarity join (or as another example, the convex hull operation, cf. [BK 01b]), data mining algorithms may become standard database applications like others. The consequence is a much tighter integration of data mining in the information infrastructure of an enterprise which yields many advantages.

Due to the dramatic increase of performance by our approaches, it will be possible to implement quite new kinds of data mining algorithms which detect new kinds of patterns. An interesting, new challenge is subspace clustering [AGGR 98]. Typically, not all attributes of feature vectors carry information which is useful in data mining. Other attributes may be noisy and should be ignored as they deteriorate the data mining result. Identifying the relevant attributes, however, is a difficult task. Subspace clustering combines the

two tasks of selecting attributes and finding clusters. Subspaces, i.e. groups of attributes, are determined such that maximal, distinguishable clusters can be found. First algorithms, however, suffer from the high computational cost. Basing them on top of powerful database primitives could open the potential to make this computation feasible.

Another approach could be to make current data mining much more interactive. The current process is to select parameters, to run a data mining algorithm and, finally, to visualize the result of the algorithm. Our dramatic performance gains could open the potential to make this process so fast that a user may change parameters and immediately see the resulting changes in the visualization. Here, it could be beneficial to apply new concepts in database systems which evaluate queries *approximately* [CP 00] or produce first results in an early stage of processing.

3.2 New Application Domains

We have identified three areas of new applications which have only superficially been considered as database applications, in spite of vast data amounts and clear relations to similarity search and data mining.

Electronic Commerce

Many stages in electronic commerce require concepts from similarity search and data mining. In the early stage, marketing, it is essential to perform a customer segmentation, a typical data mining application, to make directed offers to which the customers are maximum responsive.

In the core area of e-commerce, booking and sales systems, users specify their needs in an inexact way. For instance, they have initial ideas about features their product should have and the corresponding price. Then, in an interactive process the system has to find out which of the initial features are how relevant to the specific customer and will find in this way a product which fits best the users notions.

After commitment of the trade the final stage is marketing for additional products. A good (human) salesman develops a sense what additional high-revenue products could be of interest for the customer, based on his experience with previous customers purchasing similar products. This behavior could also be imitated using concepts of similarity search and data mining.

For the above mentioned applications, it is necessary to extend known concepts and to develop new concepts. Classical similarity search takes the basic assumption that the similarity measure is a parameter given by the user. Therefore, weights for the individual features are assumed to be known. Here, we are rather facing the situation that the measures are initially completely unknown and develop during the selection process. Instead of assuming a uniform importance of the features, and ranking the products according to the Euclidean distance, the user should be provided with a selection of products that reveals different weighting of the features. A selection with varying weights of features essentially corresponds to the *convex hull* of a subset of the data [BK 01b]. The products which are further investigated by the customers can be used for a *relevance feedback*, to determine a suitable similarity measure. A first approach to use relevance feedback for this purpose is the MindReader [ISF 98] which determines a quadratic from distance measure [BKS 01]. For electronic commerce, we identify two additional requirements. First, the relevance feedback should be extended to a multimodal model to take into account that users in general do not only like one single “ideal” product but often have a few alternatives in their minds which are not clearly separated in their notion. The second requirement is a seamless integration of the concepts of similarity search, convex hull, and relevance feedback.

Biometry Databases

Biometry applications store human data such as features from face images, fingerprints, the hand geometry, the retina, or even voice and handwriting for identification and authentification purposes. For many applications, a high number of feature vectors are stored and due to the inexactness of the measuring devices, similarity search is needed for the identification.

In contrast to traditional similarity search, the uncertainty of the individual features is not uniform among all features and even for a single feature, the uncertainty is not uniform among all stored vectors. Instead, each feature of each feature vector is associated with an individual uncertainty which is stored in the database. With this concept, it is possible to capture problems introduced by different environments and technical devices. The uncertainty of facial features such as the eye distance, for instance, depends on the angle between camera and person, and also on the illumination. The error can be assumed to be taken from a Gaussian distribution, so the uncertainty is measured in terms of a standard deviation.

The challenge here is to develop specialized index structures to store feature vectors with individual uncertainty vectors and query processing algorithms that facilitate a fast and efficient evaluation of queries such as

- determine all persons that match the query person with a probability of at least 10%
- determine the person that matches the query person with maximum probability.

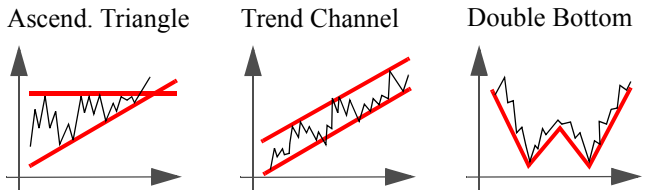

Figure 5. Chart Analysis

Technical Analysis of Share Price

One of the classical applications of similarity search and data mining is clearly the analysis of time sequences [ALSS 95] such as share price analysis. Various similarity measures have been proposed. For practical analysis, however, quite different concepts are used, such as indicators, i.e. mathematical formulas derived from the time sequence that generate trading signals (*buy, sell*). Another concept for the analysis of a time sequence is the chart analysis (cf. figure 5) which detects typical formations in the share price which are known to indicate a certain direction of the future price. Examples are triangles, trend channels, lines of support and resistance, W-formations (double bottom), head-and-shoulder-formations etc.

For effectively supporting users in their share price analysis, the requirement is to integrate both indicators as well as formation analysis into search systems and into sequence mining algorithms. Suitable index structures and query processing techniques must be developed to facilitate a fast and efficient analysis.

3.3 A Framework for the Development of Similarity Search Systems

The problem of similarity search should also be considered in a more general way. Currently, similarity search methods are tailored to specific application domains, and only very basic techniques such as the nearest neighbor search solve general problems that arise in virtually all similarity search systems.

The main difficulty in the development of similarity measures is the communication between domain experts and similarity experts, as the similarity search involves a deep knowledge of the scientific concepts of the domain. Vice versa, domain experts can hardly imagine what a similarity search system may achieve and what concepts must be applied for this purpose.

Our idea is to alleviate this problem by a common framework that bundles concepts which are often applied in similarity search in a toolbox. This toolbox could contain various methods of feature extraction such as histograms, fourier transformation, and moment invariants, and various search methods such as similarity search, query decomposition for making the search robust, search for partial similarity, etc.

This toolbox could be complemented with visualization systems, evaluation methods and the above mentioned data mining techniques such as subspace clustering, convex hull and mind reader which may be used to determine whether the resulting feature transformation is adequate.

4. Conclusion

In this paper, we have described our previous work in the research domains of similarity search and data mining. The focus of our work was to develop general purpose index structures and query processing techniques such as the similarity join that allow a tight integration of these applications into a DBMS centered information infrastructure. Since most of the applications are data intensive, the efficiency of the corresponding database primitives is of highest importance.

In our future research directions, we have identified several key fields of research such as new data mining requirements, novel applications such as e-commerce, biometry, and the analysis of time sequences. A particular challenge of the future could be the development of a common framework for similarity measures.

References

[ABKS 99] Ankerst M., Breunig M. M., Kriegel H.-P., Sander J.: *OPTICS: Ordering Points To Identify the Clustering Structure*, ACM SIGMOD Int. Conf. on Management of Data, 1999.

[AGGR 98] Agrawal R., Gehrke J., Gunopulos D., Raghavan P.: *Automatic Subspace Clustering of High-Dimensional Data for Data Mining Applications*, ACM SIGMOD Int. Conf. on Management of Data, 1998, pp. 94-105.

[ALSS 95] Agrawal R., Lin K., Sawhney H., Shim K.: *Fast Similarity Search in the Presence of Noise, Scaling, and Translation in Time-Series Databases,* Proc. of the 21st Int. Conf. on Very Large Databases, 1995, pp. 490-501.

[BBB+ 97] Berchtold S., Böhm C., Braunmüller B., Keim D. A., Kriegel H.-P.: *Fast Parallel Similarity Search in Multimedia Databases*, Proc. ACM SIGMOD Int. Conf. on Management of Data, 1997, SIGMOD BEST PAPER AWARD.

[BBBK 00] Böhm C., Braunmüller B., Breunig M. M., Kriegel H.-P.: *Fast Clustering Based on High-Dimensional Similarity Joins*, Int. Conf. on Information Knowledge Management (CIKM), 2000.

[BBK 98a] Berchtold S., Böhm C., Kriegel H.-P.: *Improving the Query Performance of High-Dimensional Index Structures Using Bulk-Load Operations,* 6th. Int. Conf. on Extending Database Technology, 1998.

[BBK 98b] Berchtold S., Böhm C., Kriegel H.-P.: *The Pyramid-Technique: Towards indexing beyond the Curse of Dimensionality,* Proc. ACM SIGMOD Int. Conf. on Management of Data, 1998, pp. 142-153.

[BBK 02] Böhm C., Berchtold S., Keim D.: *Searching in High-dimensional Spaces: Index Structures for Improving the Performance of Multimedia Databases*, ACM Computing Surveys, 2002.

[BBK+ 00] Berchtold S., Böhm C., Keim D., Kriegel H.-P., Xu X.:*Optimal Multidimensional Query Processing Using Tree Striping,* Int. Conf. Data Warehousing and Knowledge Discovery (DaWaK), 2000.

[BBKK 97] Berchtold S., Böhm C., Keim D., Kriegel H.-P.: *A Cost Model For Nearest Neighbor Search in High-Dimensional Data Space*, ACM PODS Symposium on Principles of Database Systems, 1997.

[BBKK 01] Böhm C., Braunmüller B., Krebs F., Kriegel H.-P.: *Epsilon Grid Order: An Algorithm for the Similarity Join on Massive High-Dimensional Data*, ACM SIGMOD Int. Conf. on Management of Data, 2001.

[BBKM 99] Berchtold S., Böhm C., Kriegel H.-P., Michel U.: *Implementation of Multidimensional Index Structures for Knowledge Discovery in Relational Databases*, Int. Conf. on Data Warehousing and Knowledge Discovery (DaWaK), 1999.

[BBKM 00] Böhm C., Berchtold S., Kriegel H.-P., Michel U.: *Multidimensional Index Structures in Relational Databases*, Journal of Intelligent Information Systems (JIIS) 15 (1), 2000.

[BBKS 00] Böhm C., Braunmüller B., Kriegel H.-P., Schubert M.: *Efficient Similarity Search in Digital Libraries*, IEEE Int. Conf. Advances in Digital Libraries (ADL), 2000.

[BBJ+ 00] Berchtold S., Böhm C., Jagadish H. V., Kriegel H.-P., Sander J.: *Independent Quantization: An Index Compression Technique for High-Dimensional Data Spaces*, Proc. 16th Int. Conf. on Data Engineering, 2000.

[BK 99] Böhm C., Kriegel H.-P.: *Efficient Construction of Large High-Dimensional Indexes*, Int. Conf. on Data Warehousing and Knowledge Discovery (DaWaK), 1999.

[BK 00] Böhm C., Kriegel H.-P.: *Dynamically Optimizing High-Dimensional Index Structures*, in Proc. Int. Conf. on Extending Database Technology (EDBT), 2000.

[BK 01a] Böhm C., Kriegel H.-P.: *A Cost Model and Index Architecture for the Similarity Join*, IEEE Int. Conf. on Data Engineering, 2001.

[BK 01b] Böhm C., Kriegel H.-P.: *Determining the Convex Hull in Large Multidimensional Databases*, Int. Conf. on Data Warehousing and Knowledge Discovery (DaWaK), 2001.

[BKK 96] Berchtold S., Keim D., Kriegel H.-P.: *The X-tree: An Index Structure for High-Dimensional Data.* Proc. 22nd Int. Conf. on Very Large Data Bases (VLDB), Mumbai, India. Morgan Kaufmann, 1996, pp. 28-39.

[BKK 99] Böhm C., Klump G., Kriegel H.-P.: *XZ-Ordering: A Space-Filling Curve for Objects with Spatial Extension*, 6th Int. Symposium on Large Spatial Databases (SSD), 1999.

[BKK 01] Böhm C., Krebs F., Kriegel H.-P.: *Optimal Dimension Order: A Generic Technique for the Similarity Join*, Int. Conf. on Data Warehousing and Knowledge Discovery (DaWaK), 2002.

[BKS 93] Brinkhoff T., Kriegel H.-P., Seeger B.: *Efficient Processing of Spatial Joins Using R-trees*, ACM SIGMOD Int. Conf. on Management of Data, 1993.

[BKS 01] Böhm C., Seidl T., Kriegel H.-P.: *Combining Approximation Techniques and Vector Quantization for Adaptable Similarity Search*, Journal on Intelligent Information Systems (JIIS), 19(2), 2002.

[BM 77] Bayer R., McCreight E.M.: *Organization and Maintenance of Large Ordered Indices*, Acta Informatica 1(3), 1977, pp. 173-189.

[Böh 00] Böhm C.: *A Cost Model for Query Processing in High-Dimensional Data Spaces*, ACM Transactions on Database Systems (TODS) Vol. 25, No. 2, 2000.

[Böh 01] Böhm C.: *The Similarity Join - a Powerful Database Primitive for High Performance Data Mining*, tutorial, IEEE Int. Conf. on Data Engineering, 2001.

[Com 79] Comer D.: *The Ubiquitous B-tree*, ACM Computing Surveys 11(2), 1979, pp. 121-138.

[CP 00] Ciaccia P., Patella M.: *PAC Nearest Neighbor Queries: Approximate and Controlled Search in High-Dimensional and Metric Spaces*, Int. Conf. on Data Engineering, 2000, pp. 244-255.

[CPZ 98] Ciaccia P., Patella M., Zezula P.: *A Cost Model for Similarity Queries in Metric Spaces*. ACM Symp. on Principles of Database Systems (PODS), 1998, pp. 59-68.

[FBF+ 94] Faloutsos C., Barber R., Flickner M., Hafner J., Niblack W., Petkovic D., Equitz W.: *Efficient and Effective Querying by Image Content.* Journal of Intelligent Information Systems, Vol. 3, 1994, pp. 231-262.

[FRM 94] Faloutsos C., Ranganathan M., Manolopoulos Y.: *Fast Subsequence Matching in Time-Series Databases*. Proc. ACM Int. Conf. on Management of Data (SIGMOD), 1994, pp. 419-429.

[GM 93] Gary J. E., Mehrotra R.: *Similar Shape Retrieval using a Structural Feature Index,* Information Systems, Vol. 18, No. 7, 1993, pp. 525-537.

[HJR 97] Huang Y.-W., Jing N., Rundensteiner E. A.: *Spatial Joins Using R-trees: Breadth-First Traversal with Global Optimizations*, Int. Conf. on Very Large Databases (VLDB), 1997.

[ISF 98] Ishikawa Y., Subramanya R., Faloutsos C.: *MindReader: Querying Databases Through Multiple Examples*. Proc. 24th Int. Conf. on Very Large Databases (VLDB), 1998, pp. 218-227.

[Jag 91] Jagadish H. V.: *A Retrieval Technique for Similar Shapes,* Proc. ACM SIGMOD Int. Conf. on Management of Data, 1991, pp. 208-217.

[KSF+ 96] Korn F., Sidiropoulos N., Faloutsos C., Siegel E., Protopapas Z.: *Fast Nearest Neighbor Search in Medical Image Databases,* Proc. 22nd VLDB Conference, 1996, pp. 215-226.

[LJF 95] Lin K., Jagadish H. V., Faloutsos C.: *The TV-Tree: An Index Structure for High-Dimensional Data,* VLDB Journal, Vol. 3, pp. 517-542, 1995.

[LR 96] Lo M.-L., Ravishankar C. V.: *Spatial Hash Joins*, ACM SIGMOD Int. Conf. on Management of Data, 1996.

[PD 96] Patel J.M., DeWitt D.J.: *Partition Based Spatial-Merge Join*, ACM SIGMOD Int. Conf. on Management of Data, 1996.

[PM 97] Papadopoulos A., Manolopoulos Y.: *Performance of Nearest Neighbor Queries in R-Trees*, 6th Int. Conf. on Database Theory, LNCS Vol. 1186, 1997, pp. 394-408.

[RGF 98] Riedel, Gibson, Faloutsos: *Active Storage for Large Scale Data Mining and Multimedia*, Int. Conf. on Very Large Databases VLDB, 1998.

[Sch 95] Schiele O. H.: *Forschung und Entwicklung im Maschinenbau auf dem Weg in die Informationsgesellschaft* (in German, translation by the author), Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie, 1995, http://www.iid.de/informationen/vdma/infoway3.html.

[SEKX 98] Sander J., Ester M., Kriegel H.-P., Xu X.: *Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and its Applications*, Data Mining and Knowledge Discovery, Kluwer Academic Publishers, 2(2), 1998.

[SSA 97] Shim K., Srikant R., Agrawal R.: *High-Dimensional Similarity Joins*, Int. Conf. on Data Engineering (ICDE), 1997.

[TTSF 00] Traina C. Jr., Traina A., Seeger B., Faloutsos C.: *Slim-Trees: High Performance Metric Trees Minimizing Overlap Between Nodes*. Int. Conf. on Extending Database Technology (EDBT), 2000, pp. 51-65.

[WJ 96] White D.A., Jain R.: *Similarity indexing with the SS-tree*, Proc. 12th Int. Conf on Data Engineering, 1996.

[WSB 98] Weber R., Schek H.-J., Blott S.: *A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces.* Proc. 24th Int. Conf. on Very Large Databases (VLDB), 1998, pp. 194-205.