Lecture Notes in Computer Science 2341
Edited by G. Goos, J. Hartmanis, and J. van Leeuwen

Springer
Berlin
Heidelberg
New York
Barcelona
Hong Kong
London
Milan
Paris
Tokyo

Cui Yu

High-Dimensional Indexing

Transformational Approaches
to High-Dimensional Range and Similarity Searches

Springer

Author

Cui Yu
Monmouth University, Department of Computer Science
West Long Branch, NJ 07764, USA

National University of Singapore, Department of Computer Science
Kent Ridge, Singapore 117543, Singapore

E-mail:cyu@monmouth.edu

Cataloging-in-Publication Data applied for

Bibliographic information published by Die Deutsche Bibliothek

Die Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data is available in the Internet at <http://dnb.ddb.de>.

CR Subject Classification (1998): H.3.1, H.2.8, H.3, H.2, E.2, E.1, H.4, H.5.1

ISSN 0302-9743
ISBN 3-540-44199-9 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag Berlin Heidelberg New York
a member of BertelsmannSpringer Science+Business Media GmbH

http://www.springer.de

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Boller Mediendesign
Printed on acid-free paper SPIN: 10869935 06/3142 5 4 3 2 1 0

Preface

Many new applications, such as multimedia databases, employ the so-called feature transformation which transforms important features or properties of data objects into high-dimensional points. Searching for 'similar' objects based on these features is thus a search of points in this feature space. Another high-dimensional database example is stock price information systems, where time series data are stored and searched as high-dimensional data points. To support efficient query processing and knowledge discovery in these high-dimensional databases, high-dimensional indexes are required to prune the search space and efficient similarity join strategies employing these indexes have to be designed.

High-dimensional indexing has been considered an important means to facilitate fast query processing in data mining, fast retrieval, and similarity search in image and scientific databases. Existing multi-dimensional indexes such as R-trees are not scalable in terms of the number of dimensions. It has been observed that the performances of R-tree-based index structures deteriorate rapidly when the dimensionality of data is high [11, 12]. This is due to rapid growth in overlap in the directory with respect to growing dimensionality of data, requiring a large number of subtrees to be searched for each query. The problem is further exacerbated by the fact that a small high-dimensional query covering a very small fraction of the data space has actually a very large query width along each dimension. Larger query widths imply that more subtrees need to be searched. In this monograph, we study the problem of high-dimensional indexing to support range, similarity, and K-nearest neighbor (KNN) queries, and similarity joins.

To efficiently support window/range queries, we propose a simple and yet efficient transformation-based method called the iMinMax(θ). The method maps points in high-dimensional spaces to single dimensional values determined by their maximum or minimum values among all dimensions. With such representations, we are able to index high-dimensional data points using a conventional B^+-tree. By varying the tuning 'knob', θ, we can obtain a different family of iMinMax structures that are optimized for different distributions of data sets. Hence, the method is tunable to yield best performance based on data distributions. For a d-dimensional space, a window query needs to be transformed into d subqueries. However, some of these subqueries can

be pruned away without evaluation, further enhancing the efficiency of the scheme. Extensive experiments were conducted, and experimental comparison with other existing methods such as the VA-file and Pyramid-tree provides an insight on the efficiency of the proposed method.

To efficiently support similarity or K-nearest neighbor (KNN) queries, we propose a specialized metric-based index called iDistance, and an extension of the iMinMax(θ). In the iDistance, a metric-based index, the high-dimensional space is split into partitions, and each partition is associated with an 'anchor' point (called a reference point) whereby other points in the same partitions can be made reference to. With such a representation, the transformed points can then be indexed using a B^+-tree, and KNN search in the high-dimensional space is performed as a sequence of increasingly larger range queries on the single dimensional space. Such an approach supports efficient filtering of data points that are obviously not in the answer set without incurring expensive distance computation. Furthermore, it facilitates fast initial response time by providing users with approximate answers *online* that are progressively refined till all correct answers are obtained (unless the users terminate prematurely). Unlike KNN search, similarity range search on iDistance is straightforward and is performed as a spherical range query with fixed search radius. Extensive experiments were conducted, and experimental results show that the iDistance is an efficient index structure for nearest neighbor search.

The iMinMax(θ) is designed as a generic structure for high-dimensional indexing. To extend the iMinMax(θ) for KNN search, we design KNN processing strategies based on range search to retrieve approximate nearest neighbor data points with respect to a given query point. With proper data sampling, accuracy up to 90% can be supported very efficiently. For a more accurate retrieval, bigger search ranges must be used, which is less efficient.

In conclusion, both iMinMax(θ) and iDistance methods are flexible, efficient, and easy to implement. Both methods can be crafted into existing DBMSs easily. This monograph shows that efficient indexes need not necessarily be complex, and the B^+-tree, which was designed for traditional single dimensional data, could be just as efficient for high-dimensional indexing. The advantage of using the B^+-tree is obvious. The B^+-tree is well tested and optimized, and so are its other related components such as concurrency control, space allocation strategies for index and leaf nodes, etc. Most importantly, it is supported by most commercial DBMSs. A note of caution is that, while it may appear to be straightforward to apply transformation on any data set to reuse B^+-trees, guaranteeing good performance is a non-trivial task. In other words, a careless choice of transformation scheme can lead to very poor performance. I hope this monograph will provide a reference for and benefit those who intend to work on high-dimensional indexing.

I am indebted to a number of people who have assisted me in one way or another in materializing this monograph. First of all, I wish to express my

appreciation to Beng Chin Ooi, for his insight, encouragement, and patience. He has taught me a great deal, instilled courage and confidence in me, and shaped my research capability. Without him, this monograph, which is an extended version of my PhD thesis [104], would not have materialized.

I would like to thank Kian-Lee Tan and Stéphane Bressan for their advice and suggestions. Kian-Lee has also proof-read this monograph and provided detailed comments that greatly improved the literary style of this monograph. I would like to thank H.V. Jagadish, for his insight, comments, and suggestions regarding iDistance; Rudolf Bayer and Mario Nascimento, for their comments and suggestions concerning the thesis; and many kind colleagues, for making their source codes available. I would like to thank Shuguang Wang, Anirban Mondal, Hengtao Shen, and Bin Cui, and the editorial staff of Springer-Verlag for their assistance in preparing this monograph. I would like to thank the School of Computing, National University of Singapore, for providing me with a graduate scholarship and facility for completing this monograph.

Last but not least, I would like to thank my family for their support, and I would like to dedicate this monograph to my parents for their love.

May 2002 *Cui Yu*

Contents