IDEA GROUP PUBLISHING

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITJ2573

Scalable QSF-Trees: Retrieving Regional Objects in High-Dimensional Spaces

Ratko Orlandic, Illinois Institute of Technology, USA*
Byunggu Yu, University of Wyoming, USA

ABSTRACT

Many database applications require effective representation of regional objects in high-dimensional spaces. By applying an original query transformation, a recently proposed access method for regional data, called the simple QSF-tree (sQSF-tree), effectively attacks the limitations of traditional spatial access methods in spaces with many dimensions. Nevertheless, sQSF-trees are not immune to all problems associated with high data dimensionality. Based on the analysis of sQSF-trees, this paper presents a new variant of sQSF-trees, called the scalable QSF-tree (cQSF-tree), which relies on a heuristic optimization to reduce the number of false drops into pages that contain no object satisfying the query. By increasing the selectivity of search predicates, cQSF-trees improve the performance of multi-dimensional selections. Experimental evidence shows that cQSF-trees are more scalable than sQSF-trees to the growing data dimensionality. The performance improvements also increase with more skewed data distribution.

Keywords: data dimensionality, database management, spatial access methods.

INTRODUCTION

There is a large body of literature on the problem of accessing data in high-dimensional spaces (Berchtold et al., 1996, 1998; Lin et al., 1995; Orlandic & Yu, 2002; Sakurai et al., 2000; Weber et al., 1998; White & Jain, 1996). However, the proposed techniques almost always assume data sets representing points in space. In many applications, effective representation of extended (regional) data is also important. Regional data are usually associated with low-dimensional spaces of geographic applications. However, through aggregation or clustering, such data may naturally appear in high-dimensional spaces as well.

For example, when the massive high-dimensional data of advanced scientific applications are clustered in files on tertiary storage, storage considerations often prevent the corresponding access structure from keeping the descriptors of all items in the repository. Instead, the content of each

* *This author's work was partially supported by the NSF grant IIS-0312266.*

file can be approximated in the access structure by the minimal bounding rectangle (MBR) enclosing all data points of the given file (Orlandic, 2003). Similarly, in order to reduce the cost of dynamic updates, the multi-dimensional databases of location-based services frequently approximate the position of a moving object by the bounded rectangle of a larger area in which the object currently resides. Since the position is usually only one of many relevant parameters describing a moving object, the index (access) structure appropriate for these environments must deal with regional data in spaces with possibly many dimensions.

Other applications in which regional objects naturally appear in higher dimensional spaces include online analytical processing as well as multimedia and image-recognition systems. In the latter systems, objects are usually mapped to long *d*-dimensional feature vectors. However, for the purposes of recognition, only a subset of features, called principal components, is actually used (Swets & Weng, 1996). After populating the projected space, images are grouped in classes, each of which can be represented by its approximate region in the space and stored in a spatial access method. In order to identify the most likely class for the given object, the process of image recognition must employ a form of spatial retrieval with the probabilistic ranking of retrieved objects.

Unlike point access methods (PAMs), spatial access methods (SAMs) are designed to support different search operators (e.g., overlap, containment, and enclosure) over both points and regional objects in multi-dimensional spaces (Gaede & Gunther, 1998). To reduce the storage overhead of the index structure, extended regional objects are typically approximated by their MBRs. There are many MBR-based SAMs, which are usually classified into: region-overlapping (Guttman, 1984; Beckmann et al., 1990), object-clipping (Sellis et al., 1987) and object-transformation (Pagel et al., 1993) schemes.

Unfortunately, each group of traditional SAMs suffers from major conceptual problems that have a tendency to grow with data dimensionality (Orlandic & Yu, 2000). We call these problems *conceptual* because they tend to be associated with the very idea underlying a group of SAMs. For example, the region overlap in R-trees (Guttman, 1984) and R*-trees (Beckmann et al., 1990) translates into a necessity to traverse many index paths, which increases the number of accessed *nodes* (*index pages*). However, the amount of overlap in these structures rapidly grows with data dimensionality (Berchtold et al., 1996). Object clipping (Sellis et al., 1987) creates multiple clips of a single regional object, which increases the size of the structure and degrades retrieval performance. Because the probability of clipping an object grows with dimensionality, these negative effects of clipping are more pronounced in higher dimensional spaces. A major drawback of object-transformation schemes (Pagel et al., 1993) is that a relatively small query window in the original space may map into a large search region in the transformed space. The magnitude of this problem increases rapidly as the number of dimensions grows (Orlandic & Yu, 2000).

Few access methods for high-dimensional data can accommodate extended regional objects. *X-trees* (Berchtold et al., 1996) and *simple QSF-trees*, or just *sQSF-trees* (Orlandic & Yu, 2000; Yu et al., 1999), are the exceptions. X-trees are designed to address the problem of region overlap in R*-trees. Instead of allowing splits that introduce high overlap, they extend index pages over the usual size. These clusters of pages, called super-nodes, are searched