Appl. Math. Lett. Vol. 4, No. 5, pp. 61–62, 1991
Printed in Great Britain

0893-9659/91 \$3.00 + 0.00
Pergamon Press plc

Metric Trees

JEFFREY K. UHLMANN

Information Technology Division
Naval Research Laboratory

(Received March 1991 and in revised form May 1991)

Abstract. A new multidimensional search structure is described that is able to exploit metric information to efficiently satisfy a large class of proximity queries.

Numerous problems in computational geometry require the efficient identification of elements from a finite set of points that are in some defined proximity of a given query point, where an efficient algorithm is one that avoids examining every point in the set [1]. A variety of data structures are known that satisfy particular types of region queries, with scaling in time that is rootic or polylogarithmic in the cardinality of the searched set [2,3]. In this letter an alternative data structure is described for efficiently solving a more general class of queries.

Given a finite set P of k-dimensional points (or vectors) with metric $d(P_i, P_j), P_i, P_j \in P$, a *metric tree* $\mathcal{T}$ is constructed from P as follows:

(1) If $|P| = 0$, then the empty tree $\mathcal{T} = nil$ is created.
(2) Otherwise, select an arbitrary element x from P and identify the median element m of the set $\{d(x,y), \forall\, y \in P\}$. Let the point x be associated with the root of the tree and let its *left* subtree be a metric tree constructed from the set $\{P_i \mid d(x, P_i) \leq m, P_i \neq x\}$ and its *right* subtree be a metric tree constructed from the set $\{P_i \mid d(x, P_i) > m\}$. (Thus the tree is balanced as long as the median distance is unique.) The median value m is also associated with the root. $\mathcal{T}_x, \mathcal{T}_m, \mathcal{T}_{\text{left}}$ and $\mathcal{T}_{\text{right}}$ refer, respectively, to the point x, the real value m, the left subtree and the right subtree of $\mathcal{T}$.

This procedure will be referred to as a *ball* decomposition, since each vertex represents a partition of the space into a ball of radius $\mathcal{T}_m$ about $\mathcal{T}_x$ and into its complement. This decomposition is useful for half-space queries because both subtrees must be examined only when the hypersurface delimiting the half-space intersects the bounded region. It can also be used to satisfy other types of queries. For example, given a point v, the set $\{P_i \mid d(P_i, v) \leq r\}$ can be identified by using the metric tree $\mathcal{T}$ as follows:

(1) If $d(\mathcal{T}_x, v) \leq r$, then $\mathcal{T}_x$ is an element of the desired set.
(2) Otherwise,
If $d(\mathcal{T}_x, v) + r > \mathcal{T}_m$, then recursively identify the set $\{P_i \mid d(P_i, v) \leq r\}$ by using the metric tree $\mathcal{T}_{\text{right}}$.
If $d(\mathcal{T}_x, v) + \mathcal{T}_m \leq r$, then every point in $\mathcal{T}_{\text{left}}$ is an element of the desired set. Otherwise, if $d(\mathcal{T}_x, v) - r \leq \mathcal{T}_m$, then recursively identify the set $\{P_i \mid d(P_i, v) \leq r\}$ by using the metric tree $\mathcal{T}_{\text{left}}$.

While efficient performance can usually be expected for queries of this kind, it is not difficult to construct cases that require an examination of every point. For example, a set of points distributed on the surface of the unit ball and a query asking for the set of points strictly within the unit ball allows no subtrees to be pruned during the search.

The number of points and the equation used to define the partition surfaces can be chosen according to the specifics of the application. For example, the equation $d(x, z) + d(y, z) = c$, $\forall z$, defines a generalized elliptical surface that has applications to multivariate Gaussian correlation problems. More generally, though, the value of metric trees is that they can permit the divide-and-conquer strategy to be applied to proximity search problems in arbitrary metric spaces. Although efficient query-time performance cannot be guaranteed for all types of queries in all metric spaces, good performance can be expected for a large number of practical problems in multidimensional search and pattern recognition.

REFERENCES

1. F. Preparata and M. Shamos, *Computational Geometry: An Introduction*, Springer-Verlag, New York, (1985).
2. K. Mehlhorn, *Multidimensional Searching and Computational Geometry*, Springer-Verlag, Berlin, (1984).
3. H. Samet, *The Design and Analysis of Spatial Data Structures*, Addison-Wesley, (1990).

Information Technology Division, Naval Research Laboratory (CODE 5570), Washington, D.C. 20375-5000.