

Exact L_∞ -Nearest-Neighbor Search in High Dimensions

(Exakte L_∞ -Nächster-Nachbar-Suche in hohen Dimensionen)

Dissertation zur Erlangung des Doktorgrades

vorgelegt am

Fachbereich Mathematik und Informatik
der Freien Universität Berlin

2002

von

Laura Heinrich-Litan

Institut für Informatik

Freie Universität Berlin

Takustraße 9

14195 Berlin

litan@inf.fu-berlin.de

Betreuer: Prof. Dr. Helmut Alt
Institut für Informatik
Freie Universität Berlin
Takustraße 9
14195 Berlin
Germany
alt@inf.fu-berlin.de

Gutachter: Prof. Dr. Helmut Alt
Institut für Informatik
Freie Universität Berlin
Takustraße 9
14195 Berlin
Germany
alt@inf.fu-berlin.de

Prof. Dr. techn. Franz Aurenhammer
Institut für Grundlagen
der Informationsverarbeitung
Technische Universität Graz
Inffeldgasse 16b/1
8010 Graz
Austria
auren@igi.tu-graz.ac.at

Vorlage zur Begutachtung: 15. Juli 2002
Termin der Disputation: 4. November 2002

Hiermit versichere ich an Eides Statt, dass ich diese Arbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe.

Berlin, im Juli 2002

Contents

1	Introduction	1
2	Nearest-Neighbor Search without preprocessing	5
2.1	The CUBE METHOD	5
2.1.1	The idea and the core algorithm	5
2.1.2	Searching the cube by scanning	7
2.1.3	Computation of the side length α	11
2.1.4	The total expected runtime	13
2.2	The ADAPTIVE METHOD	15
3	Nearest-Neighbor Search with preprocessing	19
3.1	Speeding up the CUBE METHOD by rejecting points	19
3.1.1	Searching the cube by rejecting	20
3.1.2	Searching the cube by rejecting and scanning	24
3.2	Speeding up the CUBE METHOD by using monotone sequences	34
3.2.1	Preliminaries	34
3.2.2	Computing a partition into monotone subsequences	37
3.2.3	A set with longest monotone subsequences of length $\lceil n^{\frac{1}{2d-1}} \rceil$	39
3.2.4	Expected height and width of a random order	41
3.2.5	Searching by using a partition into monotone sequences	43
4	Extensions of the CUBE METHOD	53
4.1	The GROWING-CUBE variant	53
4.2	k -Nearest-Neighbor Search	62
4.2.1	The algorithm	62
4.2.2	The expected runtime analysis	63
4.3	Other distributions	69
4.4	External-Memory Nearest-Neighbor Search	73
4.4.1	The External-Memory Model of Computation	73
4.4.2	The External-Memory Data Structure	73
4.5	Experiments	78
5	Time-Space Tradeoffs for Nearest-Neighbor Search	81
5.1	The data structure	81

5.2 The expected runtime and expected space complexity	82
Conclusions	89
Bibliography	94
Curriculum Vitae	96
Zusammenfassung	97