Approximate Nearest Neighbor in High Dimensions via Hashing

Aris Gionis
Piotr Indyk
Rajeev Motwani

Stanford University

Nearest Neighbor Search (NNS)

Given: n points $P \subset R^d$.

Goal: a data structure which for any query $q \in R^d$ returns the point(s) in P closest to q.

In this talk: Hamming metric, l_2 (Euclidean) or l_1 (Manhattan) norms.

Past Work

k-d-tree	Bentley '75
K-D-B-tree	Robinson '81
R-tree	Guttman '84
LSD-tree	Henrich, Six, Widmayer '89
R^*-tree	Beckmann, Kriegel et al '90
hB-tree	Lomet, Salzberg '90
Buddy-tree	Seeger, Kriegel '90
TV–ree	Lin, Jagadish, Faloutsos '94
SS-tree	White, Jain '96
VAMSplit-tree	White, Jain'96
X-tree	Berchtold, Keim, Kriegel '96
SR-tree	Katayama, Satoh '97
Pyramid-tree	Berchtold, Bohm, Kriegel '98

Past Work ctd.

VP-tree	Uhlmann'91
	Yannilos'93
GNAT (a tree)	Brin'95
MVP-tree	Bozkaya, Ozoyoglu'97
M-tree	Patella, Ciaccia, Zezula'97

Computational Geometry: mainly low dimension.

The curse of dimensionality

"Theorem:" For dimensionality "high enough", any data structure has linear query time (or exponential space).

Evidence:

- experimental
- analytical (Weber, Schek, Blott'98)

An approximate approach

Given: n points $P \subset R^d$, $\epsilon > 0$.

Goal: a data structure which for any query $q \in R^d$ returns the point(s) p' in P such that

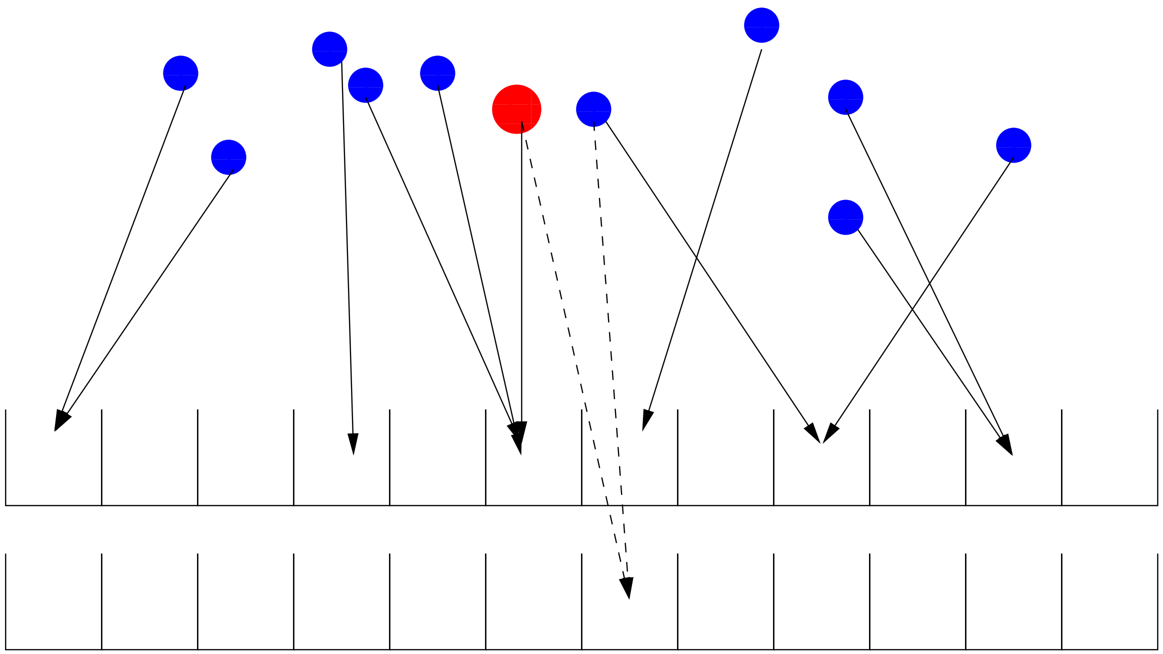
$$|q - p'| \leq (1 + \epsilon) \min_{p \in P} |q - p|$$

Also allow randomization.

Why approximation ?

- right answers much closer than irrelevant ones
- time/quality tradeoff

Locality Sensitive Hashing



- probability of collision higher for close objects
- repeat hashing to increase the probability

Setting

- Hamming space, i.e. $\{0,1\}^d$
- ϵ-neighbor with parameter r:
  - if $\min_{p \in P} |q-p| \leq r$: return p s.t. $|q-p| \leq r(1+\epsilon)$
  - if $\min_{p \in P} |q-p| > (1+\epsilon)r$: return NONE
  - otherwise: unspecifed

LSH functions

Family $\mathcal{G} = \{g : \{0,1\}^d \to Y\}$ is (r_1, r_2, P_1, P_2)*-sensitive* if for all q, p

- if $|p - q| \leq r_1$ then $\Pr_{\mathcal{G}}[g(q) = g(p)] \geq P_1$,
- if $|p - q| > r_2$ then $\Pr_{\mathcal{G}}[g(q) = g(p)] \leq P_2$.

LSH ctd

Hamming metric: sample k random bits, i.e.

$$g(p) = p_{|I}$$

for $I \subset \{1 \ldots d\}, |I| = k$.

We have

$$\Pr_{\mathcal{G}}[g(q) = g(p)] = \left(1 - \frac{|p-q|}{d}\right)^k$$

LSH: Theorem

Question: How many indices l do we need for correctness?

Theorem: Setting $l = n^{\rho}$ for $\rho = \frac{\log 1/P_1}{\log 1/P_2}$ is sufficient with constant probability.

(Hamming metric $\Rightarrow$ $\rho = 1/(1+\epsilon)$)

LSH: Analysis

Define:

- p^* - a point s.t. $|q - p^*| \leq r$
- FAR(q) - all p with $|q - p| > (1 + \epsilon)r$
- BUCKET$_j(q)$ - all p s.t. $g_j(p) = g_j(q)$

Main idea:

Events:

- $E_1 : \sum_{j=1}^{l} |\mathsf{FAR}(q) \cap \mathsf{BUCKET}_j(q)| \leq 3l$
- $E_2 : g_j(p^*) = g_j(q)$ for some g_j, $1 \leq j \leq l$

Will show: $\Pr[E_1] > \frac{1}{2} + \alpha$ and $\Pr[E_2] > \frac{1}{2} + \alpha$

Thus: $\Pr[E_1 \wedge E_2] > 2\alpha$

LSH: Bad collisions

Let $p \in \mathsf{FAR}(q)$. Then $\Pr[p \in \mathsf{BUCKET}_j(q)] \leq P_2^k$

For $k = \log_{\frac{1}{P_2}} n$ we have $\Pr[p \in \mathsf{BUCKET}_j(q)] \leq P_2^{\log_{\frac{1}{P_2}} n} = \frac{1}{n}$

By linearity of expectation:

$$E[|\mathsf{FAR}(q) \cap \mathsf{BUCKET}_j(q)|] \quad \leq n \cdot \frac{1}{n} \quad = 1$$

$$E[\sum_{j=1}^{l} |\mathsf{FAR}(q) \cap \mathsf{BUCKET}_j(q)|] \qquad \leq l$$

By Markov inequality $\Pr[\sum_{j=1}^{l} |\mathsf{FAR}(q) \cap \mathsf{BUCKET}_j(q)| > 3l] \leq \frac{1}{3}$

LSH: Good collision

For any g_j, $1 \leq j \leq l$

$$\Pr[g_j(p^*) = g_j(q)] \geq P_1^k = P_1^{\log_{1/P_2} n} = n^{-\frac{\log 1/P_1}{\log 1/P_2}} = n^{-\rho}$$

For $l = n^\rho$ we have

$$\begin{aligned} \Pr[E_2] &\geq 1 - (1 - \Pr[g_j(p^*) = g_j(q)])^l \\ &\geq 1 - (1 - n^{-\rho})^{n^\rho} \\ &\geq 1 - 1/e > \frac{1}{2} \end{aligned}$$

Extension to l_1

- round the coordinates to $\{1 \dots M\}$
- embed $\{1 \dots M\}^d$ into $\{0,1\}^{dM}$ via unary mapping:

$$
\begin{aligned}
(x_1, \dots, x_d) \quad &\to \quad \underbrace{1\dots1}_{x_1}\underbrace{0\dots0}_{d-x_1}\dots\underbrace{1\dots1}_{x_d}\underbrace{0\dots0}_{d-x_d} \\
(y_1, \dots, y_d) \quad &\to \quad \underbrace{1\dots1}_{y_1}\underbrace{0\dots0}_{d-y_1}\dots\underbrace{1\dots1}_{y_d}\underbrace{0\dots0}_{d-y_d}
\end{aligned}
$$

Extension to l_2

- little difference between l_1 and l_2 in high dimensions: $\epsilon < 3\%$
- embedding of l_2^d into $l_1^{O(d)}$ using random rotations [Figiel et al'77]

Other extensions

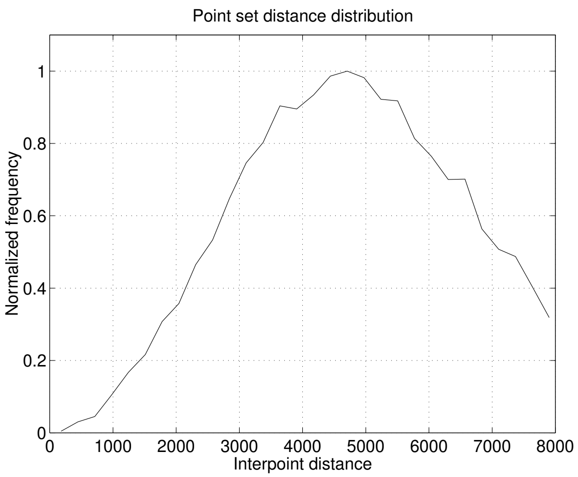
- insertions/deletions
- nearest neighbor:
  - index for $r = 1, (1+\epsilon), (1+\epsilon)^2, \ldots$
  - binary search for nearest neighbor
- secondary storage
- join version (via sorting)
- other metrics (e.g. set intersection size in Altavista)

Experimental setup

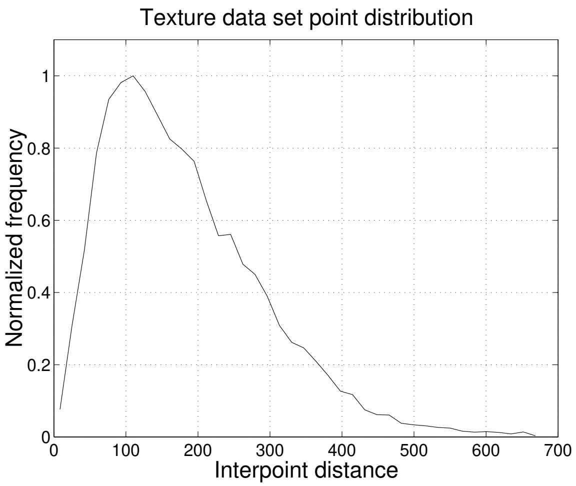
- datasets:
  - color histograms (Corel Draw): $n = 20,000$; $d = 8 \ldots 64$
  - texture data (aerial photos): $n = 270,000$; $d = 60$
- disk: 8 KB/block $\Rightarrow \frac{8192}{d}$ pts/block, $2 \cdot d \cdot n$ bytes/index.
- performance measures:
  - speed: avg number of blocks accessed
  - error:

$$\text{avg}_q \frac{\text{LSH NN distance}(q)}{\text{NN distance}(q)}$$

Profiles

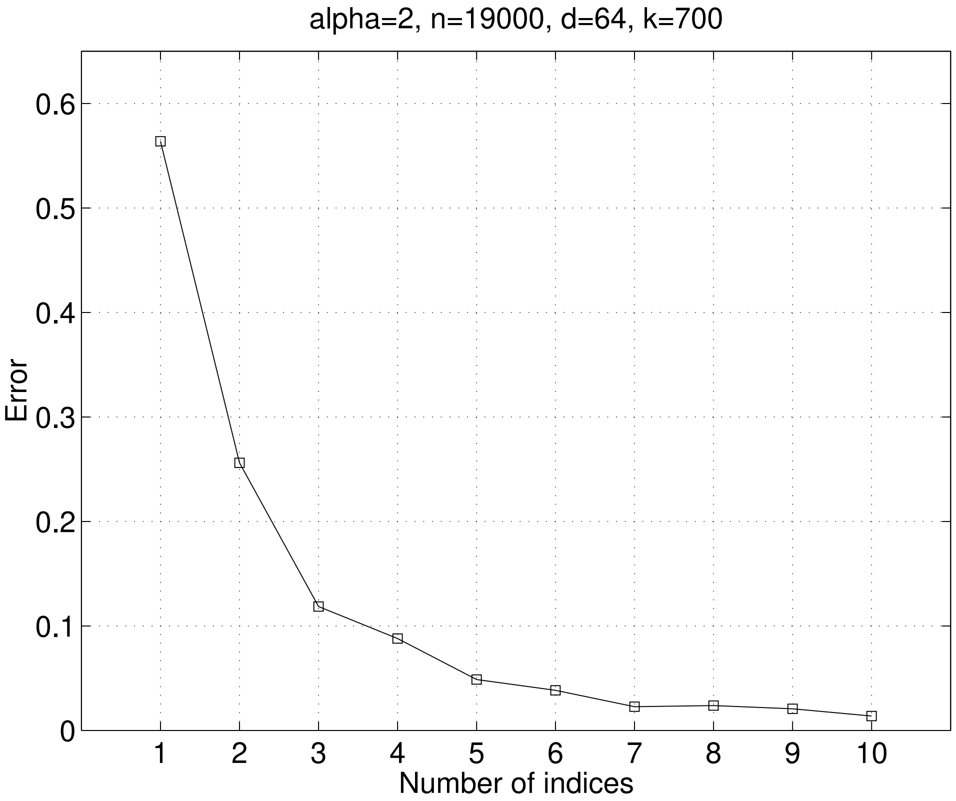


(a) histograms

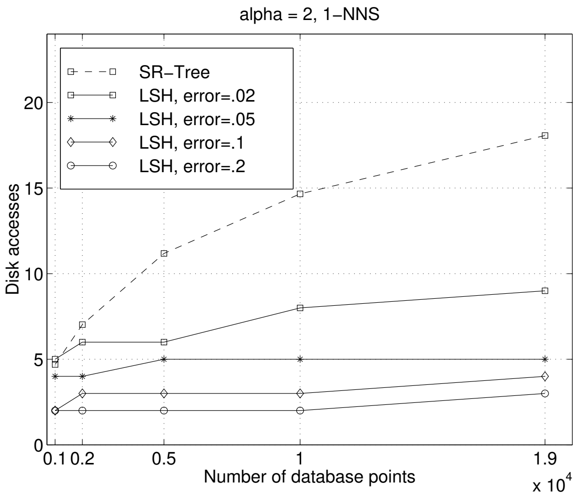


(b) texture

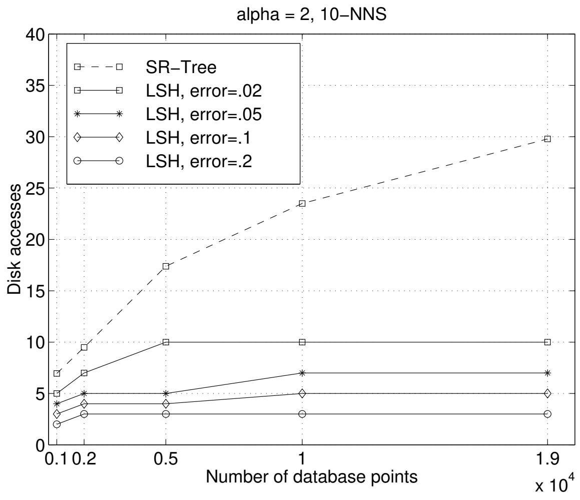
Error vs number of indices



Dependence on n

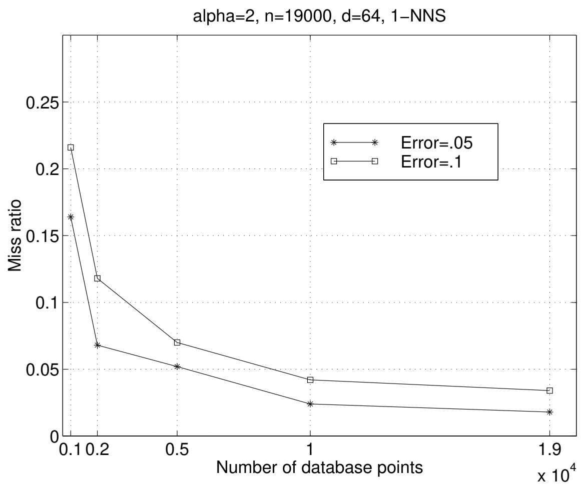


(a) 1-NNS

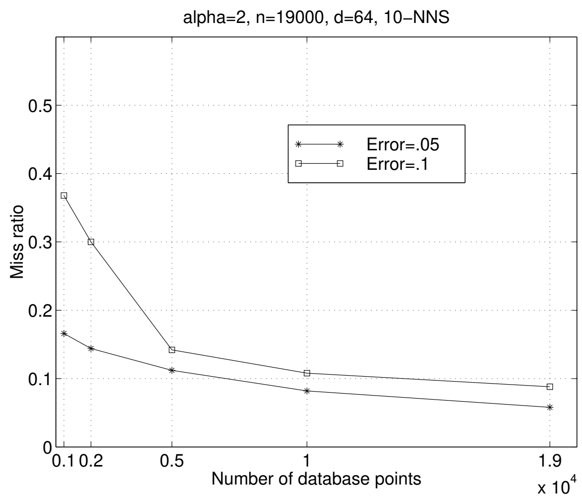


(b) 10-NNS

Miss ratios

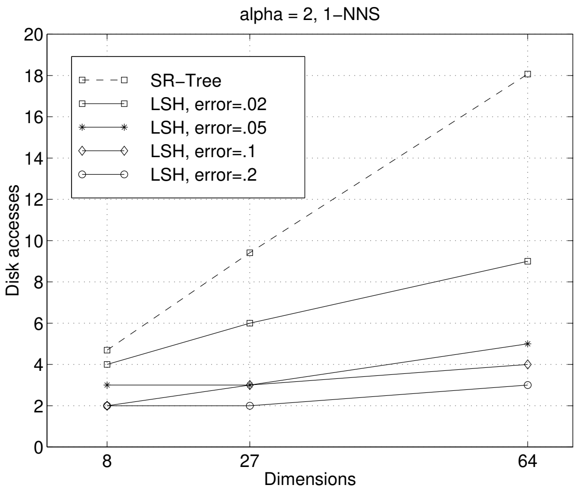


(a) 1-NNS

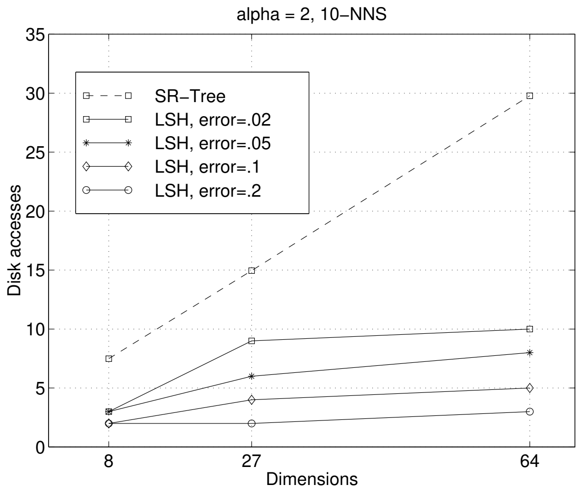


(b) 10-NNS

Dependence on d

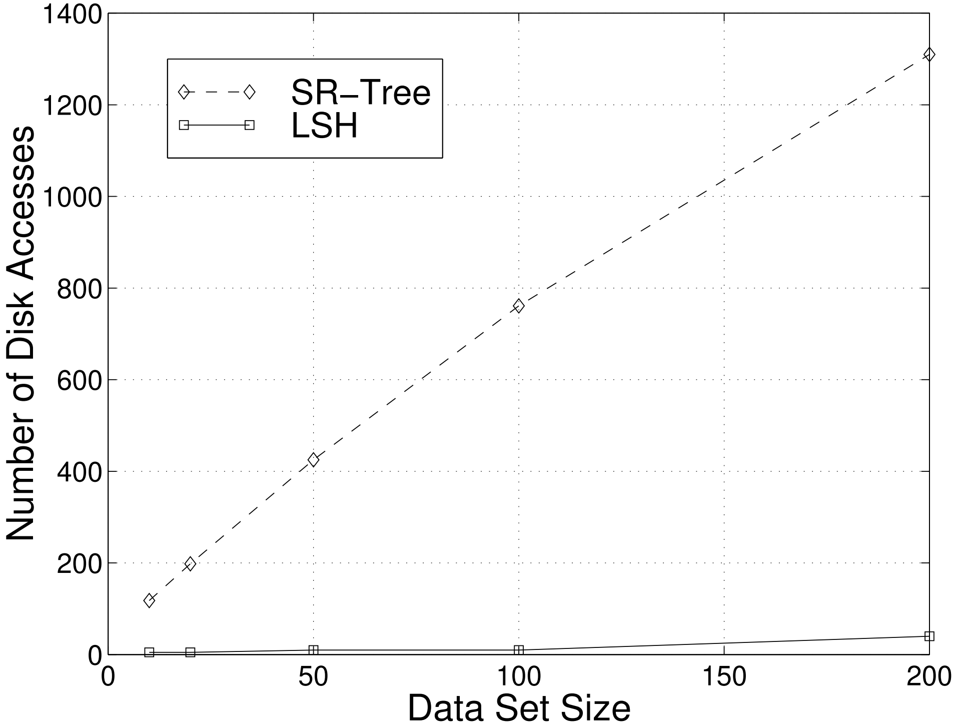


(a) 1-NNS

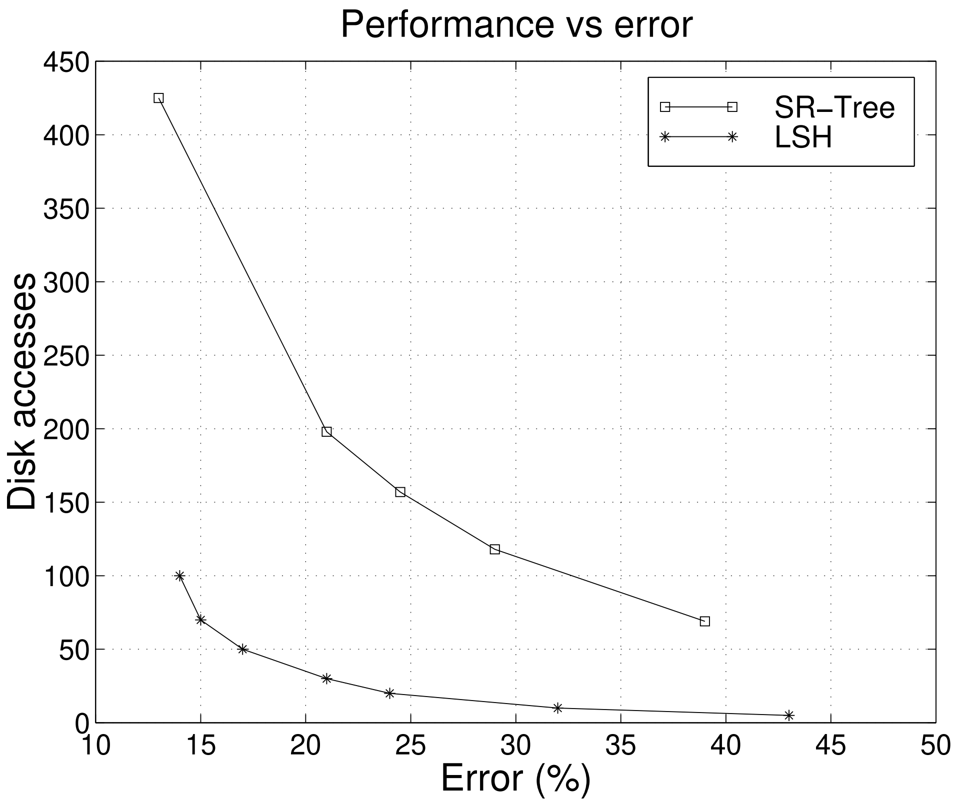


(b) 10-NNS

Texture data: dependence on n



Texture data: approximate SR-tree



Conclusions

- Locality Sensitive Hashing:
  - fast approximation
  - dynamic/join versions
- improvements:
  - deterministic index construction
  - hybrid techniques