View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Dagstuhl Research Online Publication Server

Approximate Nearest-Neighbor Search for Line Segments

Ahmed Abdelkader
Oden Institute for Computational Engineering and Sciences,
The University of Texas at Austin, TX, USA

David M. Mount
Department of Computer Science and Institute of Advanced Computer Studies,
University of Maryland, College Park, MD, USA

Abstract

Approximate nearest-neighbor search is a fundamental algorithmic problem that continues to inspire study due its essential role in numerous contexts. In contrast to most prior work, which has focused on point sets, we consider nearest-neighbor queries against a set of line segments in $\mathbb{R}^d$, for constant dimension d. Given a set S of n disjoint line segments in $\mathbb{R}^d$ and an error parameter $\varepsilon > 0$, the objective is to build a data structure such that for any query point q, it is possible to return a line segment whose Euclidean distance from q is at most $(1 + \varepsilon)$ times the distance from q to its nearest line segment. We present a data structure for this problem with storage $O((n^2/\varepsilon^d)\log(\Delta/\varepsilon))$ and query time $O(\log(\max(n, \Delta)/\varepsilon))$, where Δ is the spread of the set of segments S. Our approach is based on a covering of space by anisotropic elements, which align themselves according to the orientations of nearby segments.

2012 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases Approximate nearest-neighbor searching, Approximate Voronoi diagrams, Line segments, Macbeath regions

Digital Object Identifier 10.4230/LIPIcs.SoCG.2021.4

Related Version *Full Version*: https://arxiv.org/abs/2103.16071

Funding Supported by NSF grant CCF-1618866 and an Ann G. Wylie Dissertation Fellowship.

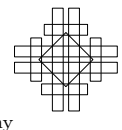
Acknowledgements The first author's work was conducted in part at the University of Maryland.

1 Introduction

Proximity queries are essential building blocks in many important algorithms with numerous applications [21–25, 28–30]. A primary example is *nearest-neighbor searching*, where a given set of n points P in $\mathbb{R}^d$ is preprocessed into a data structure so that queries can be answered efficiently. As the complexity bounds for such query problems grow very rapidly as the dimension increases, either in terms of query time or space, most research has focused on approximate solutions. There has been a great deal of work on approximate proximity searching in spaces of very high dimension [5, 19, 32, 34] and in general metric spaces [5, 33, 39, 40]. Nonetheless, there are many important applications that naturally reside in real spaces of relatively low dimensions.

In this paper, we consider approximate nearest-neighbor searching for a query point against a discrete set of line segments in $\mathbb{R}^d$, where d is a fixed constant. We are given a set S of n disjoint line segments in $\mathbb{R}^d$. The distance from any point $q \in \mathbb{R}^d$ to a segment s, denoted $\text{dist}(q, s)$, is the minimum Euclidean distance between q and any point of s. For $\varepsilon > 0$, a segment $s' \in S$ is an *ε-approximate nearest neighbor* (*ε-ANN*) of q if $\text{dist}(q, s')$ is within a factor of $1 + \varepsilon$ of the distance to q's closest segment in S. Given S and $\varepsilon > 0$, the objective is to construct a data structure so that given any $q \in \mathbb{R}^d$, it is possible to compute an ε-ANN of q efficiently. We refer to this problem as *segment-ANN*.

© Ahmed Abdelkader and David M. Mount;
licensed under Creative Commons License CC-BY 4.0



Clearly, nearest-neighbor searching with respect to segments is at least as hard as for point sets, and there are quadratic worst-case lower bounds in the exact and approximate settings [6,31]. The difficulty of the problem can be appreciated by considering the increased complexity of the Voronoi diagram of a set of lines or line segments, which is not fully understood to date [14,16]. Recall that such Voronoi diagrams consist of cells bounded by hyperplanes and algebraic surfaces of constant degree, and hence are generally nonconvex; see [36,49] for the computation of Voronoi diagrams of line segments in $\mathbb{R}^2$. A set of just three straight lines in $\mathbb{R}^3$ suffices to induce a highly intricate Voronoi diagram [27]. Better bounds are known in restricted scenarios, for example, by bounding the number of orientations [26,37] or working with a polyhedral distance function [20,38].

Our work follows in this tradition with the main motivation of developing a better understanding of proximity searching among more complex objects than discrete sets of points. We are particularly interested in distance functions whose rate of change is much larger in some directions compared to others. This sort of behavior is characterized by the notion of *anisotropy*, which can be defined for smooth convex functions as the ratio of the largest to the smallest eigenvalues of the Hessian matrix at the point in question. Line segments are perhaps the simplest objects inducing such distance functions. This type of proximity searching against linear and affine subspaces has recently been applied to problems in pattern recognition [17,51] and active learning [50]. A notion of *direction-sensitive* distances in the plane has been studied in [3].

In this paper we present a new data structure for segment-ANN. Our data structure is an AVD-style data structure [9,12,13,31]. By this we mean that it employs a hierarchical subdivision of space (a covering in our case) by elements of constant complexity (ellipsoids in our case). At the leaf level of the hierarchy, each element stores a *representative segment* of S that is an ε-ANN for any query point lying within the element. Queries are answered by a simple descent through the hierarchy, reporting the representative of the leaf-level element. Up to now, AVD structures have relied on quadtree-based subdivisions. A novel feature of our approach is that the elements are anisotropic, where their shapes are sensitive to the local distribution of segments. The advantages of such an approach are illustrated intuitively in Figure 1. Our approach is inspired by recent progress on the use of anisotropic covering elements based on Macbeath regions [15,42] used in convex approximation [1,7,8,10,11].

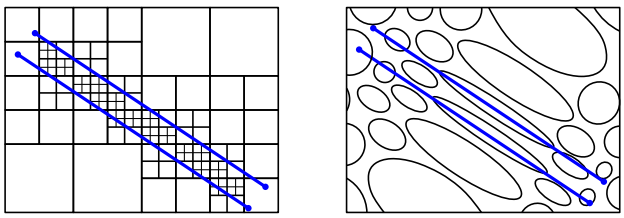


Figure 1 Approximation using isotropic (quadtree) elements compared to anisotropic elements.

Our input consists of a set S of n pairwise disjoint line segments in $\mathbb{R}^d$. Define the *spread*, denoted $\Delta(S)$ to be $\mathrm{diam}(S)/\delta_{\min}(S)$, where $\mathrm{diam}(S)$ is the diameter of the set S (the maximum distance between any two points lying on these segments), and $\delta_{\min}$ is the minimum distance between any two segments. Since S will be fixed throughout, we will just refer to this as Δ. Here is our main result.

▶ **Theorem 1.** *Given a set S of n disjoint line segments in $\mathbb{R}^d$ of spread Δ and $\varepsilon > 0$, there exists a data structure that can answer ε-ANN queries in time $O(\log(\max(n,\Delta)/\varepsilon))$ using $O((n^2/\varepsilon^d)\log\frac{\Delta}{\varepsilon})$ storage.*

Note that because the line segments are disjoint and the dimension is constant, $n = O(\Delta^d)$, the query time bound can be simplified to $O(\log \frac{\Delta}{\varepsilon})$.

The most closely related works to ours are segment-ANN data structures by Mahabadi [43] and Agarwal, Rubin, and Sharir [2]. Mahabadi's solution is based on reducing segment-ANN to point-ANN through a combination of reductions. These reductions produce $n^{O(1)}$ point-ANN modules, where each module involves $O(n/\varepsilon^{O(1)})$ points. The space bounds obtained are inferior to ours in terms of n and ε, and while individual modules can be solved within the AVD model, the overall data structure is not in this model. Agarwal, Rubin, and Sharir [2] consider the more general problem of ANN queries against k-flats in $\mathbb{R}^d$. As in our case, d is assumed to be constant. They solve the problem by approximating the Euclidean ball with a polyhedron distance function of complexity $1/\varepsilon^{O(1)}$ [18], and they show that it is possible to compute nearest neighbors exactly among k-flats with respect to the induced polyhedral distance function through the use of multi-level partition trees. In the case of line segments, their approach provides polylogarithmic query time with $n^2(\log(n)/\varepsilon)^{O(1)}$ storage, but the approach makes critical use of the fact that the objects are (infinite) flats. As with Mahabadi's result, there is no dependence on the spread. There are also works that consider the problem in its dual form, where the data set consists of points and the query is a k-flat [2, 4, 44].

Our data structure has a number of notable features. First, it is in the AVD model (which partially answers an open problem posed by Agarwal, Rubin, and Sharir [2]. The query algorithm is almost trivial, involving a descent through a rooted directed acyclic graph (DAG) of constant degree. The decision of which neighbor to visit next is just a membership test for an ellipsoid. By abandoning the quadtree-based approaches used in prior AVD solutions, we demonstrate how to exploit the anisotropic nature of the nearest-neighbor distance function to obtain a space-efficient hierarchical spatial decomposition.

The remainder of the paper is organized as follows. Section 2 formalizes the notion of anisotropy by examining the differential properties of the distance to the segments. Section 3 introduces the notion of a *capsule*, the basic shape upon which our data structure is built and introduces the relevant properties of these objects. Section 4 presents our ANN search structure, and finally Section 5 analyzes its storage requirements.

2 Exposing Anisotropy

In this section, we formally characterize how the distance function associated by a set of line segments naturally induces a Riemannian metric whose metric tensor is anisotropic; see Figure 2. This characterization underpins the design of our data structure which draw inspiration from classical constructions in convex optimization. For the sake of efficiency, the construction of our data structure will be based on a simpler approach, and this section may be skimmed without hampering the understanding of the material that follows.

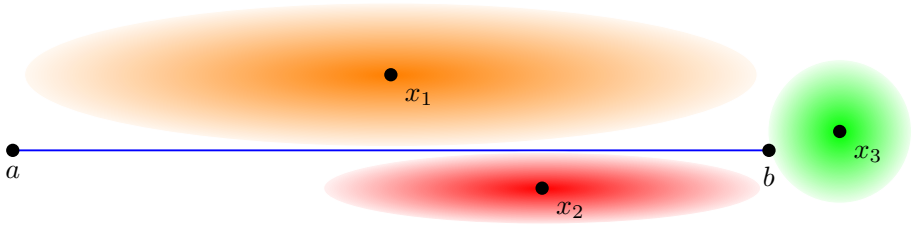


Figure 2 Demonstrating the local tensors induced by a segment $\overline{ab}$ as defined in Equation 5.

Given a set of pairwise-disjoint segments $S = \{s_1, \ldots, s_n\}$, denote by ℓ_i the line supporting $s_i = \overline{a_i b_i}$ parallel to the unit vector v_i. We define the distance functions at any $x \in \mathbb{R}^d$ as

$$D_i(x) \;=\; \begin{cases} D_i^{\ell}(x), & \text{if } x^{\perp} \in \operatorname{int}(s_i), \\ D_i^{\bullet}(x), & \text{otherwise,} \end{cases} \tag{1}$$

where D_i^{ℓ} is half the squared distance to the line ℓ_i, and $D_i^{\bullet}(x) = \min\{D^{a_i}(x), D^{b_i}(x)\}$ with D^{a_i} and D^{b_i} being half the squared distance to the endpoints a_i and b_i, respectively, $x^{\perp}$ the projection of x onto ℓ_i, and $\operatorname{int}(s_i)$ the interior of s_i. As is common for similar definitions, we work with squared distances and introduce the $\frac{1}{2}$ factor to simplify the resulting derivatives.

For every $x \in \mathbb{R}^d$, we seek a definition of a *local tensor* to effectively consolidate the two cases in the definition of $D_i(x)$ per Equation 1. Using such local tensors, we can define a *local descriptor*, e.g., an ellipsoid, whose shape describes the rate of change of the distance function D_i in the neighborhood of x. We achieve this by first examining the Hessian of the distance functions defining $D_i(x)$ for each segment in isolation. Then, we consider the consolidation of all distance functions as needed for nearest-neighbor searching.

2.1 Distance Hessians

For a fixed point $p \in \mathbb{R}^d$, the associated distance takes the form

$$D^p(x) \;=\; \frac{1}{2}\|x - p\|^2 \;=\; \frac{1}{2}\sum_{i=1}^{d}(x_i - p_i)^2, \quad \text{for which } \nabla^2 D^p = I, \tag{2}$$

where ∇^2 denotes the function's Hessian and I is the identity matrix. For a fixed line $\ell = \{p + tv \mid t \in \mathbb{R}\}$, with $p, v \in \mathbb{R}^d$ and $\|v\| = 1$, the distance takes the form

$$D^{\ell}(x) \;=\; \frac{1}{2}\|x - x^{\perp}\|^2 \;=\; \frac{1}{2}\sum_{i=1}^{d}\left((x_i - p_i) - \langle x - p, v\rangle v_i\right)^2,$$

where $x^{\perp}$ is the projection of x onto ℓ. We proceed to compute the Hessian $\nabla^2 D^{\ell}$ as follows.

$$\frac{\partial D^{\ell}}{\partial x_k} \;=\; (x_k - p_k) - \langle x - p, v\rangle v_k, \qquad \frac{\partial^2 D^{\ell}}{\partial x_k^2} \;=\; 1 - v_k^2, \qquad \frac{\partial^2 D^{\ell}}{\partial x_k \partial x_n} \;=\; -v_k v_n,$$

$$\nabla^2 D^{\ell} \;=\; I - vv^{\intercal}. \tag{3}$$

It is easy to verify that v is an eigenvector of $\nabla^2 D^{\ell}$ with eigenvalue 0. Letting T be any rotation matrix such that $Tv = [1, 0, \ldots, 0]^{\intercal}$, we obtain

$$\nabla^2(D^{\ell} \circ T) \;=\; \begin{bmatrix} 0 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 \end{bmatrix}.$$

Noting that $\nabla^2 D^{\ell} = T^{-1}\nabla^2(D^{\ell} \circ T)T^{-1}$, and that eigenvalues are invariant under change of basis, the remaining eigenvalues of $\nabla^2 D^{\ell}$ are all equal to 1, where the corresponding eigenvectors can be chosen as any basis of the subspace orthogonal to v. This form of the Hessian reflects the constancy of the distance along trajectories parallel to the line.

2.2 Local Tensors and Ellipsoids

Per the previous subsection, the Hessian $\nabla^2 D_i^\ell$, of the distance to a line ℓ_i, is rank-deficient with v_i an eigenvector with eigenvalue 0 and all remaining eigenvalues equal to 1. We remedy this deficiency by defining the local tensor as

$$\mathbb{H}_i(x) \;=\; \frac{1}{D_i(x)}\nabla^2 D_i^\ell + \frac{1}{D_i^\bullet(x)} v_i v_i^\intercal. \tag{4}$$

By construction, the local tensor $\mathbb{H}_i(x)$ has a single eigenvalue equal to $1/D_i^\bullet(x)$ with all remaining $d-1$ eigenvalues equal to $1/D_i(x)$. The anisotropy of the distance function D_i reflected by this local tensor at x is equal to the ratio of the maximum of $D_i^\bullet(x)$ and $D_i(x)$ to their minimum. As x moves along any smooth trajectory, this anisotropy varies continuously between of 1 and ∞. We use the tensor $\mathbb{H}_i(x)$ to define the ellipsoid

$$\mathcal{E}_i(x) \;=\; \left\{ y \in \mathbb{R}^d \;\middle|\; \frac{1}{2}(y-x)^\intercal \mathbb{H}_i(x)(y-x) \le 1 \right\}. \tag{5}$$

(See Figure 2 for examples.) Observe that as $D_i^\bullet$ becomes larger, the eigenvalue associated with v_i becomes smaller, and the ellipsoid $\mathcal{E}_i(x)$ extends further in the direction of v_i. On the other hand, as $D_i^\ell(x)$ approaches $D_i^\bullet(x)$, $\mathbb{H}_i(x)$ approaches a scaled identity matrix, and the ellipsoid $\mathcal{E}_i(x)$ becomes more spherical.

One approach to account for the influence of all n segments is to define a *blended tensor* at every $x \in \mathbb{R}^d$, $\mathbb{H}(x) = \sum_{i=1}^n \mathbb{H}_i(x)$ along with an induced *local norm*[1] and a corresponding *local ellipsoid* acting as a *metric ball* at x

$$\widetilde{\mathcal{E}}(x) \;=\; \left\{ y \in \mathbb{R}^d \mid \|y-x\|_x^2 \le 1 \right\}, \quad \text{where} \quad \|y-x\|_x^2 \;=\; \frac{1}{2}(y-x)^\intercal \mathbb{H}(x)(y-x). \tag{6}$$

Alternatively, we may directly bound the relative change of *all* distance functions in the neighborhood of x by restricting attention to the cell [2]

$$\widehat{\mathcal{E}}(x) \;=\; \bigcap_{i=1}^n \mathcal{E}_i(x). \tag{7}$$

The next lemma formalizes the relationship between the ellipsoids $\widetilde{\mathcal{E}}(x)$ and the cells $\widehat{\mathcal{E}}(x)$.[3]

▶ **Lemma 2.** *For any set of n segments and any point $x \in \mathbb{R}^d$, we have the inclusions*

$$\widetilde{\mathcal{E}}(x) \;\subseteq\; \widehat{\mathcal{E}}(x) \;\subseteq\; \widetilde{\mathcal{E}}^{\sqrt{n}}(x), \quad \textit{where the superscript denotes the central scaling about } x.$$

Proof. The first inclusion is immediate. For the second inclusion, observe that any $y \in \widehat{\mathcal{E}}(x)$ satisfies $\max_i \frac{1}{2}(y-x)^\intercal \mathbb{H}_i(x)(y-x) \le 1$. Hence, $\|y-x\|_x^2 \le n$, implying $y \in \widetilde{\mathcal{E}}^{\sqrt{n}}(x)$. ◀

Unfortunately, each of those two approaches has its own drawbacks. While the blended tensors $\mathbb{H}(x)$ are easier to compute, the corresponding ellipsoids $\widetilde{\mathcal{E}}(x)$ are unnecessarily small. On the other hand, the cells $\widehat{\mathcal{E}}(x)$ can retain a suitable size but are difficult to construct. This motivates an alternative, and more geometric, definition of a more efficient shape primitive.

---

[1] See [45, 47] for related derivations of Riemannian metrics from local tensors.

[2] This can be seen by recognizing $\mathbb{H}_i(x)$ as the Hessian of a closely related function derived from D_i, and writing its Taylor expansion about x for points within $\widetilde{\mathcal{E}}(x)$.

[3] Readers familiar with the *Dikin ellipsoid* from convex optimization will recognize the similarities with the local ellipsoids $\widetilde{\mathcal{E}}(x)$. It is well-known that Macbeath regions and Dikin ellipsoids are related by a similar inclusion as in Lemma 2: for a polytope K defined as the intersection of m halfspaces and a point $x \in K$, the Macbeath region $K \cap (2x - K)$ contains the Dikin ellipsoid at x and is contained in its $\sqrt{m}$ expansion; see, e.g., [41, 48]. While Dikin ellipsoids are derived from barrier functions [46, 52], we derive our ellipsoids from the Euclidean distance functions.

3 Anisotropic Space Covers

Building upon the derivations in the previous section, we propose a simple primitive shape for constructing a hierarchical space covering, which will be amenable to computation and analysis. Recall that S is a set of n disjoint line segments in $\mathbb{R}^d$, each defined by its two endpoints. Fix a segment $s = \overline{ab} \in S$, and let $r > 0$ be a given distance parameter, to be defined later. Recall that for any $x \in \mathbb{R}^d$, its distance to segment s is denoted by $\mathrm{dist}(x, s)$.

For any point $x \in \mathbb{R}^d$, we define a convex and centrally-symmetric subregion centered about x, called a *capsule* and denote by $\mathcal{C}_s(x, r)$. If the closest point to x on s is an endpoint, then $\mathcal{C}_s(x, r)$ is simply the ball centered at x with radius $\max(r, \mathrm{dist}(x, s))$. Otherwise, $\mathcal{C}_s(x, r)$ is defined as follows. First, construct the infinite cylinder of radius $\max(r, \mathrm{dist}(x, s))$ with axis parallel to s and passing through x. Consider a ball centered at x whose radius is $\max(r, \min(\|x - a\|, \|x - b\|)$, where a and b are the endpoints of s. The capsule is the intersection of this cylinder and ball (see Figure 3(a) and (b)).

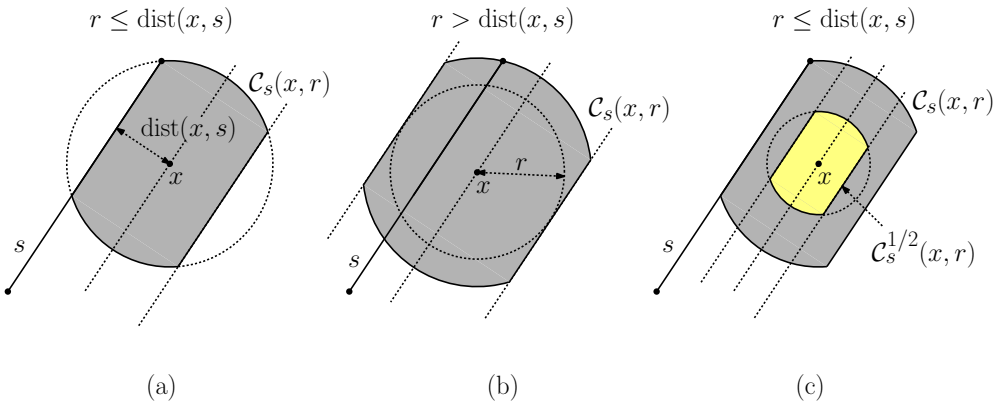


Figure 3 (a) The capsule $\mathcal{C}_s(x, r)$ at x for segment s for $r \le \mathrm{dist}(x, s)$, (b) for $r > \mathrm{dist}(x, s)$, and (c) the shrunken capsule $\mathcal{C}_s^{1/2}(x, r)$ for $r \le \mathrm{dist}(x, s)$.

We define the *capsule* associated with x for the set S of all segments as $\mathcal{C}_S(x, r) = \bigcap_{s \in S} \mathcal{C}_s(x, r)$. Since S will be fixed throughout, we will omit this subscript henceforth. Clearly, $\mathcal{C}(x, r)$ is also convex and centrally symmetric about x. We start by showing that capsules are closely related to the Hessian-based ellipsoids defined in Eq. (5).

▶ **Lemma 3.** *For all $x \in \mathbb{R}^d$ and any $r \le \min_i \mathrm{dist}(x, s_i)$,*

$$\widehat{\mathcal{E}}(x) \;\subseteq\; \mathcal{C}(x, r) \;\subseteq\; \widehat{\mathcal{E}}^{\sqrt{2}}(x). \tag{8}$$

Proof. By the definition of the local tensor per Equation 4, we may express the capsule as

$$\mathcal{C}_{s_i}(x, r) \;=\; \left\{ y \in \mathbb{R}^d \;\middle|\; \max_i \left(\frac{(y-x)^\intercal \nabla^2 D_i^\ell (y-x)}{\max\{r^2, 2 \cdot D_i(x)\}}, \frac{(y-x)^\intercal (y-x)}{\max\{r^2, 2 \cdot D_i^\bullet(x)\}} \right) \le 1 \right\}, \tag{9}$$

where, in contrast to Equation 4, we replaced $v_i v_i^\intercal$ in the second term by just the identity matrix to make the shape nicer. Recognizing the definition of both $\widehat{\mathcal{E}}$ and the capsule $\mathcal{C}$ as the intersection of n subsets, it suffices to establish the following claim: for all segments s_i, and any $r \le \mathrm{dist}(x, s_i)$, we have that

$$\mathcal{E}_i(x) \;\subseteq\; \mathcal{C}_{s_i}(x, r) \;\subseteq\; \mathcal{E}_i^{\sqrt{2}}(x). \tag{10}$$

Observing that $D_i(x) = \text{dist}^2(x, s_i)/2$ and $D_i(x) \leq D_i^\bullet(x)$, we see that r^2 cannot dominate in either of the denominators in Equation 9. In addition, for any $y \in \mathbb{R}^d$ we may write $y - x = \alpha v + \beta u$, where u is a unit vector orthogonal to v. We obtain

$$y \in \mathcal{E}_i(x) \implies \frac{1}{2}(y-x)^\intercal \mathbb{H}_i(x)(y-x) = \frac{1}{2}\left(\frac{\beta^2}{D_i(x)} + \frac{\alpha^2}{D_i^\bullet(x)}\right) \leq 1.$$

From the above, the first term in Equation 9 is at most 1. For the second term, observe that $\beta^2/D_i^\bullet(x) \leq \beta^2/D_i(x)$. By making this substitution, we find that $\frac{1}{2}(\beta^2+\alpha^2)/D_i^\bullet(x) = \frac{1}{2}\|y-x\|^2/D_i^\bullet(x) \leq 1$, implying the second term in Equation 9 is at most 1 as well. It follows that $y \in \mathcal{C}_{s_i}(x, r)$, establishing the first inclusion. For the second inclusion, observe that for all $y \in \mathcal{C}_{s_i}(x, r)$ we have that $\frac{1}{2}(y-x)^\intercal \mathbb{H}_i(q)(y-x) \leq 2$. This holds as both terms in Equation 9 are at most 1, and we have $(y-x)^\intercal vv^\intercal(y-x) \leq (y-x)^\intercal(y-x)$ for all unit vectors v. Using Equation 10, the proof follows by intersection over all $i \in [n]$. ◀

For the purposes of distance approximation, we work with a scaled version of these capsules which we denote by the superscript $\mathcal{C}^\lambda$ for a scale factor λ. The scaled version of each region is the central scaling around x by λ. When $\lambda < 1$, we say that the regions are *shrunken* (see Figure 3(c)).

Capsules enjoy a number of useful properties, similar to the Macbeath regions in the context of convex bodies; see [1, 15]. In particular, capsules satisfy the following *expansion-containment property*, which states that whenever two shrunken capsules overlap, a constant factor expansion of one contains the other.

▶ **Lemma 4.** *Let S be a set of disjoint line segments and $\lambda \in (0, 1)$ be a scale factor. For any $x, y \in \mathbb{R}^d$, if $\mathcal{C}^\lambda(x, r) \cap \mathcal{C}^\lambda(y, r) \neq \emptyset$ then $\mathcal{C}^\lambda(y, r) \subseteq \mathcal{C}^{\alpha\lambda}(x, r)$, where $\alpha = \frac{3+\lambda}{1-\lambda}$.*

3.1 Local Feature Size

In order to use capsules for space covering, we need a principled way to select the distance parameter r. We define the *local feature size* (*LFS*) at $x \in \mathbb{R}^d$ as the distance from x to the *second-nearest* segment:

$$\phi(x) = \min_{i,j \in \binom{n}{2}} \max\{\text{dist}(x, s_i), \text{dist}(x, s_j)\}, \tag{11}$$

where we assume $n \geq 2$. It is easy to see that ϕ is 1-Lipschitz, and the following lemma further quantifies the sensitivity of capsules to the distances to the set of line segments. In particular, all points within a shrunken capsule have comparable local feature size.

▶ **Lemma 5.** *For all $z \in \mathcal{C}^\lambda(x, \phi(x))$, where $0 < \lambda < 1$, $\phi(z) \in [1-\lambda, 1+\lambda] \cdot \phi(x)$.*

Proof. Let $r = \phi(x)$, and denote by s_1 and s_2 the nearest and second-nearest segments to x, respectively. Observing that $r = \text{dist}(x, s_2)$, the interior of $\mathcal{C}(x, r)$ cannot intersect any segment except for s_1. To obtain the lower bound, we bound the distance from any $z \in \mathcal{C}^\lambda(x, r)$ to $\partial\mathcal{C}(x, r)$.

Recalling the construction of capsules, for any segment s, as an infinite cylinder restricted within a ball, we define the *spine* of the capsule at x with respect to s as the projection of s onto the axis of the cylinder intersected with the capsule, or just x if the capsule is a ball; see Figure 4.

For any $z \in \mathcal{C}_{s_2}(x, r)$, define z' as the projection of z onto the spine. By construction, $\text{dist}(z, z') \leq r$ and $\text{dist}(z', s_2) = r$, for all $z \in \mathcal{C}_{s_2}(x, r)$ and z' on the spine. Upon shrinking, we obtain $\text{dist}(z, z') \leq \lambda r$. In addition, $\text{dist}(z, \partial\mathcal{C}(x, r)) \geq (1-\lambda)r$. This lower bound is

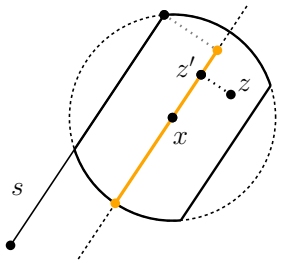


Figure 4 The spine construction used in Lemma 5.

obvious for the shrunken cylindrical shell. For the spherical caps, we note that they are at least as far from x as the cylindrical shell, so the spherical caps of the shrunken capsule are displaced by at least the same amount as the shrunken cylindrical shell. Since only s_1 may be closer to z than $\partial\mathcal{C}(x,r)$, we have $\phi(z) \geq (1-\lambda)r$.

For the upper bound, we consider $\mathcal{C}_{s_1}(x,r)$ in addition to $\mathcal{C}_{s_2}(x,r)$. For any $z \in \mathcal{C}_{s_1}(x,r) \cap \mathcal{C}_{s_2}(x,r)$, let z' and z'' denote the projections of z onto the spines of $\mathcal{C}_{s_1}(x,r)$ and $\mathcal{C}_{s_2}(x,r)$, respectively. By the above derivations $\text{dist}(z,s_2) \leq \text{dist}(z,z'') + \text{dist}(z'',s_2) \leq (1+\lambda)r$. We also have $\text{dist}(z',s_1) = \text{dist}(x,s_1) \leq \phi(x) = r$, implying $\text{dist}(z,s_1) \leq \text{dist}(z,z') + \text{dist}(z',s_1) \leq (1+\lambda)r$. It follows that,

$$\phi(z) \;\leq\; \max\{\text{dist}(z,s_1), \text{dist}(z,s_2)\} \;\leq\; (1+\lambda)r,$$

as desired. ◀

From the above lemma, it is easy to obtain an approximation of nearest-neighbor distances using the segment closest to the center point of the capsule as a representative. This qualifies capsules to serve as cells for a type of approximate Voronoi diagram (AVD) data structure [13,31].

▶ **Lemma 6.** *For any $z \in \mathcal{C}^\lambda(x,\phi(x))$ with $0 < \lambda < 1$, $\text{dist}(z,s_x) \leq \frac{1+\lambda}{1-\lambda}\,\text{dist}(z,s_z)$, where s_x is the closest segment to x and s_z is the closest segment to z.*

Proof. Assume s_z is distinct from s_x, for the assertion holds trivially otherwise. In the same notation we used to prove Lemma 5, let $r = \phi(x)$, and denote by s_x and s'_x the nearest and second-nearest segments to x, respectively. Observing that $r = \text{dist}(x,s'_x)$, the interior of $\mathcal{C}(x,r)$ cannot intersect any segment except for s_x. As seen in the proof of Lemma 5, we have that $\text{dist}(z,s_z) \geq \text{dist}(z,\partial\mathcal{C}(x,r)) \geq (1-\lambda)r$ while $\text{dist}(z,s_x) \leq (1+\lambda)r$. ◀

As an immediate corollary, we have the following.

▶ **Corollary 7.** *For $0 < \varepsilon \leq 1$ and any $z \in \mathcal{C}^\lambda(x,\phi(x))$ with $0 < \lambda \leq \frac{\varepsilon}{3}$, the nearest neighbor of x is a $(1+\varepsilon)$-approximate nearest neighbor of z.*

Capsules enjoy a number of properties (described later in this section) that make them suitable for forming hierarchical covers of space. However, as the intersection of n cylinders and/or balls, capsules can have high combinatorial complexity. For this reason, we use their John ellipsoids in our data structure instead, as in related data structures based on Macbeath regions [1,10]. For $x \in \mathbb{R}^d$ and positive scalars λ and r, define $E^\lambda(x,r)$ as the maximum volume ellipsoid enclosed within $\mathcal{C}^\lambda(x,r)$. By John's Theorem [35], $E^\lambda(x,r) \subseteq \mathcal{C}^\lambda(x,r) \subseteq E^{\lambda\sqrt{d}}(x,r)$. Hence, up to constant factors, these *capsule ellipsoids* can serve as low-complexity proxies for capsule. Our construction makes use of two particular constant scale factors independent of ε, $0 < \lambda'' < \lambda' < 1$. For any x and r, define $E''(x,r) = E^{\lambda''}(x,r)$ and $E'(x,r) = E^{\lambda'}(x,r)$.

3.2 Net-Like Properties for Capsules

In this section, present a number of properties of capsules, demonstrating that they possess similar properties to nets, which arise in the study of metric spaces [33, 39, 40]. While we prove these results for capsules, they all hold for capsule ellipsoids, subject to an adjustment of constant factors.

Our first result is a utility that relates two methods for growing capsules, first by expanding the distance parameter and second by applying a scale factor.

▶ **Lemma 8.** *For any $\gamma \geq 1$, $\mathcal{C}(x, \gamma r) \subseteq \mathcal{C}^{\gamma}(x, r)$, and for $0 < \lambda \leq 1$, $\mathcal{C}^{\lambda}(x, r) \subseteq \mathcal{C}(x, \lambda r)$.*

Proof. For each segment s, the radii used in $\mathcal{C}_s(x, r)$ are of the form $\max(r, \mathrm{dist}(x, s))$. Clearly, $\max(\gamma r, \mathrm{dist}(x, s)) \leq \gamma \cdot \max(r, \mathrm{dist}(x, s))$. The other inequality is similar. ◀

The next lemma bounds the growth in the volume of capsules upon scaling. The first part follows directly from the fact that capsules are full dimensional and convex, and the second part follows from this in combination with Lemma 8.

▶ **Lemma 9.** *For any set of disjoint line segments $S \subseteq \mathbb{R}^d$ and an arbitrary $x \in \mathbb{R}^d$:*

(i) *For $\lambda > 0$, $\mathrm{vol}(\mathcal{C}^{\lambda}(x, r)) = \lambda^d \cdot \mathrm{vol}(\mathcal{C}(x, r))$.*

(ii) *For $\beta \geq 1$, $\mathrm{vol}(\mathcal{C}(x, \beta r)) \leq \beta^d \cdot \mathrm{vol}(\mathcal{C}(x, r))$.*

Next, we derive a packing bound on the number of pairwise interior-disjoint capsules that may fit within a larger capsule.

▶ **Lemma 10.** *Given a set of disjoint line segments $S \subseteq \mathbb{R}^d$, $r \geq 0$, and two constant scale factors $0 < \lambda_p < \lambda_c < 1$, let $Y \subset \mathbb{R}^d$ denote a set of points such that the associated regions $\mathcal{C}^{\lambda_p}(y, r)$, with $y \in Y$, are disjoint. Then, for any $x \in \mathbb{R}^d$ and $\beta \geq 1$, the number of regions in $R_Y = \{\mathcal{C}^{\lambda_c}(y, r) \mid y \in Y\}$ that intersect $\mathcal{C}^{\lambda_c}(x, \beta r)$ is $O(\beta^d)$.*

Proof. Fix a $y \in Y$ such that $\mathcal{C}^{\lambda_c}(x, \beta r) \cap \mathcal{C}^{\lambda_c}(y, r) \neq \emptyset$. As $\mathcal{C}^{\lambda_c}(y, r) \subseteq \mathcal{C}^{\lambda_c}(y, \beta r)$, we also have that $\mathcal{C}^{\lambda_c}(x, \beta r) \cap \mathcal{C}^{\lambda_c}(y, \beta r) \neq \emptyset$. Applying Lemma 4 (with the roles of x and y swapped), we obtain $\mathcal{C}^{\lambda_c}(x, \beta r) \subseteq \mathcal{C}^{\alpha\lambda_c}(y, \beta r)$, where $\alpha = \frac{3+\lambda_c}{1-\lambda_c} > 1$. Lemma 9 yields

$$\begin{aligned}\mathrm{vol}(\mathcal{C}^{\lambda_p}(y, r)) &= \left(\frac{\lambda_p}{\lambda_c}\right)^d \mathrm{vol}(\mathcal{C}^{\lambda_c}(y, r)) \geq \left(\frac{\lambda_p}{\lambda_c \beta}\right)^d \mathrm{vol}(\mathcal{C}^{\lambda_c}(y, \beta r)) \\ &= \left(\frac{\lambda_p}{\lambda_c \alpha \beta}\right)^d \mathrm{vol}(\mathcal{C}^{\alpha\lambda_c}(y, \beta r)) \geq \left(\frac{\lambda_p}{\lambda_c \alpha \beta}\right)^d \mathrm{vol}(\mathcal{C}^{\lambda_c}(x, \beta r)).\end{aligned}$$

By packing, the number of regions of R_Y intersecting $\mathcal{C}^{\lambda_c}(x, \beta r)$ is $O\big(\big(\frac{\lambda_c \alpha \beta}{\lambda_p}\big)^d\big)$. The result follows since λ_c, λ_p, and α are all constants. ◀

Turning our attention to radius assignment through the local feature size ϕ, we show that expansion-containment still holds.

▶ **Lemma 11.** *Given two points $x, y \in \mathbb{R}^d$ and $0 < \lambda < 1$, if $\mathcal{C}^{\lambda}(x, \phi(x)) \cap \mathcal{C}^{\lambda}(y, \phi(y)) \neq \emptyset$, then $\mathcal{C}^{\lambda}(y, \phi(y)) \subseteq \mathcal{C}^{\beta\lambda}(x, \phi(x))$ for a constant $\beta = \frac{(3+\lambda)(1+\lambda)}{(1-\lambda)^2}$.*

Proof. Assuming first that $\phi(y) \leq \phi(x)$, $\mathcal{C}^{\lambda}(y, \phi(y)) \subseteq \mathcal{C}^{\lambda}(y, \phi(x))$ and thus

$$\mathcal{C}^{\lambda}(x, \phi(x)) \cap \mathcal{C}^{\lambda}(y, \phi(y)) \neq \emptyset \implies \mathcal{C}^{\lambda}(x, \phi(x)) \cap \mathcal{C}^{\lambda}(y, \phi(x)) \neq \emptyset.$$

Applying Lemma 4 with $r = \phi(x)$, implies that $\mathcal{C}^{\lambda}(y, \phi(y)) \subseteq \mathcal{C}^{\alpha\lambda}(x, \phi(x))$.

Otherwise, $\phi(x) \leq \phi(y)$, and by applying Lemma 5 twice with any $z \in \mathcal{C}^\lambda(x, \phi(x)) \cap \mathcal{C}^\lambda(y, \phi(y))$, we obtain

$$\phi(y) \;\leq\; \frac{1}{1-\lambda}\phi(z) \;\leq\; \frac{1+\lambda}{1-\lambda}\phi(x).$$

By Lemma 8, $\mathcal{C}^\lambda(x, \gamma\,\phi(x)) \subseteq \mathcal{C}^{\gamma\lambda}(x, \phi(x))$, where $\gamma = \frac{1+\lambda}{1-\lambda} > 1$. Therefore

$$\mathcal{C}^\lambda(x, \phi(x)) \cap \mathcal{C}^\lambda(y, \phi(y)) \neq \emptyset \implies \mathcal{C}^\lambda(x, \gamma \cdot \phi(x)) \cap \mathcal{C}^\lambda(y, \gamma \cdot \phi(x)) \neq \emptyset.$$

Applying Lemma 4 with $r = \gamma\,\phi(x)$, implies that $\mathcal{C}^\lambda(y, \phi(y)) \subseteq \mathcal{C}^{\alpha\gamma\lambda}(x, \phi(x))$. ◀

4 The ANN Data Structure

In this section, we apply the results of the previous section to present our data structure for answering ε-ANN queries. Again, $S = \{s_1, \ldots, s_n\}$ is a set of n disjoint line segments in $\mathbb{R}^d$, and $\varepsilon > 0$ is the approximation parameter. Let $B(S)$ be a minimum volume Euclidean ball that contains S, and let $B^+(S)$ denote a concentric expansion of $B(S)$ about its center by a factor of $1 + 2/\varepsilon$. It is easy to see that if the query point q lies outside of $B^+(S)$, any segment may be reported as an ε-ANN of q. Thus, for the rest of the construction, we focus on query points lying within $B^+(S)$. Let x_0 and r^+ denote the center and radius of this ball, respectively. Clearly, $r^+ = \Theta(\mathrm{diam}(S)/\varepsilon)$. Let $\delta(S)$ denote the minimum distance between any two segments of S. Observe that for every point $x \in B^+(S)$, its local feature size, $\phi(x)$, is at least $\delta(S)/2$.

Here is a high-level overview of the data structure. It consists of a rooted directed acyclic graph (DAG), which is based on covering $B^+(S)$ with a hierarchy of capsule ellipsoids of exponentially diminishing scales. The DAG is organized in levels, with a single root node at level zero whose associated capsule ellipsoid contains $B^+(S)$. For $i \geq 0$, the capsule ellipsoids associated with the nodes of level i employ the scale parameter $r_i = r^+/2^i$, and thus successive levels are more refined. Each level of the DAG will be associated with a collection of capsule ellipsoids that cover $B^+(S)$. In particular, each node at level i of the DAG stores a point $x \in B^+(S)$, and the associated capsule ellipsoid, denoted $E'(x)$, centered at this point is defined to be the shrunken ellipsoid $E'(x, r_i)$ with respect to S. (The shrinking factor $\frac{1}{2}$ can be taken to be any constant smaller than 1, subject to an adjustment in the various constant factors used in our construction.) Each node at level i will either be declared to be a leaf, or it will be linked to those nodes at level $i + 1$ whose capsule ellipsoids it overlaps, which we call its *children*. (We will show that the out-degree of any node is a constant.)

We continue this refinement process until $r_i \leq \frac{\varepsilon}{3}\phi(x)$. The resulting terminal nodes are called *leaves*, and each stores the segment of S that is closest to the capsule center as its *representative*.

Queries are answered by a simple descent through this DAG. Assuming that the query point q lies within $B^+(S)$, we descend level by level through the DAG. On arriving at a non-leaf node, we inspect the ellipsoids of its children on level $i+1$. Their associated capsules cover $E'(x)$, and the search continues with any one of these children whose associated ellipsoid contains q. When the search arrives at a leaf node, the associated representative segment is returned as the answer to the query.

The DAG is constructed in a top-down manner, starting with the root of the DAG. The root capsule is $E'(x_0)$, where x_0 is the center of $B^+(S)$. We assert that this covers $B^+(S)$. To see this, observe that by definition, $E'(x, r)$ contains the ball of radius r centered at x, and hence the same holds for $E'(x, r)$. For $i = 1, 2, \ldots$, let U_i denote the portion of $B^+(S)$ that

is not covered by any of the leaves of the structure from prior levels. For any $x \in \mathbb{R}^d$, define $E''(x) = E''(x, r_i)$. Let X_i be any maximal set of points x within U_i such that the associated ellipsoids $E''(x)$ are pairwise disjoint. It follows from maximality and expansion-containment that the union of the expanded ellipsoids $E'(x)$ for $x \in X_i$ covers U_i. We create a node at level i of the DAG for each point $x \in X_i$, and we link each such node as a child of any non-leaf node from the previous level whose E' capsule ellipsoid (computed with respect to level $i-1$) it overlaps.

In the remainder of this section, we analyze the correctness, query time and storage requirements of this data structure. Our first two lemmas establish correctness and bound the depth of the data structure. Correctness follows from Corollary 7.

▶ **Lemma 12.** *Given a set S of line segments, the above search algorithm returns an ε-ANN among the segments of S for any query point $q \in B^+(S)$.*

Next we analyze the depth of the DAG.

▶ **Lemma 13.** *Given a set S of n disjoint segments in $\mathbb{R}^d$ with spread Δ, the ε-AVD structure described above has $O(\log(\max(n, \Delta)/\varepsilon))$ levels.*

Proof. Recall that $B(S)$ is the minimum Euclidean ball containing S and $B^+(S)$ is its expansion by $1 + \frac{2}{\varepsilon}$, and r^+ is its radius. Clearly, $r^+ = \Theta(\mathrm{diam}(S)/\varepsilon)$. Letting $\delta_{\min}$ denote the minimum distance between any pair of segments, $\Delta = \mathrm{diam}(S)/\delta_{\min}$. Clearly, for any point x, $\phi(x) \geq \delta_{\min}$. The refinement process terminates when the scale falls below $\frac{\varepsilon}{3}\phi(x) \geq \frac{\varepsilon}{3}\delta_{\min}$. Since the scale decreases by a factor of 2 with each level of the data structure, the total number of levels is

$$O\left(\log \frac{r^+}{\varepsilon\,\delta_{\min}}\right) = O\left(\log \frac{\mathrm{diam}(S)}{\varepsilon^2 \delta_{\min}}\right) = O\left(\log \frac{\Delta}{\varepsilon}\right),$$

as desired. Recall that the spread of a set of segments in $\mathbb{R}^d$ grows at least polynomially with n, therefore $\log n$ is $O(\log \Delta)$. ◀

Our next result bounds the number of children for each node.

▶ **Lemma 14.** *Each non-leaf node of the data structure has $O(1)$ children.*

Proof. Consider a node at some level i centered at a point x. Let $r = r_{i+1} = r_i/2$. This node's children consist of the nodes y of level $i+1$ whose ellipsoid $E'(y, r_{i+1})$ overlaps $E'(x, r_i)$. By construction, all such points y come from a set Y whose ellipsoids $E''(y, r_{i+1})$ are disjoint. By applying Lemma 10 in the elliptical setting, the number of such overlapping ellipsoids is $O(2^d) = O(1)$, given our assumption that the dimension d is fixed. ◀

Since each node of the DAG has constant degree, it follows that the overall query time is proportional to DAG's height, which is $O(\log(\Delta/\varepsilon))$. Finally, we bound the total space used by the data structure.

▶ **Lemma 15.** *Given a set S of n line segments in $\mathbb{R}^d$, the total storage required by the ε-AVD is $O((n^2/\varepsilon^d)\log\frac{\Delta}{\varepsilon})$.*

Proof. We distinguish between two transitions within the DAG structure. When the scale parameter r_i of a node x first falls below $\phi(x)$, we say that this is a basic leaf, and when it falls below $\frac{\varepsilon}{3}\phi(x)$ (the actual termination condition), we say it is a final leaf. Lemma 17 (presented in Section 5) states that the number of capsules at the basic leaf level that are charged to any pair of segments is $O(\log\frac{\Delta}{\varepsilon})$. Therefore, the total number basic leaves is $O(n^2 \log\frac{\Delta}{\varepsilon})$.

Observe that all of the points lying within $E'(x, \phi(x))$ share the same local-feature size values up to constant factors, and therefore, all the descendants of this node lie within the next $O(\log \frac{1}{\varepsilon})$ levels of the structure. They are all similarly shaped (up to constant factors), but their sizes are smaller by a factor of at least $\frac{\varepsilon}{3}$. By the disjointness of the shrunken E'' capsule ellipsoids, it follows that the number of descendants is $O(1/\varepsilon^d)$. Therefore, the total number of nodes in the DAG is $O((n^2/\varepsilon^d) \log \frac{\Delta}{\varepsilon})$. ◀

By combining the results of the previous lemmas, we obtain Theorem 1.

5 Storage Bounds for Capsules

Recall the definition of capsules from Section 3. The infinite cylindrical component in the definition of $\mathcal{C}(x, r)$ induced by a particular segment $s_i \in S$ will be denoted $\mathrm{Cyl}_i(x, r)$ and its radius will be denoted by $t_i(x)$.

For a given point $x \in \mathbb{R}^d$ and $r > 0$, we distinguish two such cylinders. Without loss of generality, let s_1 denote the nearest neighbor of x in S. Letting v_1 be a unit vector parallel to s_1 (the direction does not matter by central symmetry). Denote by ℓ_x the line passing through x in the direction of v_1. Using ℓ_x, we define the set of points p_i as the intersection of $\partial \mathrm{Cyl}_i(x, r)$ and ℓ_x such that $\langle v_1, p_i - x \rangle > 0$. Again, without loss of generality, let p_2 denote the closest intersection point to x, such that $\mathrm{Cyl}_2(x, r)$ is the cylinder generating p_2. We use the two cylinders $\mathrm{Cyl}_1(x, r)$ and $\mathrm{Cyl}_2(x, r)$ to sandwich the capsule $\mathcal{C}(x, r)$ between two simple shapes providing lower and upper bounds on its volume. The radii of $\mathrm{Cyl}_1(x, r)$ and $\mathrm{Cyl}_2(x, r)$ will be denoted by t_1 and t_2, respectively. By definition, $t_1 = r$. When r is chosen as the local feature size $\phi(x)$ at x, we also have $t_2 \geq r$.

The inner bounding volume $V^-(x, r)$ is defined as the double cone whose axis is ℓ_x and base is the $(d-1)$-dimensional disk of radius r centered at x orthogonal to ℓ_x, with two apexes at p_2 and (symmetrically about x) $2x - p_2$. The outer bounding volume $V^+(x, r)$ is defined as the cylinder with ℓ_x as axis whose radius is r and height is equal to the length of the projection of $\mathrm{Cyl}_1(x, r) \cap \mathrm{Cyl}_2(x, r)$ onto ℓ_x.

▶ **Lemma 16.** *Fix a point $x \in \mathbb{R}^d$ and let $C(x, r)$ be the capsule of radius r induced at x by a set S of n line segments. Then,*

$$V^-(x, r) \subseteq \mathcal{C}(x, r) \subseteq V^+(x, r), \quad \text{and} \quad \frac{\mathrm{vol}(V^+(x, r))}{\mathrm{vol}(V^-(x, r))} \leq 2(d+1).$$

Proof. The containment follows by the construction of the bounding volumes. In bounding the ratio of the two volumes, we use the same notation and assumptions as above, without loss of generality. Letting θ denote the acute angle between the two lines supporting s_1 and s_2, and $\mathbb{V}_{d-1}$ denote the volume of a unit ball in $\mathbb{R}^{d-1}$, we have

$$\frac{\mathrm{vol}(V^+(x, r))}{\mathrm{vol}(V^-(x, r))} \leq \frac{4\mathbb{V}_{d-1} r^{d-1} t_1(x) \cdot \csc(\theta)}{\frac{2}{d+1}\mathbb{V}_{d-1} r^{d-1} t_1(x) \cdot \csc(\theta)} \leq 2 \cdot (d+1),$$

where the numerator is the volume of the intersection of two cylinders, and the denominator is the volume of a cone in $\mathbb{R}^d$. ◀

In order to bound the number of leaf-level capsules within a ball of radius $O\left(\frac{1}{\varepsilon} \cdot \mathrm{diam}(S)\right)$, we use the following charging scheme. Again, we use the simplified notation and assumptions from before. A capsule $\mathcal{C}(x, r)$ will be charged to the two line segments s_1 and s_2. For a fixed pair of segments s_i and s_j, acting as respectively as s_1 and s_2 for the point x in consideration, we may restrict attention to all center points x lying in the cylinder of radius r with the line supporting s_i as axis; denote this cylinder by $\mathrm{Cyl}(s_i, r)$.

▶ **Lemma 17.** *The number of leaf level capsules charged to any pair of segments is* $O(\log \frac{\Delta}{\varepsilon})$.

Proof. We cover the points in $\mathrm{Cyl}(s_i, r)$ using a sequence of growing cylindrical intervals based on their distances to s_j. Define $\mathcal{I}_k(s_i, r)$ to be the set of points in $\mathrm{Cyl}(s_i, r)$ such that for all $x \in \mathcal{I}_k(s_i, r)$ we have that $2^{k-1}r \leq \mathrm{dist}(x, s_j) \leq 2^k r$. In other words, $\mathcal{I}_k(s_i, r)$ is the intersection of $\mathrm{Cyl}(s_i, r)$ with the *cylindrical shell* $\mathrm{Cyl}(s_j, 2^k r) \setminus \mathrm{Cyl}(s_j, 2^{k-1}r)$. It is easy to see that the volume of the intersection is maximized when s_i intersects s_j due to the symmetry of the cylinder $\mathrm{Cyl}(s_i, r)$ about the line supporting s_i. Similar to the upper bound on the volume of $\mathcal{C}(x, r)$ by that of $V^+(x, r)$, we see that $\mathrm{vol}(\mathcal{I}_k(s_i, r))$ is at most $4\mathbb{V}_{d-1}2^k r^d \cdot \csc(\theta)$, where θ is the acute angle between the two lines supporting s_i and s_j. For any capsule charged to s_i and s_j with center $x \in \mathcal{I}_k(s_i, r)$, we use the inner volume $V^-(x, r) \subseteq \mathcal{C}(x, r)$ to obtain a lower bound $\mathrm{vol}(C^\lambda(x, r)) \geq \frac{1}{d+1}\mathbb{V}_{d-1}2^k r^d \lambda^d \cdot \csc(\theta)$. By choosing the capsule centers to have global packing-covering properties as a Delone set, it follows that there can be at most $4(d+1)/\lambda^d = O(1)$ capsules centered within $\mathcal{I}_k(s_i, r)$.

The desired bound follows by repeating the above argument over all scales r, a total of $\log(\Delta)$, and all k with $2^k r \leq \frac{1}{\varepsilon} \cdot \mathrm{diam}(S)$, a total of $\log(1/\varepsilon)$. ◀

6 Conclusions and Future Work

We have presented a new AVD-based approach to answering ε-segment ANN queries based on a hierarchical covering of space by ellipsoids. By elaborating on the intrinsic geometry underlying more general distance functions, our work helps pave the way to extend well-established techniques from data structure design and approximation algorithms in Euclidean and metric spaces to more general geometries. Specifically, we anticipate further progress in understanding the metric-like structure defined by the local tensors derived from the Hessians of distance functions. This should directly benefit the development of more efficient space covers, e.g., circumventing dependence on the spread and other geometric parameters. In addition, we expect polytope approximation techniques to enable better ε-dependencies in the storage requirements. Put together, we leave it to future work to achieve both remaining tasks of eliminating dependence on the spread and improving low-level query processing to obtain $O(\log n/\varepsilon)$ query times with only $O(n^2/\varepsilon^{d/2})$ storage, by analogy with the best known corresponding results for ANN against point sets under the Euclidean metric modulo the quadratic dependence on n needed for storage in the case of segment-ANN.

References

1. A. Abdelkader and D. M. Mount. Economical Delone sets for approximating convex bodies. In *Proc. 16th Scand. Workshop Algorithm Theory*, pages 4:1–4:12, 2018.
2. P. K. Agarwal, N. Rubin, and M. Sharir. Approximate nearest neighbor search amid higher-dimensional flats. In *Proc. 25th Annu. European Sympos. Algorithms*, volume 87, pages 4:1–4:13, 2017.
3. O. Aichholzer, D. Z. Chen, D. T. Lee, A. Mukhopadhyay, E. Papadopoulou, and F. Aurenhammer. Voronoi diagrams for direction-sensitive distances. In *Proc. 13th Annu. Sympos. Comput. Geom.*, pages 418–420, 1997.
4. A. Andoni, P. Indyk, R. Krauthgamer, and H. L. Nguyên. Approximate line nearest neighbor in high dimensions. In *Proc. 20th Annu. ACM-SIAM Sympos. Discrete Algorithms*, pages 293–301, 2009.
5. A. Andoni, A. Naor, A. Nikolov, I. Razenshteyn, and E. Waingarten. Hölder homeomorphisms and approximate nearest neighbors. In *Proc. 59th Annu. IEEE Sympos. Found. Comput. Sci.*, pages 159–169, 2018.

6 Boris Aronov. A lower bound on Voronoi diagram complexity. *Inform. Process. Lett.*, 83(4):183–185, 2002.

7 S. Arya, G. D. da Fonseca, and D. M. Mount. Near-optimal ε-kernel construction and related problems. In *Proc. 33rd Internat. Sympos. Comput. Geom.*, pages 10:1–15, 2017. arXiv:1604.01175.

8 S. Arya, G. D. da Fonseca, and D. M. Mount. On the combinatorial complexity of approximating polytopes. *Discrete Comput. Geom.*, 58(4):849–870, 2017. doi:10.1007/s00454-016-9856-5.

9 S. Arya, G. D. da Fonseca, and D. M. Mount. Optimal approximate polytope membership. In *Proc. 28th Annu. ACM-SIAM Sympos. Discrete Algorithms*, pages 270–288, 2017.

10 S. Arya, G. D. da Fonseca, and D. M. Mount. Approximate convex intersection detection with applications to width and Minkowski sums. In *Proc. 26th Annu. European Sympos. Algorithms*, pages 3:1–14, 2018. doi:10.4230/LIPIcs.ESA.2018.3.

11 S. Arya, G. D. da Fonseca, and D. M. Mount. Approximate polytope membership queries. *SIAM J. Comput.*, 47(1):1–51, 2018. doi:10.1137/16M1061096.

12 S. Arya and T. Malamatos. Linear-size approximate Voronoi diagrams. In *Proc. 13th Annu. ACM-SIAM Sympos. Discrete Algorithms*, pages 147–155, 2002.

13 S. Arya, T. Malamatos, and D. M. Mount. Space-time tradeoffs for approximate nearest neighbor searching. *J. Assoc. Comput. Mach.*, 57:1–54, 2009. doi:10.1145/1613676.1613677.

14 F. Aurenhammer, B. Jüttler, and G. Paulini. Voronoi Diagrams for Parallel Halflines and Line Segments in Space. In *Proc. 28th Annu. Internat. Sympos. Algorithms Comput.*, volume 92, pages 7:1–7:10, 2017.

15 I. Bárány. The technique of M-regions and cap-coverings: A survey. *Rend. Circ. Mat. Palermo*, 65:21–38, 2000.

16 G. Barequet, E. Papadopoulou, and M. Suderland. Unbounded Regions of High-Order Voronoi Diagrams of Lines and Segments in Higher Dimensions. In *Proc. 30th Annu. Internat. Sympos. Algorithms Comput.*, volume 149, pages 62:1–62:15, 2019.

17 R. Basri, T. Hassner, and L. Zelnik-Manor. Approximate nearest subspace search. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33:266–278, 2011.

18 J.-D. Boissonnat, M. Sharir, B. Tagansky, and M. Yvinec. Voronoi diagrams in higher dimensions under certain polyhedral distance functions. *Discrete Comput. Geom.*, 19(4):485–519, April 1998.

19 B. Chazelle, D. Liu, and A. Magen. Approximate range searching in higher dimension. *Comput. Geom. Theory Appl.*, 39:24–29, 2008.

20 L.P. Chew, K. Kedem, M. Sharir, B. Tagansky, and E. Welzl. Voronoi diagrams of lines in 3-space under polyhedral convex distance functions. *J. Algorithms*, 29(2):238–255, 1998.

21 S. Cost and S. Salzberg. A weighted nearest neighbor algorithm for learning with symbolic features. *Machine Learning*, 10:57–78, 1993.

22 T. M. Cover and P. E. Hart. Nearest neighbor pattern classification. *IEEE Trans. Image Proc.*, 13:57–67, 1967.

23 S. Deerwester, S. T. Dumals, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *J. Amer. Soc. Inform. Sci.*, 41:391–407, 1990.

24 L. Devroye and T. J. Wagner. Nearest neighbor methods in discrimination. In P. R. Krishnaiah and L. N. Kanal, editors, *Handbook of Statistics*, volume 2. North-Holland, 1982.

25 R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis.* John Wiley & Sons, NY, 1973.

26 I. Z. Emiris, T. Malamatos, and E. Tsigaridas. Approximate nearest neighbor queries among parallel segments. In *26th European Workshop on Computational Geometry (EuroCG 2010)*, 2010.

27 H. Everett, D. Lazard, S. Lazard, and M. Safey El Din. The Voronoi diagram of three lines. *Discrete Comput. Geom.*, 42(1):94–130, July 2009.

28 U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. *Advances in Knowledge Discovery and Data Mining.* AAAI Press/Mit Press, 1996.

29 M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: The QBIC system. *IEEE Computer*, 28:23–32, 1995.

30 A. Gersho and R. M. Gray. *Vector Quantization and Signal Compression*. Kluwer Academic, Boston, MA, 1991.

31 S. Har-Peled. A replacement for Voronoi diagrams of near linear size. In *Proc. 42nd Annu. IEEE Sympos. Found. Comput. Sci.*, pages 94–103, 2001.

32 S. Har-Peled, P. Indyk, and R. Motwani. Approximate nearest neighbor: Towards removing the curse of dimensionality. *Theo. of Comput.*, 8:321–350, 2012.

33 S. Har-Peled and M. Mendel. Fast construction of nets in low dimensional metrics, and their applications. *SIAM J. Comput.*, 35:1148–1184, 2006.

34 P. Indyk. Nearest neighbors in high-dimensional spaces. In J. E. Goodman and J. O'Rourke, editors, *The Handbook of Discrete and Computational Geometry, 2nd Edition*, pages 877–892. Chapman & Hall/CRC, Boca Raton, FL, 2004.

35 F. John. Extremum problems with inequalities as subsidiary conditions. In *Studies and Essays Presented to R. Courant on his 60th Birthday*, pages 187–204. Interscience Publishers, Inc., New York, 1948.

36 M. I. Karavelas. A robust and efficient implementation for the segment Voronoi diagram. In *Internat. Symp. on Voronoi Diagrams in Sci. and Engin.*, volume 2004, pages 51–62, 2004.

37 V. Koltun and M. Sharir. 3-dimensional Euclidean Voronoi diagrams of lines with a fixed number of orientations. *SIAM J. Comput.*, 32(3):616–642, 2003.

38 V. Koltun and M. Sharir. Polyhedral Voronoi diagrams of polyhedra in three dimensions. *Discrete Comput. Geom.*, 31(1):83–124, January 2004.

39 R. Krauthgamer and J. R. Lee. Navigating nets: Simple algorithms for proximity search. In *Proc. 15th Annu. ACM-SIAM Sympos. Discrete Algorithms*, pages 798–807, 2004.

40 R. Krauthgamer and J. R. Lee. The black-box complexity of nearest-neighbor search. *Theo. Comp. Sci.*, 348:262–276, 2005.

41 A. Laddha, Y. T. Lee, and S. Vempala. Strong self-concordance and sampling. In *Proc. 52nd Annu. ACM Sympos. Theory Comput.*, pages 1212–1222, 2020.

42 A. M. Macbeath. A compactness theorem for affine equivalence-classes of convex regions. *Canad. J. Math*, 3:54–61, 1950.

43 S. Mahabadi. Approximate nearest line search in high dimensions. In *Proc. 26th Annu. ACM-SIAM Sympos. Discrete Algorithms*, pages 337–354, 2015.

44 W. Mulzer, H. L. Nguyên, P. Seiferth, and Y. Stein. Approximate k-flat nearest neighbor search. In *Proc. 47th Annu. ACM Sympos. Theory Comput.*, pages 783–792, 2015.

45 H. Narayanan. Randomized interior point methods for sampling and optimization. *Ann. Appl. Probab.*, 26(1):597–641, 2016.

46 Y. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Publishing Company, Incorporated, 2014.

47 Y. Nesterov and M. Todd. On the Riemannian geometry defined by self-concordant barriers and interior-point methods. *Found. Comput. Math.*, 2(4):333–361, 2002.

48 S. Sachdeva and N. K. Vishnoi. The mixing time of the Dikin walk in a polytope—A simple proof. *Oper. Res. Lett.*, 44(5):630–634, 2016.

49 K. Sugihara and M. Iri. A robust topology-oriented incremental algorithm for Voronoi diagrams. *Internat. J. Comput. Geom. Appl.*, 04:179–228, 1994.

50 S. Vijayanarasimhan, P. Jain, and K. Grauman. Hashing hyperplane queries to near points with applications to large-scale active learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36:276–288, 2014.

51 X. Wang, S. Atev, J. Wright, and G. Lerman. Fast subspace search via Grassmannian based hashing. In *Proc. IEEE Internat. Conf. Comput. Vision (ICCV)*, December 2013.

52 S. J. Wright. *Primal-Dual Interior-Point Methods*. Society for Industrial and Applied Mathematics, 1997.