Stability analysis of kinetic orientation-based shape descriptors

Stability analysis of kinetic orientation-based shape descriptors

Wouter Meulemans[1], Kevin Verbeek[1], and Jules Wulms[1]

1 **Dept. of Mathematics and Computer Science, TU Eindhoven, The Netherlands**
[w.meulemans|k.a.b.verbeek|j.j.h.m.wulms]@tue.nl

Abstract

We study three *orientation-based* shape descriptors on a set of continuously moving points P: the first principal component, the smallest oriented bounding box and the thinnest strip. Each of these shape descriptors essentially defines a cost capturing the quality of the descriptor (sum of squared distances for principal component, area for oriented bounding box, and width for strip), and uses the orientation that minimizes the cost. This optimal orientation may be very unstable as the points are moving, which is undesirable in many practical scenarios. Alternatively, we can bound the speed with which the orientation of the descriptor may change. However, this can increase the cost (and hence lower the quality) of the resulting shape descriptor. In this paper we study the trade-off between stability and quality of these shape descriptors. We first show that there is no *stateless algorithm*, which depends only on the input points in one time step and not on previous states, that both approximates the minimum cost of a shape descriptor and achieves bounded speed. On the other hand, if we can use the previous state of the shape descriptor to compute the new state, then we can define "chasing" algorithms that attempt to follow the optimal orientation with bounded speed. Under mild conditions, we show that chasing algorithms with sufficient bounded speed approximate the optimal cost at every time step for oriented bounding boxes and strips, but not for principal components. The analysis of such chasing algorithms is challenging and has received little attention in literature, hence we believe that our methods used to perform this analysis are of independent interest.

1998 ACM Subject Classification Theory of computation → Computational geometry

Keywords and phrases Stability analysis, Time-varying data, Shape descriptors

1 Introduction

Shape descriptors are simplified representations of (complex) shapes. The lower complexity of shape descriptors typically allows for more efficient computations on the underlying shapes, provided that the shape descriptors capture the relevant properties of the shapes. As such, they are often used to efficiently compute shape classification, shape similarity, shape retrieval, or simply to provide an effective summary of complex data. Shape descriptors thus play an important role in fields that rely on shape analysis, like computer vision (shape recognition) [6, 8, 36], computer graphics (bounding boxes for efficient broad-phase collision detection) [2, 17, 22, 31], medical imaging (diagnosis or surgical planning) [10, 18, 21, 37], and machine learning (shape classification) [28, 33, 34, 35]. Many shape descriptors exist, capturing various aspects of shape. In this paper we focus on shape descriptors that capture the overall orientation of the underlying shape. We assume that the underlying shape is represented by a point cloud. Specifically, we are interested in the following shape descriptors on point sets:

Principal component (PC): The first principal component vector of the point set;

Oriented bounding box (OBB): The smallest (in area) oriented bounding box that contains all points. We are interested in the orientation of the major axis of the bounding box;

Covering strip (STRIP): The thinnest strip containing all points. Here we are interested in the orientation of the lines bounding the strip.

The above shape descriptors essentially describe optimization problems, where each descriptor minimizes a particular optimization function over all orientations. This optimization function is the sum of squared distances (to a line with the given orientation) for PC, the area of the bounding box with a given orientation for OBB, and the width of the strip with a given orientation for STRIP. These optimization functions capture the *quality* of the shape descriptor, that is, how well the descriptor represents the underlying shape. With this interpretation of shape descriptors we can also consider shape descriptors of suboptimal quality (e.g., bounding boxes with a slightly larger area than the smallest one). This allows us to make a trade-off between quality and other desirable properties of shape descriptors. One such property is *stability*: small changes in the underlying shape should lead to small changes in the descriptor. This property is useful for effective shape classification, but becomes especially important when analyzing or visualizing shapes that change over time. For example, we can effectively visualize how a large set of moving points evolves by drawing a single glyph capturing orientation (e.g., an arrow or line segment) for a small number of subsets of points, which gives a clear and comprehensible overview of the data. Or we can plot a set of high-dimensional moving points in 2 dimensions by projecting it on the first two principal components for each time step. It should be clear that, in these scenarios, a sudden change of the orientation of the shape descriptor renders the visualization completely ineffective, as it will be hard or impossible to follow the temporal patterns. Unfortunately, all shape descriptors with optimal quality mentioned above are unstable and may have discrete "flips" in their orientation, even for continuously moving point sets (see Section 2 for details). It is therefore of interest to sacrifice some quality for stability for these shape descriptors.

Problem description. The main goal of this paper is to formally analyze the trade-off between quality and stability for the three shape descriptors PC, OBB, and STRIP. Our input consists of a set of n moving points, where each point can move with at most unit speed. The output, representing the shape descriptor, is an orientation for each point in time. To perform the stability analysis, we apply the recently introduced stability analysis framework by Meulemans et al. [24]. This framework distinguishes between topological stability and Lipschitz stability (formal definitions are given in Section 2). An algorithm is *topologically stable* if its output changes continuously over time. An algorithm is *K-Lipschitz stable* if its output changes with speed at most K (assuming the input points move with at most unit speed). Although topological stability only partially covers aspects of stability (since the output may still move with unbounded speed), it is typically much easier to analyze and allows for much tighter bounds. In this paper we study both the topological stability and the Lipschitz stability of the orientation-based shape descriptors PC, OBB, and STRIP.

Algorithms for kinetic (moving) input can adhere to different models, which may influence the results of the stability analysis. Let $\mathcal{A}$ be an algorithm that maps input to output, and let $I(t)$ be the input that depends on time t. We can distinguish the following models:

Stateless algorithms: The output depends only on the input $I(t)$ at a particular point in time, and no other information of earlier time steps. This in particular means that if $I(t_1) = I(t_2)$, then $\mathcal{A}$ produces the same output at time t_1 and at time t_2;

State-aware algorithms: The algorithm $\mathcal{A}$ also maintains a state S (typically the output at the previous time step) over time. Thus, even if $I(t_1) = I(t_2)$, then $\mathcal{A}$ may produce different results at time t_1 and t_2 if the states at those times were different;

Clairvoyant algorithms: The algorithm $\mathcal{A}$ has access to the complete function $I(t)$ and can adapt to future inputs. Thus, the complete output over time can be computed offline.

In this paper we consider only stateless and state-aware algorithms. A stateless algorithm can be stable only if it defines a mapping from input to output that is naturally continuous. On the other hand, a state-aware algorithm can easily enforce continuity by keeping the output at the previous time step as the state. We can then define a *chasing algorithm*, which always moves the output of the previous time step with maximum allowed speed towards the optimal solution of the current time step. The challenge here lies in bounding ratio between the quality of the solutions used by the algorithm and the (unstable) optimal quality.

Related work. Shape descriptors are a wide topic, studied in various subfields of computer science. A full exposition of literature is out of scope. We focus on results in computational geometry or related to PC, OBB or STRIP, and results on kinetic data or stability.

Not surprisingly, the oriented shape descriptors are related, yet slightly different. The bounding boxes that align the PC direction have been investigated by Dimitrov et al. [13]: they show that the ratio between the volume of the "principal-component bounding box" and the minimal-volume bounding box is unbounded. The first principal component minimizes the sum of squared distances to a line. Other fitness measures have been considered, such as the sum of distances or vertical distances [11]. Computing OBB for static point sets is a classic problem in computational geometry. In two dimensions, one side of the optimal box aligns with a side of the convex hull and it can be computed in linear time after finding the convex hull [16, 29]; a similar property holds in three dimensions, allowing a cubic-time algorithm [25]. The relevance of bounding boxes in 3D as a component of other algorithms also led to efficient approximation algorithms, such as a $(1+\epsilon)$-approximation algorithm in $O(n+1/\epsilon^{4.5})$ time or an easier algorithm with running time $O(n \log n + n/\epsilon^3)$ [3]. Bounding boxes find applications in tree structures for spatial indexing [5, 20, 26, 27] and in speeding up collision detection and ray tracing techniques [2, 17, 31]. The optimal STRIP can be computed using the same techniques as OBB in two dimensions [16, 29]. Agarwal et al. [1] provide an $O(n+1/\epsilon^{O(1)})$-time approximation algorithm via ϵ-kernels for various measures of a point set, including STRIP and OBB.

Kinetic data structures, introduced by Basch et al. in 1999 [4], provide an analysis framework for maintaining optimal solutions to a computational-geometry problems for moving objects. It has been studied for a variety of problems, such as convex hull [4, 19], diameter, STRIP and OBB [6]. The before-mentioned approximation algorithms by Agarwal et al. [1] can also be generalized to this kinetic setting. Primary difference to our results is that with kinetic data structures, emphasis lies on maintaining (an approximation of) the optimal solution with efficient updates as the data changes. Whereas with stability, we focus on the need for continuity and analyze the ratio between optimal and continuous solutions.

The stability of kinetic 1-center and 1-median problems has been studied by Bespamyatnikh et al. [7]. They developed approximations by fixing the speed at which the center/median point moves and provide some results on the trade-off between solution quality and speed. Durocher and Kirkpatrick studied the stability of the kinetic 2-center problem, that is, covering a set of moving points with two disks. Their approach allows a trade-off between solution quality and the speed at which solution changes [14, 15]. De Berg et al. [12] show similar results in the black-box kinetic-data-structures model. Letscher et al. studied the stability of the medial axis for union of disks [23]. They show that the medial axis changes continuously under certain conditions or if the medial axis is pruned appropriately. Meulemans et al. introduced a framework for analyzing stability [24] and illustrate this framework by applying

it to maintaining the Euclidean minimum spanning tree on a set of moving points. They show a bounded topological stability for various topology definitions on the space of spanning trees, and that the Lipschitz stability is at most linear, but also at least linear if the allowed speed for the changes in the tree is too low. Van der Hoog et al. study the topological stability of the k-center problem [32], showing upper and lower bounds on the stability ratio for various measures: minimizing the maximum radius or the sum of radii, and covering using disks or squares. They show that for three of these cases, the topological stability is tightly bounded by 2; only for the min-max case with disks, the topological stability is lower than 2 for finite k. They also provide a clairvoyant algorithm to determine the best ratio attainable for a given set of moving points.

Results and organization. In Section 2 we introduce the three shape descriptors PC, OBB, and STRIP in detail, and we give formal definitions for topological and Lipschitz stability analysis. In Section 3 we prove that there exists no stateless algorithm for any of the shape descriptors that is both topologically stable and achieves a bounded approximation ratio for the quality of the optimal shape descriptor. We then consider state-aware algorithms and analyze the topological stability for each of the shape descriptors in Section 4. In Section 5 we analyze the Lipschitz stability of chasing algorithms for all three shape descriptors. We show that, under mild assumptions, chasing algorithms with sufficient speed can achieve a constant approximation ratio for OBB and STRIP. To the best of our knowledge, this is the first time a chasing algorithm has been analyzed. We believe that the new methods that we developed for this challenging stability analysis are of independent interest, and may be applied to the analysis of other chasing algorithms. We conclude the paper in Section 7.

2 Preliminaries

Our input consists of a set of n moving points $P = P(t) = \{p_1(t), \ldots, p_n(t)\}$ in 2 dimensions, where each $p_i(t)$ is a function $p_i\colon [0,T] \to \mathbb{R}^2$. We assume that, at each time t, not all points are at the same position. We further assume that each point moves with at most unit speed, that is, $\|p_i'(t)\| \leq 1$ for all times t. Beyond that, there are no further restrictions on the motion of points. Note that, since we study either stateless or state-aware algorithms, the algorithms will not have access to the full function $P(t)$, but only to the point set at a particular time t (and a state for state-aware algorithms). The output consists of an orientation $\alpha(t)$ for every time step $t \in [0,T]$. Technically this orientation is an element of the real projective line $\mathbb{RP}^1$, but we simply represent $\alpha(t)$ by a unit vector in $\mathbb{R}^2$ and implicitly identify opposite vectors, which is equivalent. Furthermore, although practically not possible (since algorithms can be executed only finitely often), we assume that the output $\alpha(t)$ is computed for all real values $t \in [0,T]$. This assumption significantly simplifies our analysis and does not strongly affect the results.

The shape descriptors we study typically compute more than just an orientation. However, it is easy to see that the stability of the shape descriptors is most strongly affected by the optimal orientation. We therefore ignore all other aspects of the shape descriptor to analyze the stability, and simply assume that these aspects are chosen optimally for the given orientation without any cost with regard to the stability. Below we discuss each shape descriptor in more detail and formally state the corresponding optimization functions.

The first *principal component* (PC) is the vector along which the point set has the most variance (see Figure 1 left). It may change with unbounded speed as the points are moving (see Figure 2 left). It can be computed by centering the point set at the mean of

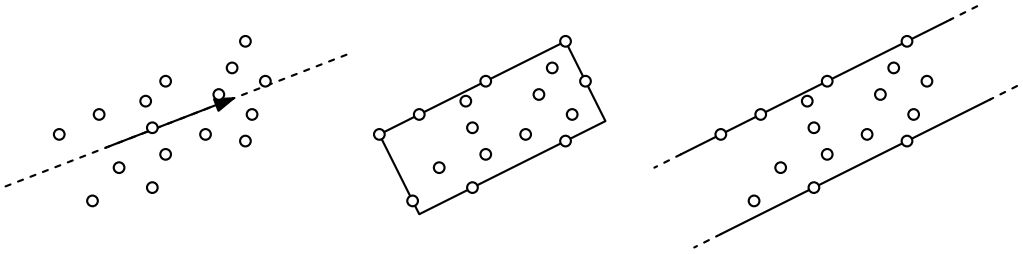

Figure 1 Oriented shape descriptors PC, OBB and STRIP on the same point set.

the coordinates, computing the covariance matrix of the resulting point coordinates, and extracting the eigenvector of this matrix with the largest eigenvalue. Alternatively, the first principal component is the orientation of the line that minimizes the sum of squared distances between the points and the line. Let $\mathcal{L}(\alpha)$ be the set of lines with orientation α, and let $d(L,p)$ be the distance between a line L and a point p. We define the following corresponding quality function for an orientation α, which is minimized by PC.

$$f_{\text{PC}}(\alpha, P) = \min_{L \in \mathcal{L}(\alpha)} \sum_{p \in P} d(L,p)^2 \tag{1}$$

The smallest *oriented bounding box* (OBB) minimizes the area over all (oriented) rectangles covering all points (see Figure 1 middle). It may flip orientation as the points are moving (see Figure 2 middle). Let the *extent* of P along a unit vector α be $w_\alpha(P) = \max_{p,q \in P}(p-q)\cdot\alpha$, and let $\alpha^\perp$ be the vector (orientation) orthogonal to α. We define the following quality function for an orientation α, which is minimized by OBB.

$$f_{\text{OBB}}(\alpha, P) = w_\alpha(P) w_{\alpha^\perp}(P) \tag{2}$$

The *covering strip* (STRIP) minimizes the width of an (infinite) strip covering all points (see Figure 1 right). It may flip orientation as the points are moving (see Figure 2 right). It can be computed in a similar way as the smallest oriented bounding box. We define the following corresponding quality function for an orientation α, which is minimized by STRIP.

$$f_{\text{STRIP}}(\alpha, P) = w_{\alpha^\perp}(P) \tag{3}$$

Topological and Lipschitz stability. To analyze the stability of the shape descriptors we use the framework introduced by Meulemans et al. [24]. We slightly change their definitions by restricting it to our setting. Let $\mathcal{A}$ be an algorithm that takes a point set as input and computes an orientation (where the potential state for state-aware algorithms is implicit). The definition of the topological stability ratio depends on the topology of the input space and the topology of the output space. This is straightforward: the input space is $\mathbb{R}^{2n}$ and the output space is topologically equivalent to S^1, and we can use those topologies in the analysis. We can then define the topological stability ratio of a shape descriptor Π as follows.

$$\rho_{TS}(\Pi) = \inf_{\mathcal{A}} \sup_{P(t)} \max_{t \in [0,T]} \frac{f_\Pi(\mathcal{A}(P(t)), P(t))}{\min_\gamma f_\Pi(\gamma, P(t))} \tag{4}$$

where the infimum is taken over all algorithms $\mathcal{A}$ for which $\mathcal{A}(P(t))$ is continuous. The K-Lipschitz stability ratio is defined almost exactly the same:

$$\rho_{LS}(\Pi, K) = \inf_{\mathcal{A}} \sup_{P(t)} \max_{t \in [0,T]} \frac{f_\Pi(\mathcal{A}(P(t)), P(t))}{\min_\gamma f_\Pi(\gamma, P(t))} \tag{5}$$

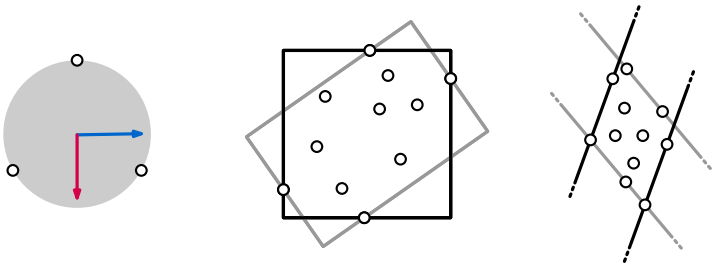

Figure 2 A flip in orientation for PC, OBB and STRIP. Small changes in the positions of the points make one of the orientations optimal over the other.

where instead the infimum is taken over all algorithms $\mathcal{A}$ for which $\mathcal{A}(P(t))$ is K-Lipschitz. Here we assume that the output is measured in radians. Thus, the orientation or angle can change with a speed of at most K radians per time unit.

3 Stateless algorithms

Here we prove that stateless algorithms cannot achieve bounded topological stability ratio for any shape descriptor. This readily implies an unbounded K-Lipschitz stability ratio for any K. The argument is entirely topological. A stateless topologically stable algorithm (for which the output behaves continuously) is a continuous map from the input space to the output space. If all points of set P lie on a single line with orientation α, then $f_\Pi(\gamma, P) = 0$ if and only if $\gamma = \alpha$, for all considered shape descriptors.

▶ **Theorem 1.** *For stateless algorithms* $\rho_{TS}(\text{OBB}) = \rho_{TS}(\text{PC}) = \rho_{TS}(\text{STRIP}) = \infty$ *if the point set contains at least three points.*

Proof. The idea is to construct a continuous map $g\colon D^2 \to \mathbb{R}^{2n}$ on the two-dimensional closed disk, such that the image of g consists of valid point sets (not all points at the same position), and that the image of ∂D^2 under g forces the orientation of the shape descriptor. We parameterize D^2 using polar coordinates (r, ϕ) for $0 \leq \phi < 2\pi$ and $0 \leq r \leq 1$. We first construct a map g' as follows:

$$g'(r, \phi) = \{(\frac{ri}{n} \sin \phi, \frac{ri}{n} \cos \phi) \mid 1 \leq i \leq n\}$$

Since g' always places all points on a line, the orientation of the shape descriptor is always forced (otherwise the approximation ratio is ∞). However, $g'(0, \phi)$ is not a valid point set, since it places all points at the origin. Now let P^* be a suitably chosen point set. We define $g(r, \phi) = g'(r, \phi) + (1 - r)P^*$. If the number of points is at least three, then we can easily find a point set P^* such that $g(r, \phi)$ is always a valid point set (in fact, it suffices if not all points of P^* lie on a single line). Furthermore, the orientation of the shape descriptors is still fixed for point sets $g(1, \phi)$, namely $\alpha = \phi \pmod{\pi}$. As a result, any stateless algorithm with an approximation ratio $\rho < \infty$ defines, along with g, a continuous mapping h from D^2 to S^1 where ∂D^2 is mapped to a double cover of S^1 ($\mathbb{RP}^1$ is topologically equivalent to S^1). We claim that such a mapping from D^2 to S^1 cannot exist.

For the sake of contradiction, assume that such a map h exists. Now let $j : S^1 \to D^2$ be a continuous function such that every point $x \in S^1$ is mapped to the boundary $\partial D^2 \subseteq D^2$ with a slight rotation. Brouwer's Fixed Point Theorem [9] tells us that every continuous map f from a compact convex set to itself has a fixed point x_0, such that $f(x_0) = x_0$. Hence,

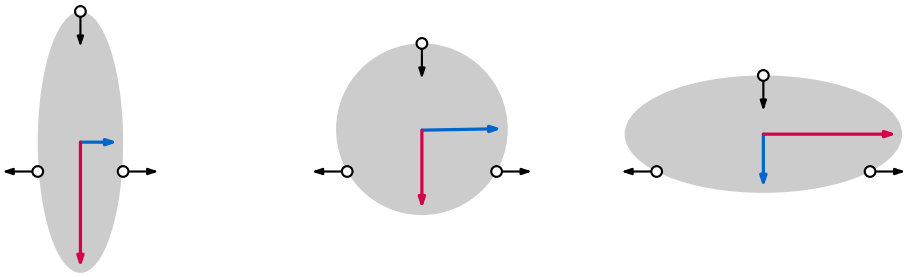

Figure 3 Moving points causing a flip between the first principal component (red) and second principal component (blue).

$j \circ h : D^2 \to D^2$ should have a fixed point. However, since j maps all points with a slight rotation, no point of ∂D^2 can be a fixed point. All points in the interior of D^2 are mapped to the boundary and are not fixed either. This contradiction implies that h cannot exist. Thus the topological stability ratio is ∞ for stateless algorithms. ◀

4 Topological stability

In this section we turn to state-aware algorithms, and we analyze the topological stability of the shape descriptors. Specifically, we prove the following results.

▶ **Theorem 2.** *The topological stability ratios of shape descriptors* PC, OBB *and* STRIP *are:*

- $\rho_{\text{TS}}(\text{PC}) = 1$
- $\rho_{\text{TS}}(\text{OBB}) = \frac{1}{2} + \frac{1}{2}\sqrt{2}$
- $\rho_{\text{TS}}(\text{STRIP}) = \sqrt{2}$

4.1 First principal component

The first principal component can flip orientation in a single time step as the points are moving (see Figure 3). Interestingly though, we can always maintain an orientation α continuously such that f_{PC} is minimized at every time step.

▶ **Lemma 3.** $\rho_{\text{TS}}(\text{PC}) = 1$

Proof. Consider a time t where the first principal component flips between two orientations, represented by unit vectors $\vec{v}_1$ and $\vec{v}_2$. As described in Section 2, the first principal component is the eigenvector of the covariance matrix C with the largest eigenvalue. Since eigenvalues change continuously if the data changes continuously [30, Theorem 3.9.1], both $\vec{v}_1$ and $\vec{v}_2$ must have some eigenvalue λ^* at time t. But that means that every interpolated vector $\vec{v} = (1-u)\vec{v}_1 + u\vec{v}_2$ also has eigenvalue λ^*, since $C\vec{v} = (1-u)C\vec{v}_1 + uC\vec{v}_2 = (1-u)\lambda^*\vec{v}_1 + u\lambda^*\vec{v}_2 = \lambda^*\vec{v}$. As a result, $f_{\text{PC}}(\vec{v}) = f_{\text{PC}}(\vec{v_1})$, and we can continuously change orientation from $\vec{v}_1$ to $\vec{v}_2$ without decreasing the quality of the shape descriptor. ◀

4.2 Oriented bounding box

We cannot always continuously maintain an oriented bounding box of minimum area. Therefore, to prove a tight bound on the topological stability ratio, we also need to prove a lower bound. To prove an upper bound on the topological stability ratio, it is sufficient to consider a single point in time, since the output can change arbitrarily fast. On the other hand, to prove a lower bound on the topological stability ratio, we must construct a full time-varying

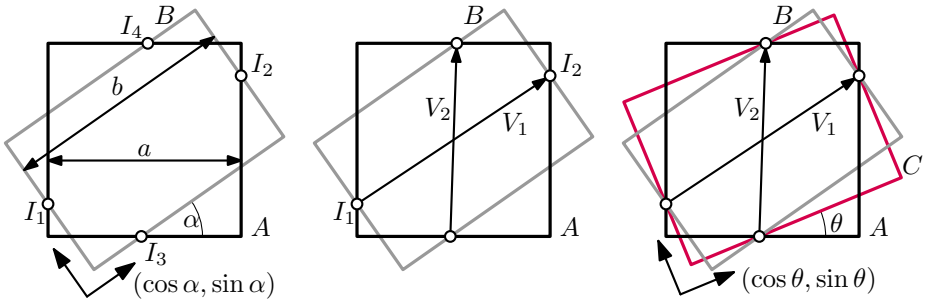

Figure 4 Construction of closed formula for area of intermediate solution C.

point set such that the corresponding approximation ratio must occur at some point in time during this motion. Note that a lower bound on the topological stability ratio is immediately also a lower bound on the K-Lipschitz stability ratio for any value of K.

▶ **Lemma 4.** $\rho_{\text{TS}}(\text{OBB}) \leq \frac{1}{2} + \frac{1}{2}\sqrt{2}$

Proof. Consider a time t at which two oriented bounding boxes A and B both have the smallest area 1. Let angle α denote the difference in orientation between A and B (chosen as small as possible, so $0 < \alpha \leq \pi/4$). Let a and b denote the length of the major axes of A and B, respectively. At time t we continuously change the orientation of the box between that of A and B while making sure that the box still contains all points (see Figure 4). The goal is to compute the maximal size of the intermediate box in the worst case.

The points P must all be contained inside the intersection of A and B, for otherwise A and B would not be bounding boxes. Furthermore, no side of A may be completely outside B or vice versa, for otherwise one of the boxes could be made smaller. Thus, all sides intersect, and we are interested in four of these intersections $I_1, \ldots, I_4$ (which depends on the direction of rotation, see Figure 4). In general, A and B do not share a center, so we assume that the center of A is at (dx, dy). Now consider an intermediate box C with angle $\theta \leq \alpha$ with respect to box A. Note that C contains the intersection of A and B as long as it contains $I_1, \ldots, I_4$. We can define $V_1 = I_2 - I_1$ and $V_2 = I_4 - I_3$ to obtain a formula for the area of C. Note that V_1 and V_2 do not depend on dx and dy, and thus we can eliminate those variables.

$$\mathcal{C}(a, b, \alpha, \theta) = V_1 \cdot (\cos\theta, \sin\theta) \times V_2 \cdot (-\sin\theta, \cos\theta) \tag{6}$$

We are now interested in the maximum of $\mathcal{C}$ under the given constraints. We first observe that $\frac{\partial \mathcal{C}}{\partial \theta} = 0$ if and only if $\theta = \alpha/2$ in the domain, which implies that we can set $\theta = \alpha/2$. Next, we assume w.l.o.g. that $b \geq a$ and $b = ca$ for some $c \geq 1$. Since b is at most the length of the diagonal of A, we get that $c \leq \sqrt{a^2 + (1/a)^2}/a \leq \sqrt{2}$ (since $a \geq 1$). The resulting formula is $\mathcal{C}(a, ca, \alpha, \alpha/2) = \frac{(1+c)^2}{2c(1+\cos\alpha)}$. It is easy to derive that this function is maximized when c and α are maximized. We thus set $\alpha = \pi/4$ and $c = \sqrt{2}$ to obtain that $\mathcal{C}(1, \sqrt{2}, \pi/4, \pi/8) \leq \frac{1}{2} + \frac{1}{2}\sqrt{2}$. ◀

▶ **Lemma 5.** $\rho_{\text{TS}}(\text{OBB}) \geq \frac{1}{2} + \frac{1}{2}\sqrt{2}$

Proof. Consider a point set P consisting of four points positioned in a square of area 1. The smallest oriented bounding box A is the square outlined by P. Any triple of points in P forms a triangle, for which the smallest OBB has the same area as A. Let p be the bottom-right point of p and let $P' = P \setminus \{p\}$. Furthermore, let B be the smallest OBB of P' with a different orientation than A. If we would continuously rotate box C from A to B

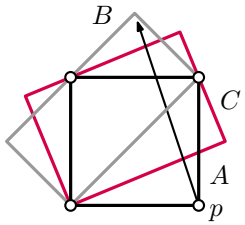

Figure 5 Moving points forcing a flip; a bounding box at least as big as the red box is required.

whilst containing the points in P', then C will have area $\frac{1}{2} + \frac{1}{2}\sqrt{2}$ when the angle with A is $\pi/8$, since this is the exact configuration used in the upper bound proof (see Figure 5).

Consider moving the point p from the right-bottom corner of A to the top corner of B. At any time during the motion, either A or B contains p, so the area of the smallest OBB is at most 1. Also, since P' does not move, whenever box C makes an angle of $\pi/8$ with A, its area is $\frac{1}{2} + \frac{1}{2}\sqrt{2}$. Finally, if C starts at an orientation more than $\pi/8$ away from A, then its starting area would be more than $\frac{1}{2} + \frac{1}{2}\sqrt{2}$, since C would have to extend to include p, and rotating C further away from A only makes the area larger. Similarly, if C ends at an orientation less than $\pi/8$ away from A, then the final area would be more than $\frac{1}{2} + \frac{1}{2}\sqrt{2}$. So at some point C must make an angle of $\pi/8$ with A, which concludes the proof. ◀

4.3 Covering strip

▶ **Lemma 6.** $\rho_{\text{TS}}(\text{STRIP}) \leq \sqrt{2}$

Proof. Consider a time t at which there are two thinnest strips A and B of width 1 with different orientations. All points must be contained in the diamond-shaped intersection D of A and B (see Figure 6). If we continuously rotate a strip C from A to B, then at some point the width of C must be at least the length of one of the diagonals of D. To maximize the length of the shortest diagonal, D must be a square with side length 1. Therefore, the width of C is at most $\sqrt{2}$ during the rotation from A to B. ◀

▶ **Lemma 7.** $\rho_{\text{TS}}(\text{STRIP}) \geq \sqrt{2}$

Proof. Let P consist of four points positioned in a unit square S. There are two thinnest strips A and B for P, each of which is oriented along a different pair of parallel edges of S (see Figure 7). If the orientation of a strip is $\pi/4$ away from A and B, then its width is $\sqrt{2}$.

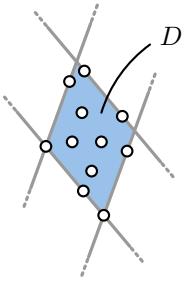

Figure 6 A configuration having two strips of minimal width and overlapping area D.

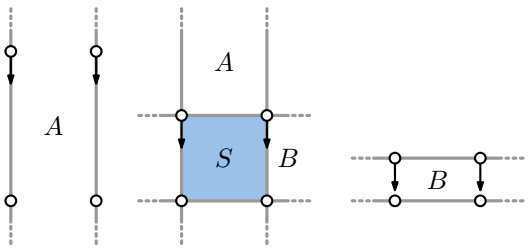

Figure 7 An instance of moving points where the thinnest strip changes orientations. The configuration that leads to the best intermediate solutions is shown in the middle.

Now assume that the top points of S are moving along the vertical sides of S, starting from high above. Clearly, at the start of this motion, any strip C approximating the thinnest strip must align with A. At the end of this motion, when the top points align with the bottom points of S, the strip C must align with B. Therefore, the strip C must at some point make an angle of $\pi/4$ with A and B. If x is the distance between the top and bottom points of S, then the width of C at this orientation is $(1+x)\sqrt{2}/2$. The minimal width is $\min(x, 1)$. It is easy to verify that this ratio is at least $\sqrt{2}$, which concludes the proof. ◀

5 Lipschitz stability

Here we analyze the Lipschitz stability of the three shape descriptors for state-aware algorithms. To derive meaningful bounds, the relation between distances/speeds in input and output space should be *scale-invariant* [24]. This is currently not the case: if we scale the coordinates of the points, then the distances in the input space change accordingly, but the distances in the output space (between orientations) do not. To remedy this problem, we require that diameter D of $P(t)$ is at least 1 for every time t. This assumption is sufficient to prove a bounded Lipschitz stability ratio for both OBB and STRIP. For PC this is not sufficient, as we argue at the end of this section.

To bound the Lipschitz stability ratio, we use a chasing algorithm as introduced in Section 1. However, instead of chasing the orientation of the optimal shape descriptor, we chase the orientation of a diametrical pair. Although chasing the optimal shape descriptor would be better in general, chasing a diametrical pair is easier to analyze and sufficient to obtain a bounded Lipschitz stability ratio for both OBB and STRIP simultaneously.

5.1 Chasing the diametrical pair

We denote the orientation of a diametrical pair as $\alpha = \alpha(t)$ and the diameter as $D = D(t) \geq 1$. Furthermore, let $W = W(t)$ be the width of the thinnest strip with orientation $\alpha(t)$ covering all points in $P(t)$, and let $z = z(t) = W(t)/D(t)$ be the *aspect ratio* of the *diametric box* with orientation $\alpha(t)$. We will generally omit the dependence on t if t is clear from the context. Finally, we have a chasing algorithm that has orientation $\beta = \beta(t)$ and can change orientation with at most a constant speed K.

Approach. The main goal is to keep β as close to α as possible, specifically within a sufficiently small interval around α. The challenge lies with the discrete flips of α. We must argue that, although flips can happen instantaneously, they cannot happen often within a short time-span – otherwise we can never keep β close to α with a bounded speed. Furthermore, the size of the interval must depend on the aspect ratio z, since if $z = 0$, the interval around α must have zero size as well to guarantee a bounded approximation ratio.

For the analysis we introduce three functions depending on z: $T(z)$, $H(z)$, and $J(z)$. Function $H(z)$ defines an interval $[\alpha - H(z), \alpha + H(z)]$ that we call the *safe zone*. We aim to show that, if β leaves the safe zone at some time t, it must return to the safe zone within the time interval $(t, t+T(z)]$. We also define a larger interval $I = [\alpha - H(z) - J(z), \alpha + H(z) + J(z)]$. We refer to the parts of I that are outside of the safe zone as the *danger zone* (see Figure 8). Although β may momentarily end up in the danger zone due to discontinuous changes, it must quickly find its way back to the safe zone. We aim to guarantee that β stays within I at all times. Let $E = E(t)$ refer to an endpoint of I (it does not matter which, due to symmetry). We call $J(z)$ the *jumping distance* and we require that $J(z)$ upper bounds how far E can "jump" in a single time step. Note that $J(z)$ is defined recursively through E,

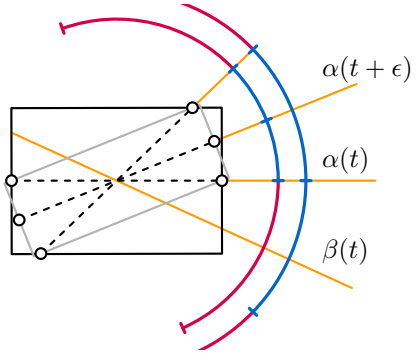

Figure 8 At time t orientation β is within the safe zone (outer blue) but at time $t+\epsilon$ the diametrical pair has changed its orientation. Since $z(t+\epsilon) < z(t)$ the safe zone shrinks and β is no longer in the safe zone at time $t+\epsilon$ (inner blue). The danger zone (red) is large enough such that β is still inside the interval at time $t+\epsilon$, if β was in the safe zone at time t. The orientations of the diametrical pair at time t and $t+\epsilon$ and of the chasing algorithm at time t are shown in yellow.

so we need to be careful to choose the right function for $J(z)$. For the other functions we choose $T(z) = z/4$ and $H(z) = c\arcsin(z)$ for a constant c that may be chosen later.

Changes in orientation and aspect ratio. To verify that the chosen functions $T(z)$ and $H(z)$ satisfy the intended requirements, and to define the function $J(z)$, we need to bound how much α and z can change over a time period of length Δt. We refer to these bounds as $\Delta\alpha(z, \Delta t)$ and $\Delta z(z, \Delta t)$, respectively. Note that, since the diameter can change discontinuously, we generally have that $\Delta\alpha(z, 0) > 0$ and $\Delta z(z, 0) > 0$.

▶ **Lemma 8.** *$\Delta\alpha(z, \Delta t) \leq \arcsin(z + \Delta t(1+z))$ for $\Delta t \leq (1-z)/(1+z)$.*

Proof. Refer to Figure 9 for illustrations. Let D be the diameter at time t, so that the width of the strip containing all points is zD. Furthermore, let D' be the diameter at time $t+\Delta t$, and let (p'_1, p'_2) be a diametrical pair at that time, such that the diametrical orientation differs an angle γ from the orientation at time t. First note that $\Delta t \geq |D - D'|/2$. Furthermore, both p'_1 and p'_2 must have been in the diametric box at time t. This additionally means that $\Delta t \geq \frac{D'}{2}\sin(\gamma) - \frac{zD}{2}$. As Δt is minimized when $D \geq D'$, we can obtain a lower bound for Δt by equalizing $(D - D')/2 = \frac{D'}{2}\sin(\gamma) - \frac{zD}{2}$. We obtain that $\Delta t \geq \frac{D'(\sin(\gamma)-z)}{1+z} \geq \frac{\sin(\gamma)-z}{1+z}$. This is equivalent to $\gamma \leq \arcsin(z + \Delta t(1+z))$. Finally note that this function is only well-defined for $\Delta t \leq \frac{1-z}{1+z}$. ◀

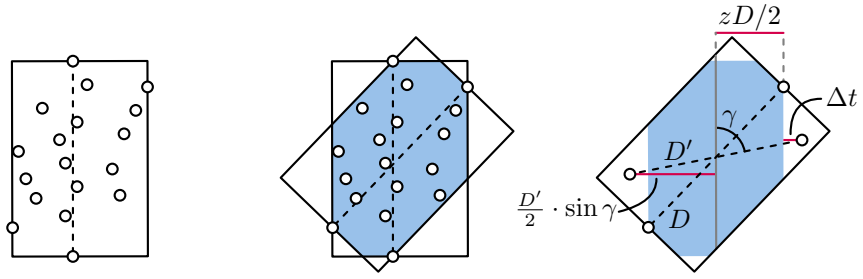

Figure 9 A diametrical box with dimensions D, zD containing all points. If the orientation of the diametrical pair changes, all points must lie in the blue area. The orientation can change further in the same direction, after some points move outside of the blue area and establish a new diameter.

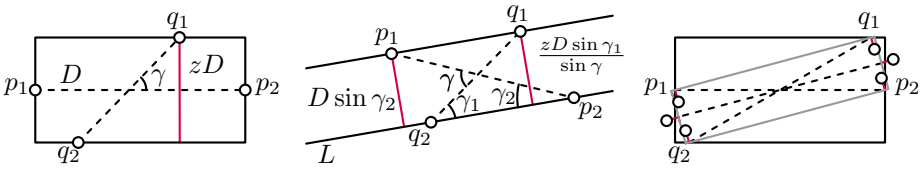

Figure 10 Illustrations for the proof of Lemma 9. (Left) A diametrical box with aspect ratio z and points p_1, p_2, q_1 and q_2 at the boundary. (Middle) A strip with points located at the boundary, the width of the strip is the minimum of the red lines. (Right) The smallest diametrical box for time $t + \epsilon$ and in red the distance the points can travel in Δt to further shrink the aspect ratio.

▶ **Lemma 9.** $\Delta z(z, \Delta t) \leq z - \frac{\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t}{1+2\Delta t}$ *for* $\Delta t \leq \sin(\frac{1}{2}\arcsin(z))/2$.

Proof. Refer to Figure 10 for illustrations. Let the diameter D at time t be realized by the pair of points (p_1, p_2) with orientation α. At time t there must also exist points q_1 and q_2 on opposite sides of the diametric box (with distance between q_1 and q_2 at most D). To minimize the aspect ratio of the diametric box at time $t + \Delta t$, we need to find the thinnest strip that contains all of p_1, p_2, q_1, and q_2. We may assume that that p_1, p_2, q_1, and q_2 are all on the boundary of the strip, and we assume w.l.o.g. that p_1 and q_1 are on the same side of the strip (same for p_2 and q_2). Let the angle between the line through q_1 and q_2 and the line through p_1 and p_2 be $\gamma \geq \arcsin(z)$. The distance between q_1 and q_2 is then $zD/\sin(\gamma)$. Furthermore, if L is a line bounding the strip, then the angle between the line through q_1 and q_2 and L is γ_1, and the angle between the line through p_1 and p_2 and L is γ_2, where $\gamma_1 + \gamma_2 = \gamma$. The width of the strip is $\max(D\sin(\gamma_2), zD\sin(\gamma_1)/\sin(\gamma))$. We show that this width is at least $D\sin(\frac{1}{2}\arcsin(z))$. This is clearly the case if $\gamma_2 \geq \frac{1}{2}\arcsin(z)$, so assume the contrary. Since the function $\sin(\gamma - \gamma_2)/\sin(\gamma)$ is increasing, it is optimal to set $\gamma = \arcsin(z)$. But then $zD\sin(\gamma_1)/\sin(\gamma) = D\sin(\gamma_1) > D\sin(\frac{1}{2}\arcsin(z))$. Thus, the width of this strip is at least $D\sin(\frac{1}{2}\arcsin(z))$. As a result, the width of the diametric box at time $t + \Delta t$ is at least $D\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t$. Since the diameter $D' \leq D + 2\Delta t$, the final aspect ratio is $z' \geq (D\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t)/(D + 2\Delta t)$. Since $D \geq 1$, we obtain that $\Delta z(z, \Delta t) \leq z - (\sin(\frac{1}{2}\arcsin(z)) - 2\Delta t)/(1 + 2\Delta t)$. Finally note that this bound is meaningful only for $\Delta t \leq \sin(\frac{1}{2}\arcsin(z))/2$. ◀

Jumping distance. Using Lemma 8 and Lemma 9 we can derive a valid function for $J(z)$. Recall that we require that $J(z)$ is at least the amount E can move in $\Delta t = 0$ time.

▶ **Lemma 10.** $J(z) = (c + 2)\arcsin(z)$ *is a valid jumping distance function.*

Proof. By Lemma 8 and Lemma 9 we get that $\Delta E(z, 0) \leq \Delta\alpha(z, 0) + H(z) - H(z - \Delta z(z, 0)) + J(z) - J(z - \Delta z(z, 0))$. Since $\Delta\alpha(z, 0) \leq \arcsin(z)$ and $\Delta z(z, 0) \leq z - \sin(\frac{1}{2}\arcsin(z))$, we get after simplification that $\Delta E(z, 0) \leq (1 + c/2)\arcsin(z) + J(z) - J(\sin(\frac{1}{2}\arcsin(z)))$. Since we require that $J(z) \geq \Delta E(z, 0)$, it suffices to show that $J(\sin(\frac{1}{2}\arcsin(z))) \geq (1 + c/2)\arcsin(z)$. Using the provided function, we get that $J(\sin(\frac{1}{2}\arcsin(z))) = (c + 2)\arcsin(z)/2$ as required, so the provided function is a valid jumping distance function. ◀

▶ **Corollary 11.** *If* β *is in* I*, then* $|\alpha - \beta| \leq (2c + 2)\arcsin(z)$.

Bounding the speed. To show that the orientation β stays within the interval I, we will argue that over a time period of $T(z)$ we can rotate β at least as far as E. As the endpoint of the safe zone moves at most as fast as E, this directly implies that if β leaves the safe zone at time t, it returns to the safe zone in the time period $(t, t + T(z)]$. More precisely, we require that $KT(z) \geq \Delta E(z, T(z))$. We need to keep up only when the safe zone does not span all orientations, that is, the above inequality must hold only when $H(z) \leq \pi/2$ or $z \leq \sin(\frac{\pi}{2c})$. For the following speed bound we choose a specific value $c = 3$ to simplify the analysis. This implies that we only need to chase α when $z \leq \sin(\frac{\pi}{6}) = \frac{1}{2}$.

In our proofs we use the following trigonometric inequalities.

▶ **Lemma 12.** *The following inequalities hold for* $0 \leq x \leq 1$*:*
1. $\sin(\lambda \arcsin(x)) \leq \lambda x$ *for* $\lambda \geq 1$
2. $\sin(\lambda \arcsin(x)) \geq \lambda x$ *for* $0 < \lambda \leq 1$.

Proof. We first show inequality (1). Let $x = \sin(y)$. We rewrite (1) into $\sin(\lambda y) \leq \lambda \sin(y)$. The derivative with respect of y is $\lambda \cos(\lambda y)$ for the left side and $\lambda \cos(y)$ for the right side. Since $\cos(y) \geq \cos(\lambda y)$ for $0 \leq y \leq \pi/\lambda$ and $\lambda \geq 1$, we get that $\sin(\lambda y) \leq \lambda \sin(y)$ for $0 \leq y \leq \pi/\lambda$. In particular, for $y = \pi/(2\lambda)$ we get that $1 = \sin(\lambda y) \leq \lambda \sin(y)$. Since $\sin(\lambda y) \leq 1$ and $\lambda \sin(y)$ attains it first maximum at $y = \pi/2$, we thus also get that $\sin(\lambda y) \leq \lambda \sin(y)$ for $0 \leq y \leq \pi/2$. Since $x = \sin(y)$ and $\sin(\pi/2) = 1$, the result follows.

For inequality (2), set $x = \sin(\frac{1}{\lambda} \arcsin(y))$. We obtain that $y \geq \lambda \sin(\frac{1}{\lambda} \arcsin(y))$, or $\frac{1}{\lambda} \geq \sin(\frac{1}{\lambda} \arcsin(y))$. As shown above, this inequality holds for $0 \leq y \leq 1$. Since $y = \sin(\lambda\pi/2) < 1$ implies $x = \sin(\frac{1}{\lambda} \arcsin(y)) = 1$, the inequality holds for $0 \leq x \leq 1$. ◀

▶ **Lemma 13.** *If* $K \geq 40$*, then* $|\beta(t) - \alpha(t)| \leq 8 \arcsin(z)$ *(using* $c = 3$*) for all times* t*.*

Proof. Consider a time t when $\beta(t)$ leaves the safe zone. We first argue that $\beta(t')$ will be in the safe zone at some time $t' \in (t, t + T(z)]$. To show this, we need to prove that $KT(z) \geq \Delta E(z, T(z))$ for $z \leq \frac{1}{2}$.

To apply the bounds of Lemmata 8 and 9, we need to ensure that $T(z) = z/4$ satisfies the bounds for Δt. For Lemma 8 we observe that $(1 - z)(1 + z)$ is decreasing and $z/4$ is increasing, and $(1 - z)(1 + z) = \frac{1}{3} \geq z/4$ for $z = \frac{1}{2}$. For Lemma 9 we can apply Lemma 12 to show that $\sin(\frac{1}{2} \arcsin(z))/2 \geq z/4$. We thus get the provided bounds on $\Delta\alpha(z, T(z))$ and $\Delta z(z, T(z))$, and as a result a bound on $\Delta E(z, T(z))$. We observe that $\Delta E(z, T(z))/T(z)$ is an increasing function in z. Since $8\Delta E(\frac{1}{2}, \frac{1}{8}) \leq 40$, we can choose $K = 40$ to satisfy $KT(z) \geq \Delta E(z, T(z))$ for $0 \leq z \leq \frac{1}{2}$.

We finally also need to argue that $\beta(t)$ does not leave I in the interval $(t, t + T(z)]$. To show this, we need to prove that $K\Delta t \geq \Delta E(z, \Delta t) - J(z)$ for all $\Delta t \in [0, T(z)]$. We again observe that $\Delta E(z, \Delta t) - J(z)$ is strictly increasing in z. For the remaining function $\Delta E(\frac{1}{2}, \Delta t) - J(\frac{1}{2})$ it is straightforward to verify that the rate of growth is much smaller than $K = 40$. Thus, $\beta(t)$ does not leave I in the time period $(t, t + T(z)]$. By repeating this argument whenever $\beta(t)$ leaves the safe zone, we can conclude that $|\beta(t) - \alpha(t)| \leq 8 \arcsin(z)$ for all times t. ◀

5.2 Lipschitz stability ratio

What remains is to analyze the approximation ratio of the chasing algorithm for both OBB and STRIP. Corollary 11 implies that the orientation β of the chasing algorithm is at most an angle $(2c + 2) \arcsin(z)$ away from the orientation of the diameter.

▶ **Lemma 14.** *If* $|\beta - \alpha| \leq (2c + 2) \arcsin(z)$*, then* $f_{\text{OBB}}(\beta) \leq (4c + 6) \min_x f_{\text{OBB}}(x)$.

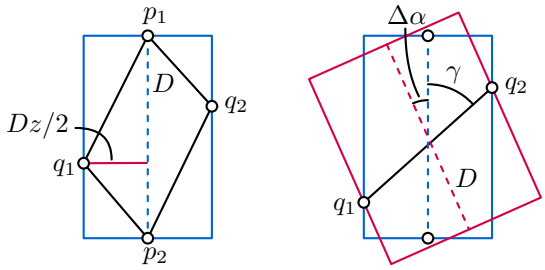

Figure 11 Illustrations supporting proof of Lemma 14. (Left) The smallest OBB, contains at least the two triangles of which the area can be calculated using the dimensions of the diametric box (in blue). (Right) The major axis of the box of the chasing algorithm (in red) has length at most D and differs in orientation by $\Delta\alpha$ from the diameter box. Its minor axis is determined by points q_1 and q_2.

Proof. Assume that at some time t we have a diametric box with diameter D and aspect ratio z, and let (p_1, p_2) be a diametrical pair. The smallest OBB must contain p_1 and p_2 and must hit the sides of the diametric box at, say, q_1 and q_2 (see Figure 11). Since the smallest OBB must contain the triangles formed by $\{p_1, p_2, q_1\}$ and $\{p_1, p_2, q_2\}$, the area of this box must be at least the sum of the areas of these two triangles, which is $D^2 z/2$.

Now consider the box of the chasing algorithm, where $\Delta\alpha = |\beta - \alpha| \leq (2c+2)\arcsin(z)$ from the orientation of the diametrical pair. We assume that the major axis of the box has length D, which is worst possible. Let the minor axis of the box be bounded by two points q_1 and q_2, where the angle between the line through q_1 and q_2 and the line through the diametrical pair is γ. Note that the distance between q_1 and q_2 is bounded by $\min(D, zD/\sin(\gamma))$. The angle between the minor axis of the box and the line through q_1 and q_2 is $\pi/2 - \gamma - \Delta\alpha$. Thus, the length of the minor axis is $\min(D, zD/\sin(\gamma))\cos(\pi/2 - \gamma - \Delta\alpha) = \min(D\sin(\gamma+\Delta\alpha), zD\sin(\gamma+\Delta\alpha)/\sin(\gamma))$. Since the function $\sin(\gamma+\Delta\alpha)/\sin(\gamma)$ is decreasing in γ, we attain the maximum when $z/\sin(\gamma) = 1$ or $\gamma = \arcsin(z)$. Thus, the area of this box is at most $D^2\sin((2c+3)\arcsin(z))$, which is at most $D^2 z(2c+3)$ by Lemma 12. We finally obtain that $f_{\text{OBB}}(\beta) \leq (4c+6) min_x f_{\text{OBB}}(x)$. ◀

▶ **Lemma 15.** *If* $|\beta - \alpha| \leq (2c+2)\arcsin(z)$, *then* $f_{\text{STRIP}}(\beta) \leq (4c+6)\min_x f_{\text{STRIP}}(x)$.

Proof. Assume that at some time t we have a diametric box with diameter D and aspect ratio z, and let (p_1, p_2) be a diametrical pair. The thinnest STRIP must contain p_1 and p_2 and must contain points q_1 and q_2 on opposite sides of the diametric box. We may assume that, for the thinnest STRIP, that all points p_1, p_2, q_1, and q_2 are on the boundary of the strip, and we assume w.l.o.g. that p_1 and q_1 are on the same side of the strip (same for p_2 and q_2). Let the angle between the line through q_1 and q_2 and the line through p_1 and p_2 be $\gamma \geq \arcsin(z)$. The distance between q_1 and q_2 is then $zD/\sin(\gamma)$. Furthermore, if L is a line bounding the STRIP, then the angle between the line through q_1 and q_2 and L is γ_1, and the angle between the line through p_1 and p_2 and L is γ_2, where $\gamma_1 + \gamma_2 = \gamma$. The width of the strip is $\max(D\sin(\gamma_2), zD\sin(\gamma_1)/\sin(\gamma))$. We show that this width is at least $D\sin(\frac{1}{2}\arcsin(z))$. This is clearly the case if $\gamma_2 \geq \frac{1}{2}\arcsin(z)$, so assume the contrary. Since the function $\sin(\gamma - \gamma_2)/\sin(\gamma)$ is increasing, it is optimal to set $\gamma = \arcsin(z)$. But then $zD\sin(\gamma_1)/\sin(\gamma) = D\sin(\gamma_1) > D\sin(\frac{1}{2}\arcsin(z))$. Thus, the width of the thinnest strip is at least $D\sin(\frac{1}{2}\arcsin(z)) \geq Dz/2$ by Lemma 12.

Now consider the strip of the chasing algorithm, with an orientation differing at most $\Delta\alpha = |\beta - \alpha| \leq (2c+2)\arcsin(z)$ from the orientation of the diametrical pair. Let width of the strip be bounded by two points q_1 and q_2, where the angle between the line through q_1 and q_2 and

the line through the diametrical pair is γ. Note that the distance between q_1 and q_2 is bounded by $\min(D, zD/\sin(\gamma))$. The angle between the vector perpendicular to the orientation of the strip and the line through q_1 and q_2 is $\pi/2 - \gamma - \Delta\alpha$. Thus, the width of the strip is $\min(D, zD/\sin(\gamma))\cos(\pi/2 - \gamma - \Delta\alpha) = \min(D\sin(\gamma + \Delta\alpha), zD\sin(\gamma + \Delta\alpha)/\sin(\gamma))$. Since the function $\sin(\gamma+\Delta\alpha)/\sin(\gamma)$ is decreasing in γ, we attain the maximum when $z/\sin(\gamma) = 1$ or $\gamma = \arcsin(z)$. Thus, the width of the strip is at most $D\sin((2c+3)\arcsin(z))$, which is at most $Dz(2c+3)$ by Lemma 12. We finally obtain that $f_{\text{STRIP}}(\beta) \leq (4c+6)min_x f_{\text{STRIP}}(x)$. ◀

By combining Lemmata 13, 14, and 15, we obtain the following bounds on the Lipschitz stability of OBB and STRIP.

▶ **Theorem 16.** *The following Lipschitz stability ratios hold for* OBB *and* STRIP*:*

- $\rho_{\text{LS}}(\text{OBB}, 40) \leq 18$,
- $\rho_{\text{LS}}(\text{STRIP}, 40) \leq 18$.

6 Lipschitz stability of principal component

The chasing algorithm does not work for the first principal component. Specifically, our scale normalization by requiring that the diameter is at least one at any point in time does not help. This can intuitively be attributed to the optimization function of PC. Rather than being defined by some form of extremal points, f_{PC} is determined by variance: although the diameter may be large, many close points may still largely determine the first principal component. We formalize this via the lemma below. It implies that requiring a minimal diameter is not sufficient for a chasing algorithm with bounded speed to approximate PC. The proof is inspired by the construction in [13] that shows the ratio between a bounding box aligned with the principal components and the optimal oriented bounding box can become infinite.

▶ **Lemma 17.** *For any constant K, there exists a point set $P(t)$ with minimum diameter 1 at all times, such that any shape descriptor that approximates the optimum of f_{PC} must move with speed strictly greater than K.*

Proof (sketch). Consider the point set P containing two points that lie on a horizontal line and form a diametrical pair for P. All other points in P form a dense subset P' that is

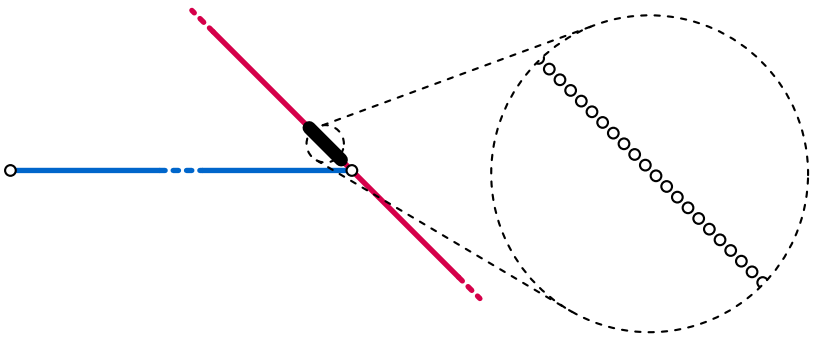

Figure 12 Construction that shows how PC can move arbitrarily fast compared, despite a minimal diameter. The two points connected by the blue line form a diametrical pair for the whole point set. The dense set of points located arbitrarily close to the right point can move around it in infinitesimally short amounts of time. Because the point set is so dense, the orientation of the first principal component (in red) will follow it regardless of how far the leftmost point is placed.

collinear with the two points that form the diametrical pair, but are located very close to only one of the two points. See Figure 12 for the construction.

The dense subset P' contains so many points that any line that differs more than ϵ from the orientation of the line l through P', will have a significantly larger sum of squared distances from the points in P to l. Hence PC follows P' regardless of the position of the two points forming the diametrical pair.

For any constant K, the points in P' can be placed and moved in such a way that in an infinitesimally small time frame, they can move around one of the points of the diametrical pair and change the orientation of PC by more than K. Thus any shape descriptor that approximates the optimum of f_{PC} must also change its orientation by more than K. ◀

7 Conclusion

We studied the topological stability (continuous solutions) and Lipschitz stability (continuous solutions with bounded speed) of three common oriented shape descriptors. Although stateless algorithms cannot achieve topological stability, we proved tight bounds on the topological stability ratio for state-aware algorithms. Our Lipschitz analysis focuses on upper bounds, showing that a chasing algorithm achieves a constant ratio for a constant maximum speed, for two of the three considered descriptors. For the first principal component, we showed that it is not related strongly enough thus cannot achieve Lipschitz stability through this approach. It remains open to establish whether there is a Lipschitz-stable algorithm for this shape descriptor and also whether lower bounds exist that are stronger than those already given by our topological stability results. We believe that our analysis techniques for the Lipschitz upper bounds are of independent interest, to analyze other problems that could be approached via a chasing algorithm.

Acknowledgments. The authors are grateful to Bettina Speckmann for insightful discussions leading up to this paper, and to Christiaan Dirkx, Pantea Haghighatkhah and Aleksandr Popov for preliminary results on the topological stability of oriented bounding boxes.

References

1 Pankaj Agarwal, Sariel Har-Peled, and Kasturi Varadarajan. Approximating extent measures of points. *J. ACM*, 51(4):606–635, 2004.

2 Gill Barequet, Bernard Chazelle, Leonidas Guibas, Joseph Mitchell, and Ayellet Tal. BOXTREE: A hierarchical representation for surfaces in 3d. *Comput. Graph. Forum*, 15(3):387–396, 1996.

3 Gill Barequet and Sariel Har-Peled. Efficiently approximating the minimum-volume bounding box of a point set in three dimensions. *J. Algorithms*, 38(1):91–109, 2001.

4 Julien Basch, Leonidas Guibas, and John Hershberger. Data structures for mobile data. *J. Algorithms*, 31(1):1–28, 1999.

5 Norbert Beckmann, Hans-Peter Kriegel, Ralf Schneider, and Bernhard Seeger. The R*-tree: An efficient and robust access method for points and rectangles. In *Proc. 1990 ACM SIGMOD International Conference on Management of Data*, pages 322–331, 1990.

6 Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002.

7 Sergei Bespamyatnikh, Binay Bhattacharya, David Kirkpatrick, and Michael Segal. Mobile facility location. In *Proc. 4th Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pages 46–53, 2000.

8 Michael Bronstein and Iasonas Kokkinos. Scale-invariant heat kernel signatures for non-rigid shape recognition. In *23rd IEEE Conference on Computer Vision and Pattern Recognition*, pages 1704–1711, 2010.

9 Luitzen Brouwer. Über Abbildung von Mannigfaltigkeiten. *Mathematische Annalen*, 71(1):97–115, 1911.

10 Dragana Brzakovic, Xiao Mei Luo, and P. Brzakovic. An approach to automated detection of tumors in mammograms. *IEEE Transactions on Medical Imaging*, 9(3):233–241, 1990.

11 Francis Chin, Cao An Wang, and Fu Lee Wang. Maximum stabbing line in 2d plane. In *Computing and Combinatorics*, pages 379–388, 1999.

12 Mark de Berg, Marcel Roeloffzen, and Bettina Speckmann. Kinetic 2-centers in the black-box model. In *Proc. 29th Symposium on Computational Geometry*, pages 145–154, 2013.

13 Darko Dimitrov, Christian Knauer, Klaus Kriegel, and Günter Rote. Bounds on the quality of the PCA bounding boxes. *Comput. Geom.*, 42(8):772–789, 2009.

14 Stephane Durocher and David Kirkpatrick. The steiner centre of a set of points: Stability, eccentricity, and applications to mobile facility location. *International Journal of Computational Geometry & Applications*, 16(04):345–371, 2006.

15 Stephane Durocher and David Kirkpatrick. Bounded-velocity approximation of mobile Euclidean 2-centres. *International Journal of Computational Geometry & Applications*, 18(03):161–183, 2008.

16 Herbert Freeman and Ruth Shapira. Determining the minimum-area encasing rectangle for an arbitrary closed curve. *Commun. ACM*, 18(7):409–413, 1975.

17 Stefan Gottschalk, Ming Lin, and Dinesh Manocha. OBBTree: A hierarchical structure for rapid interference detection. In *Proc. 23rd Annual Conference on Computer Graphics and Interactive Technique*, pages 171–180, 1996.

18 Xianfeng Gu, Yalin Wang, Tony Chan, Paul Thompson, and Shing-Tung Yau. Genus zero surface conformal mapping and its application to brain surface mapping. *IEEE Trans. Med. Imaging*, 23(8):949–958, 2004.

19 Leonidas Guibas. Kinetic data structures. In Dinesh Mehta and Sartaj Sahni, editors, *Handbook of Data Structures and Applications*, pages 23–1–23–18. Chapman and Hall/CRC, 2004.

20 Antonin Guttman. R-trees: A dynamic index structure for spatial searching. In *Proc. 1984 ACM SIGMOD International Conference on Management of Data*, pages 47–57, 1984.

21 András Kelemen, Gábor Székely, and Guido Gerig. Elastic model-based segmentation of 3-D neuroradiological data sets. *IEEE Trans. Med. Imaging*, 18(10):828–839, 1999.

22 James Klosowski, Martin Held, Joseph Mitchell, Henry Sowizral, and Karel Zikan. Efficient collision detection using bounding volume hierarchies of k-DOPs. *IEEE Trans. Vis. Comput. Graph.*, 4(1):21–36, 1998.

23 Kyle Sykes David Letscher and Kyle Sykes. On the stability of medial axis of a union of disks in the plane. In *Proc. 28th Canadian Conference on Computational Geometry*, pages 29–33, 2016.

24 Wouter Meulemans, Bettina Speckmann, Kevin Verbeek, and Jules Wulms. A framework for algorithm stability and its application to kinetic euclidean MSTs. In *Proc. 13th LATIN*, LNCS 10807, pages 805–819, 2018.

25 Joseph O'Rourke. Finding minimal enclosing boxes. *International Journal of Parallel Programming*, 14(3):183–199, 1985.

26 Nick Roussopoulos and Daniel Leifker. Direct spatial search on pictorial databases using packed r-trees. In *Proc. 1985 ACM SIGMOD International Conference on Management of Data*, pages 17–31, 1985.

27 Timos Sellis, Nick Roussopoulos, and Christos Faloutsos. The R+-tree: A dynamic index for multi-dimensional objects. In *Proc. 13th International Conference on Very Large Data Bases*, pages 507–518, 1987.

28 Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view convolutional neural networks for 3D shape recognition. In *2015 IEEE International Conference on Computer Vision*, pages 945–953, 2015.

29 Godfried Toussaint. Solving geometric problems with the rotating calipers. In *Proc. IEEE Melecon*, volume 83, page A10, 1983.

30 Eugene Tyrtyshnikov. *A brief introduction to numerical analysis*. Springer Science & Business Media, 2012.

31 Gino van den Bergen. Efficient collision detection of complex deformable models using AABB trees. *J. Graphics, GPU, & Game Tools*, 2(4):1–13, 1997.

32 Ivor van der Hoog, Marc van Kreveld, Wouter Meulemans, Kevin Verbeek, and Jules Wulms. Topological stability of kinetic k-centers. *CoRR*, abs/1810.00794, 2018. URL: `http://arxiv.org/abs/1810.00794`.

33 Manik Varma and Debajyoti Ray. Learning the discriminative power-invariance trade-off. In *Proc. IEEE 11th International Conference on Computer Vision*, pages 1–8, 2007.

34 Jin Xie, Guoxian Dai, Fan Zhu, Edward Wong, and Yi Fang. DeepShape: Deep-learned shape descriptor for 3D shape retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(7):1335–1345, 2017.

35 Hao Zhang, Alexander Berg, Michael Maire, and Jitendra Malik. SVM-KNN: discriminative nearest neighbor classification for visual category recognition. In *Proc. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2126–2136, 2006.

36 Yu Zhong. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference*, pages 689–696. IEEE, 2009.

37 Barbara Zitová and Jan Flusser. Image registration methods: a survey. *Image Vision Comput.*, 21(11):977–1000, 2003.