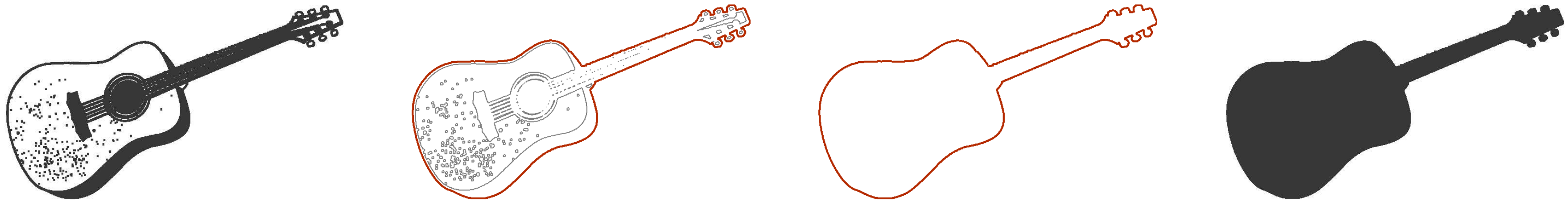
Analysis of 3D Shapes (IN2238)

Frank R. Schmidt
Matthias Vestner

Summer Semester 2016

2D Shape Matching 3 / 24

2D Objects



To summarize the last lectures, we assume that a 2D object is an open set $O \subset \mathbb{R}^2$ such that $C = \partial O$ is a closed contour that is parameterized via

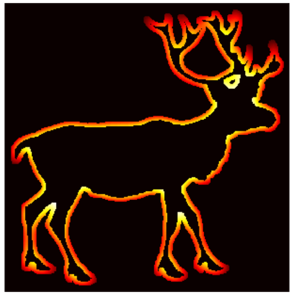
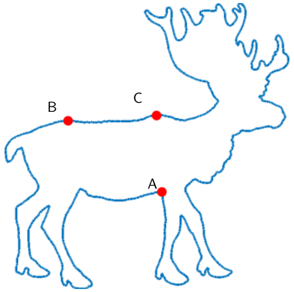
$$c\colon \mathbb{S}^1 \to \mathbb{R}^2.$$

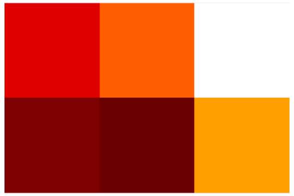
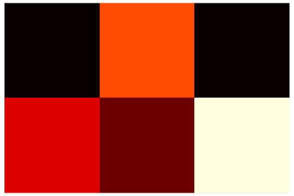
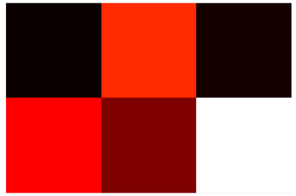
If we choose a diffeomorphism $m\colon \mathbb{S}^1 \to \mathbb{S}^1$ defined on the parametrization domain $\mathbb{S}^1$, we obtain a different parametrization

$$c \circ m\colon \mathbb{S}^1 \to \mathbb{R}^2.$$

of the contour $C = \partial O$. Thus, $c\colon \mathbb{S}^1 \to \mathbb{R}^2$ is **not a unique** representation.

2D Shapes

We also assume that we have a pointwise feature representation $f\colon \mathbb{S}_1 \to \mathbb{R}^k$ such that shape-equivalent curves $c_1, c_2\colon \mathbb{S}^1 \to \mathbb{R}^2$ lead to the same feature representation $f_1 \equiv f_2$.

The problem of shape matching can now be formulated as finding a mapping $m\colon \mathbb{S}^1 \to \mathbb{S}^1$ such that

$$f_1(s) \approx f_2(m(s)) \qquad \text{for all } s \in \mathbb{S}^1$$

2D Shape Distance (naïve version)

Solving the shape matching problem results in minimizing the following energy

$$E_0(m) = \int_{\mathbb{S}^1} \mathrm{dist}_{\mathcal{F}}(f_1(s), f_2 \circ m(s)) \mathrm{ds} \qquad m\colon \mathbb{S}^1 \to \mathbb{S}^1,$$

where $\mathrm{dist}_{\mathcal{F}}(\cdot,\cdot)$ measures the similarity of two features in the k-dimensional feature space $\mathbb{R}^k$.

Since $E_0(m) \geqslant 0$ for all m, we can define for two curves c_1 and c_2 their "distance" as

$$\mathrm{dist}_0(c_1, c_2) = \min_{m\colon \mathbb{S}^1 \to \mathbb{S}^1} \int_{\mathbb{S}^1} \mathrm{dist}_{\mathcal{F}}(f_1(s), f_2 \circ m(s)) \mathrm{ds},$$

where f_i are the feature representation of c_i for $i = 1, 2$.

Metric and Semi-Metric

We would like to use dist_0 as a distance function for shapes. Nonetheless, we need some extra work in order to obtain a meaningful shape distance. To this end, we need to differentiate between a metric and a semi-metric.

Definition 1. Given a space X, we call $d\colon X \times X \to \mathbb{R}_0^+$ a **metric** and X a **metric space** if

$$d(x,y) = 0 \quad \Leftrightarrow \quad x = y \qquad \text{(Positive Definiteness)}$$
$$d(x,y) = d(y,x) \qquad \text{(Symmetry)}$$
$$d(x,z) \leqslant d(x,y) + d(y,z) \qquad \text{(Triangle Inequality)}$$

If d is only positive definite and symmetric, but does not necessarily satisfy the triangle inequality, we call d a **semi-metric**.

Some Properties of dist_0

The mapping dist_0 has the following properties for all object curves $c_1, c_2 \colon \mathbb{S}^1 \to \mathbb{R}^2$

$$\begin{aligned} \mathrm{dist}_0(c_1, c_2) &= 0 && \text{for } c_1 \sim c_2 \\ \mathrm{dist}_0(c_1, c_1 \circ m) &= 0 && \text{for all bijective } m \colon \mathbb{S}^1 \to \mathbb{S}^1 \end{aligned}$$

Nonetheless, the symmetry

$$\mathrm{dist}_0(c_1, c_2) = \mathrm{dist}_0(c_2, c_1)$$

is only possible if we restrict matchings $m \colon \mathbb{S}^1 \to \mathbb{S}^1$ to bijective functions.

Then we expect that given the optimal matching m between c_1 and c_2 would lead to the optimal matching m^{-1} between c_2 and c_1.

Linear Assignment

Looking for bijections $m\colon \mathbb{S}^1 \to \mathbb{S}^1$ lead to the LAP, which only considers permutations as a valid matching.

This means, we have with $s_k = \exp\left(\frac{2\pi k}{N} i\right)$

$$
\begin{aligned}
E_0(m) &= \int_{\mathbb{S}^1} \mathrm{dist}_{\mathcal{F}}(f_1(s), f_2 \circ m(s)) \mathrm{ds} \\
&\approx \sum_{k=1}^{N} \mathrm{dist}_{\mathcal{F}} \left[f_1(s_k), f_2 \circ m(s_k) \right] \cdot \frac{2\pi}{N} \\
E_0(m^{-1}) &= \int_{\mathbb{S}^1} \mathrm{dist}_{\mathcal{F}}(f_2(s), f_1 \circ m^{-1}(s)) \mathrm{ds} \\
&\approx \sum_{k=1}^{N} \mathrm{dist}_{\mathcal{F}} \left[f_1 \circ m^{-1}(s_k), f_2(s_k) \right] \cdot \frac{2\pi}{N}
\end{aligned}
$$

Restrictions of LAP

The LAP leads to a symmetric distance between object curves c_1 and c_2, if the features are sampled equidistantly with respect to the chosen parametrization. As a result, dist_0 is not independent of the parametrization, even for the same curve. Thus, LAP does **not** compute a shape distance.

This problem can be resolved by only allowing uniform parameterizations of curves. Nonetheless, this might constrain the choice of possible matchings rather dramatically.

Another disadvantage of LAP is that it does not smoothly map one contour onto the other. In other words, $m\colon \mathbb{S}^1 \to \mathbb{S}^1$ is just a bijection and not a homeomorphism or diffeomorphism.

Our goal is it now to define a matching energy that only considers diffeomorphic matchings $m\colon \mathbb{S}^1 \to \mathbb{S}^1$. In addition, the minimum of such an energy should give rise to a semi-metric for shapes.

Diffeomorphic Matching

Given two feature representations $f_1, f_2\colon \mathbb{S}^1 \to \mathbb{R}^k$, we want to define a matching energy that provides us with the same minimal value for $g_1 := f_1 \circ \varphi$ and $g_2 := f_2 \circ \varphi$ given a diffeomorphic reparameterization $\varphi\colon \mathbb{S}^1 \to \mathbb{S}^1$.

If m is the optimal matching between f_1 and f_2, we would expect that $\tilde{m} := \varphi^{-1} \circ m \circ \varphi$ is the optimal matching between g_1 and g_2.

For the previously defined E_0 we have

$$\int_{\mathbb{S}^1} \mathrm{dist}_{\mathcal{F}}(g_1(s), g_2 \circ \tilde{m}(s))\mathrm{ds} = \int_{\mathbb{S}^1} \mathrm{dist}_{\mathcal{F}}(f_1(\varphi(s)), f_2 \circ m(\varphi(s)))\mathrm{ds}$$
$$= \int_{\varphi(\mathbb{S}^1)} \mathrm{dist}_{\mathcal{F}}(f_1(\varphi \circ \varphi^{-1}(s)), f_2 \circ m(\varphi \circ \varphi^{-1}(s))) \cdot \dot{\varphi}(\varphi^{-1}(s))^{-1}\mathrm{ds}$$
$$= \int_{\mathbb{S}^1} \mathrm{dist}_{\mathcal{F}}(f_1(s), f_2 \circ m(s)) \cdot \dot{\varphi}(\varphi^{-1}(s))^{-1}\mathrm{ds}$$

Geometrically Motivated Distance

Let us assume that two contours $C_1, C_2 \subset \mathbb{R}^2$ together with their diffeomorphic parametrization $c_i \colon \mathbb{S}^1 \to C_i$ are given. Further let $f_i \colon \mathbb{S}^1 \to \mathbb{R}^k$ be their feature reparameterization.

Then we define the torus $T := C_1 \times C_2$ and the cost function

$$D \colon T \to \mathbb{R}_0^+ \qquad (x, y) \mapsto \mathrm{dist}_{\mathcal{F}}(f_1 \circ c_1^{-1}(x), f_2 \circ c_2^{-1}(y))$$

Note that neither T nor D depend on the specific parametrization c_1 or c_2.

Given a matching $m \colon \mathbb{S}^1 \to \mathbb{S}^1$, we define the 1D manifold

$$\Gamma(m) := \{(x, y) \in T | m \circ c_1^{-1}(x) = c_2^{-1}(y)\}$$

and a new energy

$$E_1^{(C_1,C_2)}(m) = \int_{\Gamma(m)} D(s) \mathrm{ds}.$$

Line Integral

Given a contour $\Gamma \subset \mathbb{R}^N$ and a scalar function $f\colon \Gamma \to \mathbb{R}$, we would like to define the line integral $\int_\Gamma f(s)\mathrm{ds}$. To this end, let us assume that we have a diffeomorphic coordinate map $c : [0,1] \to \Gamma$.

Then, we can define the line integral as

$$\begin{aligned}\int_\Gamma f(s)\mathrm{ds} &= \lim_{N\to\infty} \sum_{i=1}^{N} f \circ c\left(\frac{i}{N}\right) \left\| c\left(\frac{i}{N}\right) - c\left(\frac{i-1}{N}\right)\right\| \\ &= \int_0^1 f \circ c(t) \cdot \left\|c'(t)\right\| \mathrm{dt} \\ &= \int_0^1 f \circ c(t) \cdot \sqrt{\det\left(c'(t)^\top c'(t)\right)}\mathrm{dt}\end{aligned}$$

Diffeomorphic Matching

This definition leads to the following representation of the energy E_1

$$E_1^{(C_1,C_2)}(m) = \int_{\Gamma(m)} D(s)\mathrm{ds} = \int_{\Gamma(m)} \mathrm{dist}_{\mathcal{F}}(f_1 \circ c_1^{-1}(s_1), f_2 \circ c_2^{-1}(s_2))\mathrm{ds}$$
$$= \int_{\mathbb{S}^1} \mathrm{dist}_{\mathcal{F}}(f_1(t), f_2 \circ m(t)) \cdot \sqrt{\dot{c}_1(t)^2 + \frac{\mathrm{d}}{\mathrm{dt}}(c_2 \circ m)(t)^2}\mathrm{dt},$$

which becomes for uniformly parameterized curves of same length 2π

$$E_1(m) = \int_{\mathbb{S}^1} \mathrm{dist}_{\mathcal{F}}(f_1(t), f_2 \circ m(t)) \cdot \sqrt{1 + \dot{m}(t)^2}\mathrm{dt},$$

While this energy looks rather technical, it does not depend on the specific parameterizations of c_1 and c_2.

Properties of dist_1

Since only feature information is used, we can see that

$$\mathrm{dist}_1(C_1, C_2) := \underset{m\colon \mathbb{S}^1 \to \mathbb{S}^1}{\operatorname{argmin}} E_1^{(C_1,C_2)}(m)$$

is a positive function defined on a shape space.

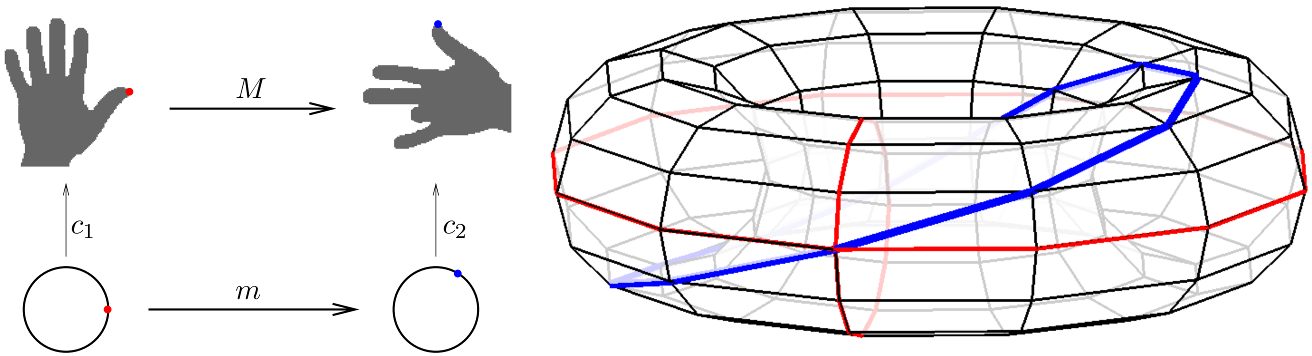
We obtain for a contour C that

$$\mathrm{dist}_1(C, C) \leqslant E_1^{(C,C)}(\mathrm{id}) = 0.$$

Whether dist_1 is positive definite depends on the chosen features.
For curvature, dist_1 is positive definite.

Since, we always have $E_1^{(C_1,C_2)}(m) = E_1^{(C_2,C_1)}(m^{-1})$, we know that dist_1 is symmetric. Thus, dist_1 provides us with a semi-metric of our shape space.

Matching Contour



A **shape matching** is a mapping $M\colon \partial O_1 \to \partial O_2$ that maps corresponding boundary points onto one another. Thus, we assume that N points are selected from each contour.

We are interested in the **matching contour** $\Gamma(m)$, which can be described as a closed contour on the grid defined by N^2 product nodes.

Graph Representation

Let us assume that we have N ordered points $x_0, \ldots, x_{N-1} \in \mathbb{R}^2$ of the first contour C_1 and N ordered points $y_0, \ldots, y_{N-1} \in \mathbb{R}^2$ of the second contour. In addition, we have the distance of the features stored in $D \in \mathbb{R}^{N \times N}$, *i.e.*, $d_{ij} = \mathrm{dist}_{\mathcal{F}}(f_1 \circ c_1^{-1}(x_i), f_2 \circ c_2^{-1}(y_j))$.

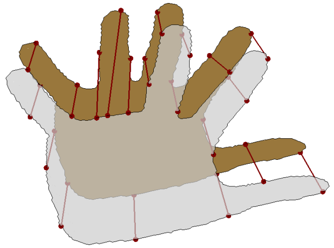
Now we define the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ that discretizes the torus T:

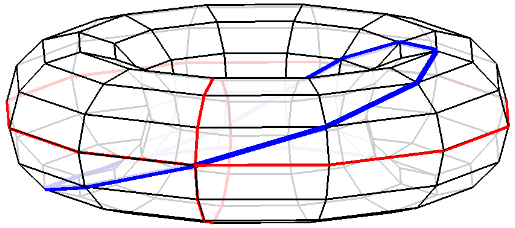
$$
\begin{aligned}
\mathcal{V} =& \{0, \ldots N-1\} \times \{0, \ldots N-1\} \\
\mathcal{E} =& \{[(i,j), (i \oplus 1, j)] | (i,j) \in \mathcal{V}\} \cup && \text{(horizontal edges)} \\
& \{[(i,j), (i, j \oplus 1)] | (i,j) \in \mathcal{V}\} \cup && \text{(vertical edges)} \\
& \{[(i,j), (i \oplus 1, j \oplus 1)] | (i,j) \in \mathcal{V}\} && \text{(diagonal edges)},
\end{aligned}
$$

where $a \oplus b := (a+b) \mod N$.

In a last step we need to define a weight function $w \colon \mathcal{E} \to \mathbb{R}$ that encodes our energy function E_1.

Edge Weights





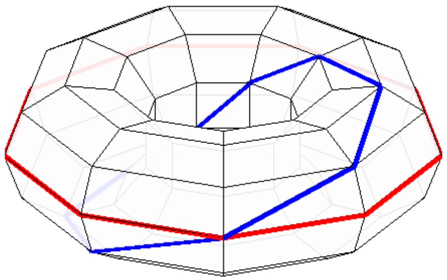
The optimal $m\colon \mathbb{S}^1 \to \mathbb{S}^1$ shall minimize the energy $E_1(m) := \int_{\Gamma(m)} D(s)\mathrm{d}s$.

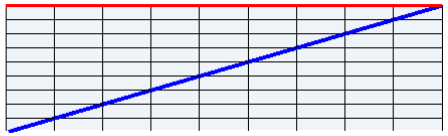
Since every edge in $\mathcal{G}$ corresponds to a potential subset of $\Gamma(m)$, we define

$$w((i_1, j_1), (i_2, j_2)) := \frac{D_{i_1,j_1} + D_{i_2,j_2}}{2} \sqrt{\|v_1 - u_1\|^2 + \|v_2 - u_2\|^2}$$
$$\approx \int_{\overrightarrow{uv}} D(s)\mathrm{d}s,$$

where $u = (u_1, u_2) = (x_{i_1}, y_{j_1})$ and $v = (v_1, v_2) = (x_{i_2}, y_{j_2})$.

Dynamic Time Warping

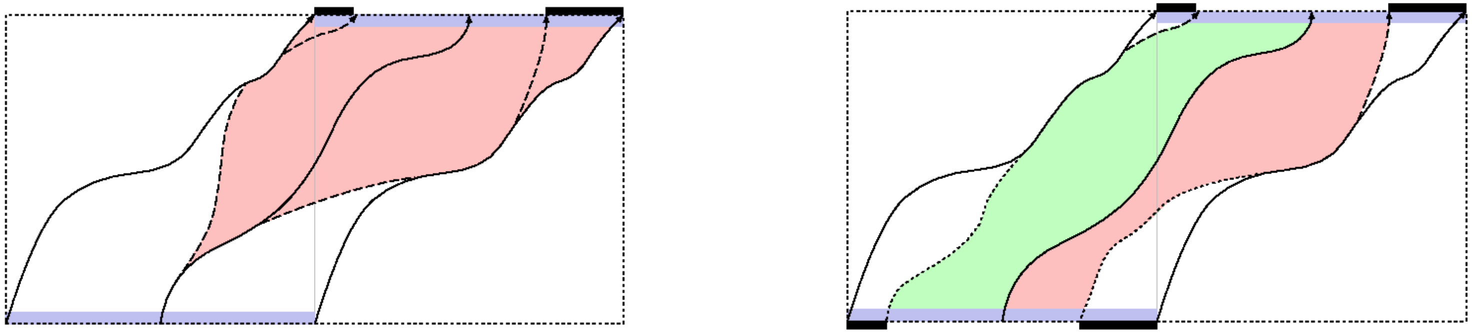




1. Guess an initial correspondence $c_1(x)$ on Shape 1 and $c_2(y)$ on Shape 2.
2. Cut the torus open along the curves $\{x\} \times \mathbb{S}^1$ and $\mathbb{S}^1 \times \{y\}$.
3. Find the shortest path between (x, y) and $(x + 2\pi, y + 2\pi)$.

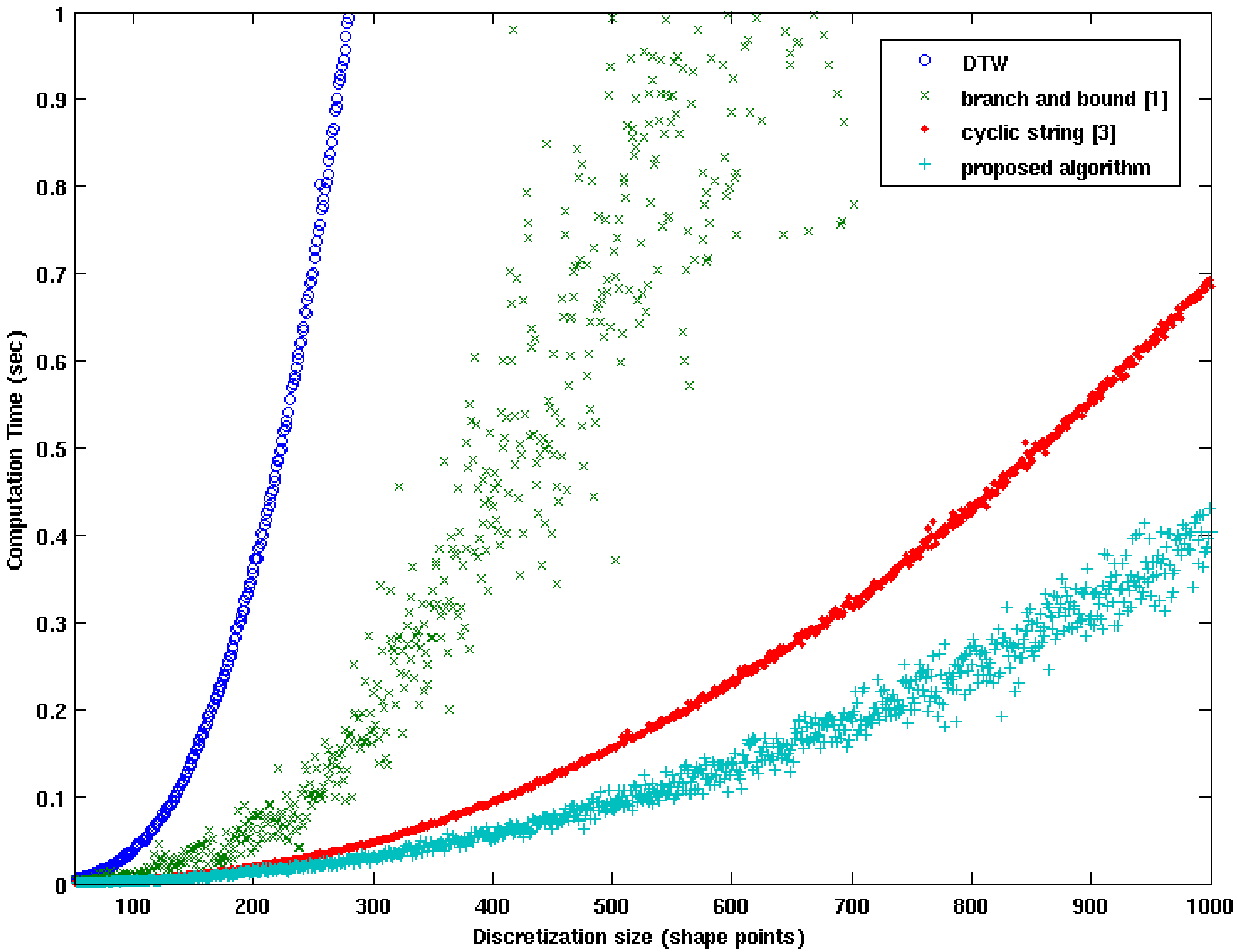
- Step 3 can be done efficiently using dynamic time warping. $\mathcal{O}(\mathbf{N^2})$
- Iterating over initial correspondences slows the method down. $\mathcal{O}(\mathbf{N^3})$

Matching in Subcubic Runtime



1. Iteratively, divide each *searching region* into two regions.
2. Compute the shortest path for the *boundary regions* independently.
3. This leads to a better *region division*.

Runtime



DTW
branch and bound [1]
cyclic string [3]
proposed algorithm
Computation Time (sec)
Discretization size (shape points)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
100
200
300
400
500
600
700
800
900
1000

Literature

Applications of Dynamic Time Warping

- Sankoff and Kruskal, *Time Warps, String Edits and Macromolecules: The Theory and Practice of Sequence Comparison*, 1983, Addison-Wesley, Reading, MA.
- Geiger et al., *Dynamic programming for detecting, tracking and matching deformable contours*, 1995, IEEE PAMI 17 (3), 294–302.

Matching as Shortest Circular Path

- Maes, *On a cyclic string-to-string correction problem*, 1990.
- Schmidt et al., *Fast Matching of Planar Shapes in Sub-cubic Runtime*, 2007, IEEE ICCV.