P23

klcluster: CENTER-BASED CLUSTERING OF TRAJECTORIES

Kevin Buchin **Anne Driemel** **Natasja van de L'Isle** **André Nusser**

TU/e Technische Universiteit Eindhoven University of Technology

max planck institut informatik

UNIVERSITÄT BONN

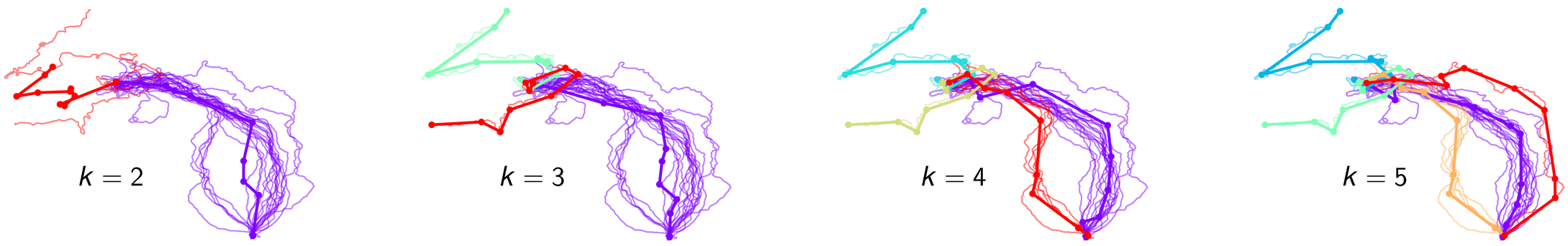


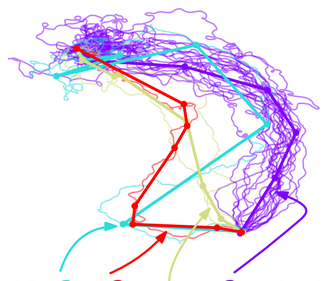
Figure: A clustering of pigeon trajectories for different values of k, i.e., different numbers of clusters. For $k = 3$, $k = 4$, and $k = 5$ there are two clusters each that contain only one curve which are outlier curves, and the remaining clusters show the different routes of the pigeon.

PROBLEM DEFINITION

Given:

A set of polylines $\mathcal{T}$ and $k, \ell \in \mathbb{N}$

Wanted:



curves are mapped to closest center

Center polylines $c_1, c_2, \ldots, c_k$, such that

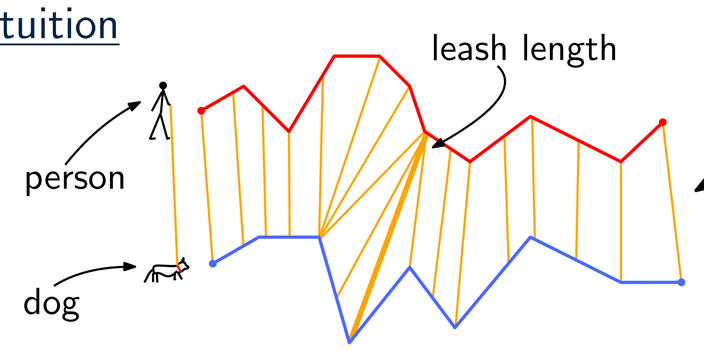
- they induce clusters $C_1 \cup C_2 \cup \cdots \cup C_k = \mathcal{T}$
- each c_i has at most ℓ nodes
- they minimize $\max_{i \in [k], t \in C_i} d_F(t, c_i)$

FRÉCHET DISTANCE

RECAP

- curve similarity measure
- has a notion of traversals
- especially good for spatio-temporal data

Intuition



Formal Definition

$$d_F(\pi, \sigma) := \min_{f \in \mathcal{T}_\pi, g \in \mathcal{T}_\sigma} \max_{t \in [0,1]} \left\| \pi_{f(t)} - \sigma_{g(t)} \right\|$$

$\pi, \sigma =$ polygonal curves
$\mathcal{T}_\pi =$ monotone, continuous traversals of π
$\mathcal{T}_\sigma =$ monotone, continuous traversals of σ

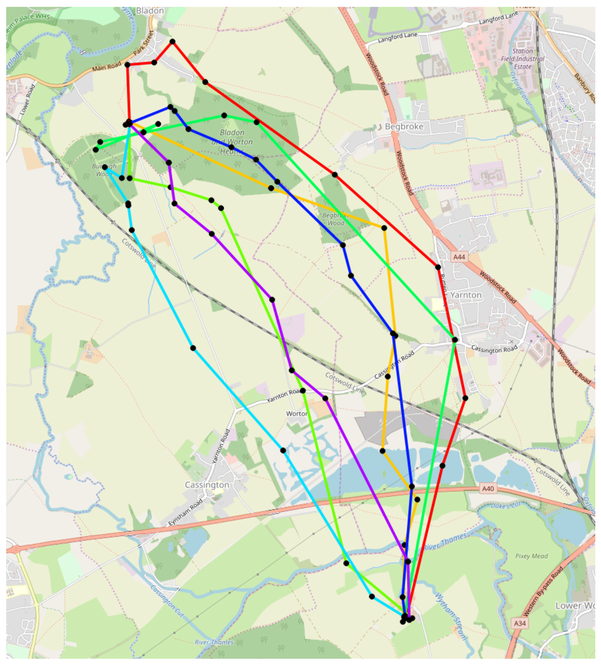


Figure: Center curves of largest clusters of seven pigeons with same start end ending points

ALGORITHM

High Level

1. Compute Initial Clustering
2. Improve Centers
3. Update Clusters
4. If the centers changed , GOTO 2

Details

①

- use Gonzalez' Algorithm

1. choose arbitrary curve, compute its ℓ-simplification, and call the result c_1
2. for $i \in \{2, \ldots, k\}$: choose input curve that is farthest from $c_1, \ldots, c_{i-1}$, compute its ℓ-simplification and call it c_i

②

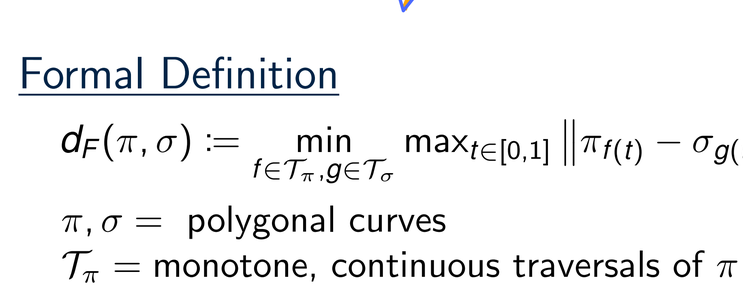
- **naive:** find the best ℓ-simplification among the curves in each cluster
- **issue:** dependent on input curves, and none of them might be suitable
- **we propose:** Fréchet centering

③

- match each curve to its closest center

FRÉCHET CENTERING

- **Idea:** Use matchings induced by the traversals to improve the center curve
- **Algorithm:**
  For each cluster
  - compute matching between center and curves
  - for each node of the old center curve: move it to the center of the matched points



IMPLEMENTATION AND EXPERIMENTS

- Implementation* in modern C++
- Uses recent advances for fast Fréchet distance computation
- Comparison to complete linkage with k-means center
- Experiments on two data sets:
  - Pigeon Data
  - Handwritten Characters

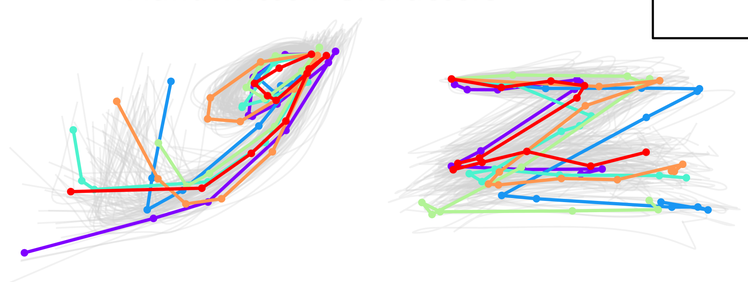


Figure: Two examples for a character curve set (gray) and their center curves (colorful). The centers adapt to different shapes of the characters (left) and to sizes (right).

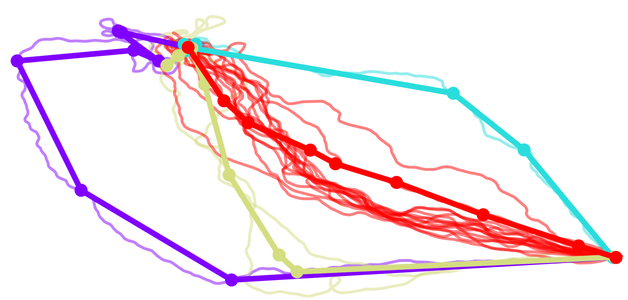


Figure: Multiple flight trajectories of a single pigeon.

	time (s)	diam.	radius
baseline approach	21.83	0.076	0.0104
our approach	2.04	0.076	0.0092

Table: Runtime comparison on the pigeon data set of our approach and the baseline approach. The quality of our approach is slightly better on the two measures while being one order of magnitude faster.

*Code available at: https://gitlab.com/anusser/klcluster-sigspatial19