GRAPP 2020 - 15th International Conference on Computer Graphics Theory and Applications

Matching-Aware Shape Simplification

Authors

Enrico S. Miranda, Rogério Luís C. Costa, Paulo Dias and José Moreira

Acknowledgments This work is partially funded by National Funds through the FCT - Foundation for Science and Technology, in the context of the projects UID/CEC/00127/2013 and POCI-01-0145-FEDER-032636.

FCT Fundação para a Ciência e a Tecnologia

IEETA
Institute of Electronics and Informatics Engineering of Aveiro

University of Aveiro

Context

University of Aveiro

- **Spatiotemporal Data**
  - Moving Objects
    - Moving Points
    - Moving Lines
    - **Moving Regions**
- **Sample uses**

Satellite images of the evolution of an iceberg

Evolution of burned areas

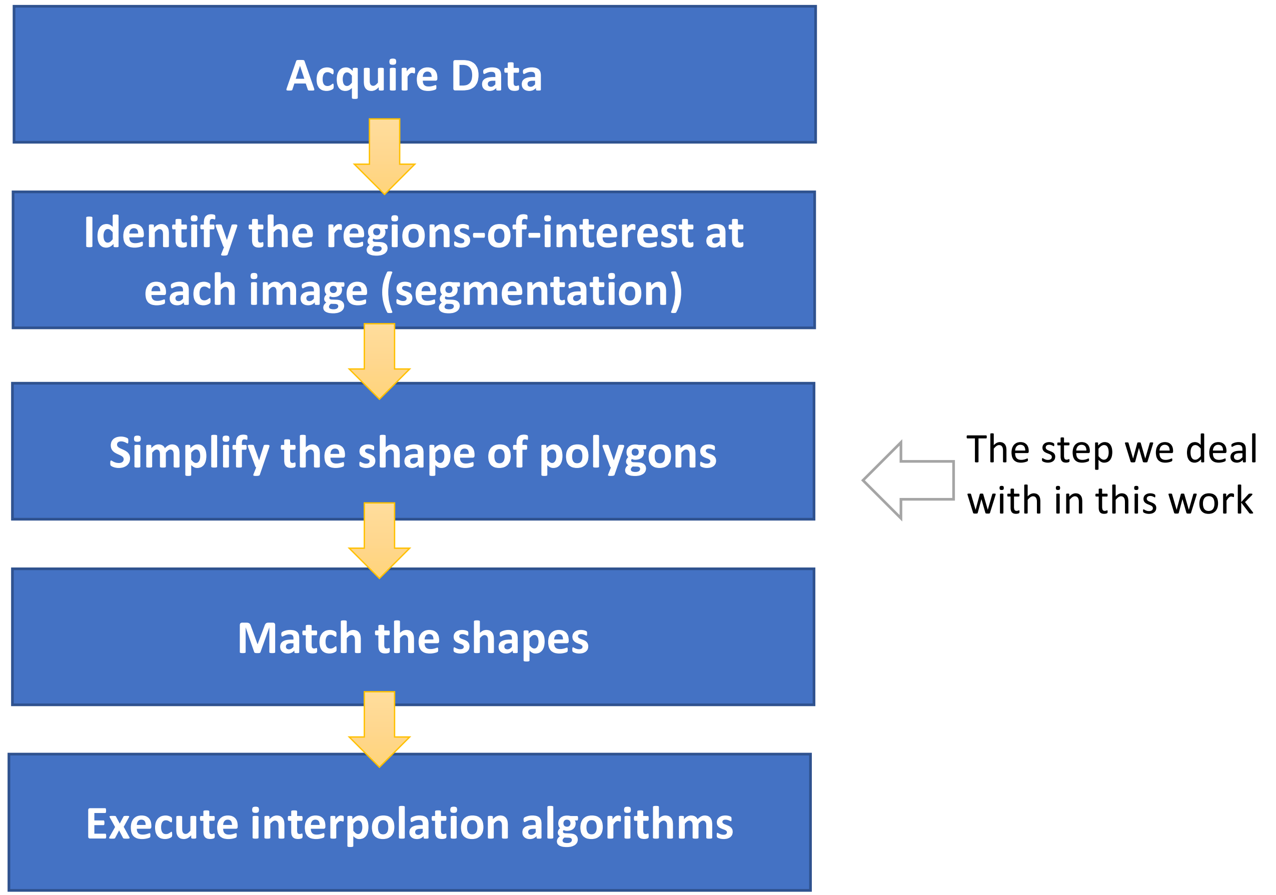
- **Continuous Representation**
  - Sliced representation (Forlizzi, L. et al. 2000) - *moving region* is represented by: (i) geometry + (ii) function (to represent the object's evolution)
  - *Region interpolation problem*: With object representations at two timestamps, use a *function* to interpolate the representations

Simulating the burnt area evolution
using a region interpolation algorithm

Continuous Representation

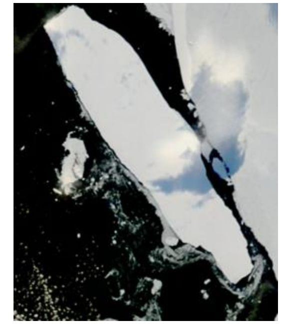
University of Aveiro

- **Data Preparation Workflow**



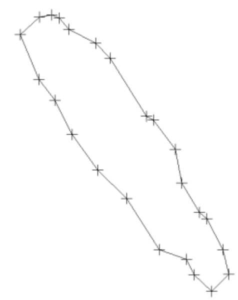
Polygon Simplification

- In simplification algorithms focused in one-by-one simplification...
  - May select vertices that are distant from their corresponding in another polygon
  - Corresponding shapes may have different number of vertices (but some interpolation algorithms require the same number of vertices)

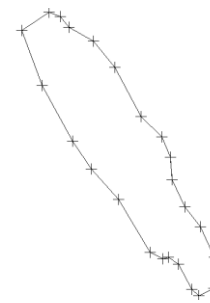


(a) Raw data - ice01

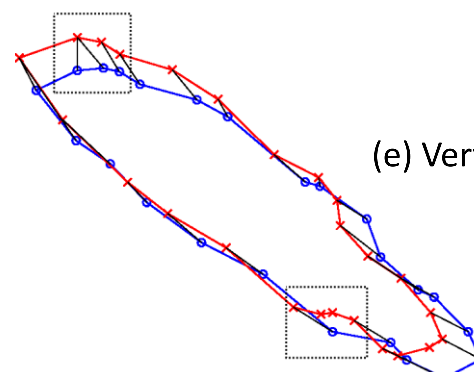
(b) Raw data - ice02



(c) Simplified Polygon - ice01



(d) Simplified Polygon - ice02



(e) Vertex matching

Our proposal

- Simplify two shapes simultaneously
- Considers the importance of vertices
  - to each shape representation
  - to inter-shape correspondence

- Objectives
  - Reduce adjustments during matching (e.g. adding extra vertices)
  - Provide locally-aware vertices to matching algorithms (to reduce vertex matching complexity)

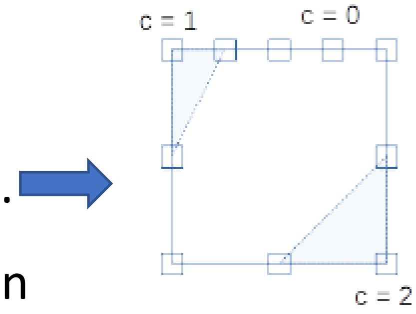
Matching-Aware Simplification

University of Aveiro

- Considering two geometries *P* and *Q*, and the vertices *p and q* ($p \in P, q \in Q$)
- For each vertex *p*, the loss on removing *p* is:

$$cost_p = max(cost\_single_p, cost\_matched_p * t\_factor)$$

  - $cost\_single_p$ = cost to remove for a single shape (e.g. the area of the triangle given by *p*, *p-1* and *p+1*).



  - *t_factor* = preference to local or temporal information
  - $cost\_matched_p$ = cost for loss of feature representation on Q:

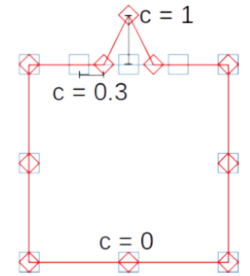
$$cost\_matched_p = max(cost\_unique\_feature_p, cost\_matched\_feature_q)$$

Matching-Aware Simplification

A significant vertex represents either:

- A feature present in Q and not in P

$$cost\_unique\_feature_p = min(d_{pq})$$

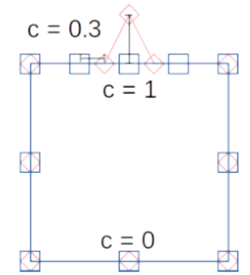


(a) Distinct features cost on Q (P on background)

- A vertex on P needed to morph into the feature of Q:

$$cost\_matched\_feature_p = max(d_{pq})$$

$$d_{pq} = min(d_{kq})$$

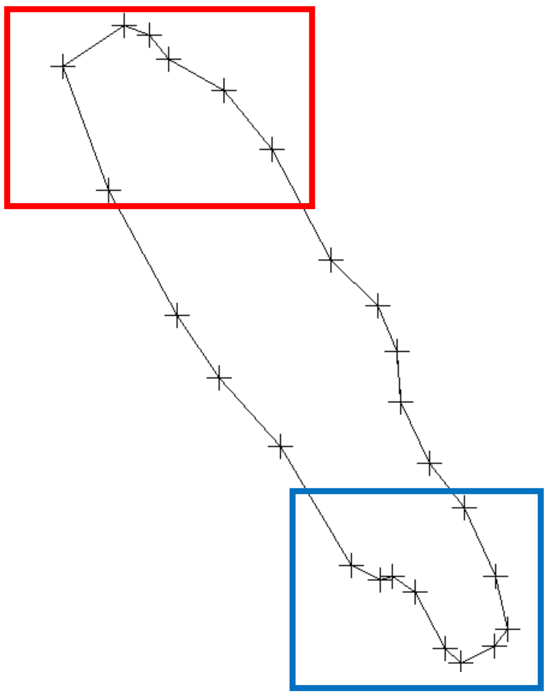


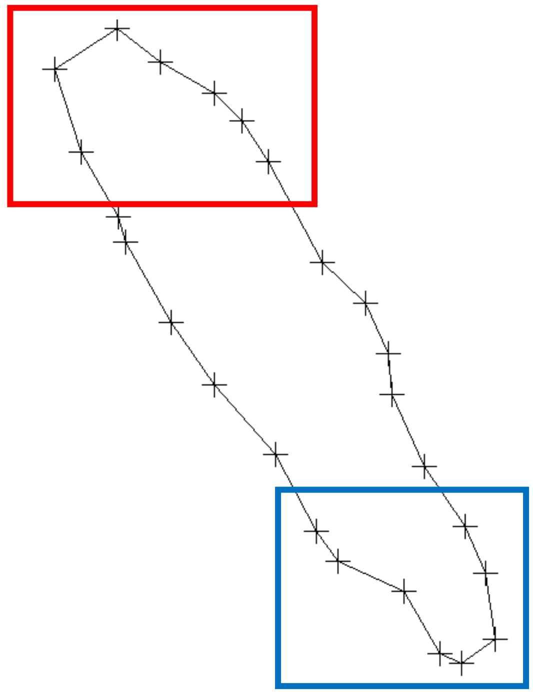
(b) Matching vertices on P (Q on background)

- Our algorithm starts removing the vertex in *P* with the lowest cost, then remove the one from *Q*, then remove another from *P*, and so on until *P* and *Q* have a desired size.

- **Simplification – Real World Data** (Iceberg B-15a)

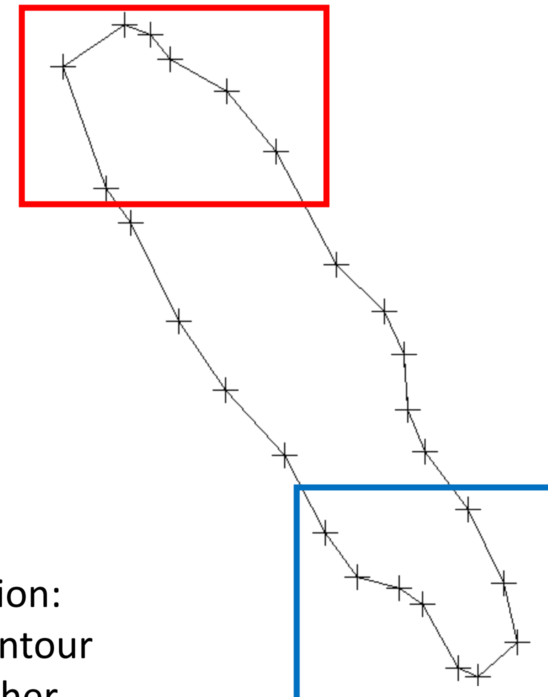
Douglas–Peucker algorithm





Matching-Aware Simplification: density of points along the contour more similar than that of other methods

Visvalingam-Whyatt algorithm

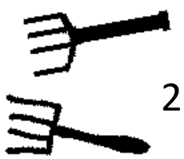


- **Simplification and Matching (216 Binary Shape Database)**
  - Dataset: 216 images from 18 classes
  - Executed each of the evaluated simplification algorithms
  - Executed the matching using the algorithm from Van Kaick et al.*
  - Recorded the number of *anomalous* points during matching:
    - vertices without correspondence
    - vertices with multiple correspondences

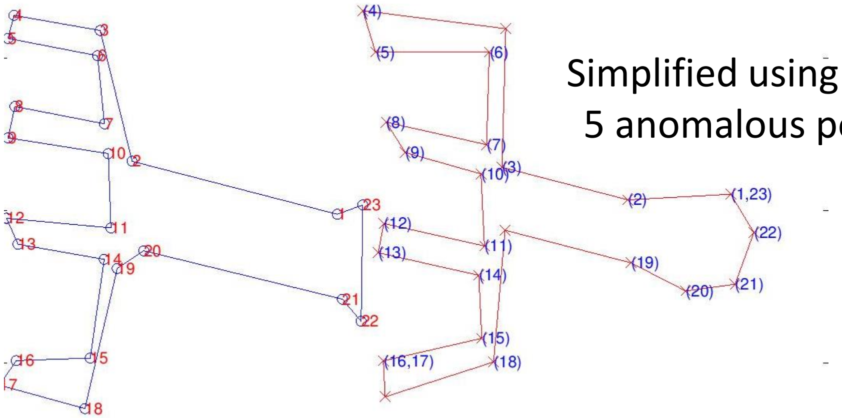
* O. van Kaick, G. Hamarneh, H. Zhang and P. Wighton, "Contour Correspondence via Ant Colony Optimization," *15th Pacific Conference on Computer Graphics and Applications (PG'07)*, 2007, pp. 271-280.

Experimental Results

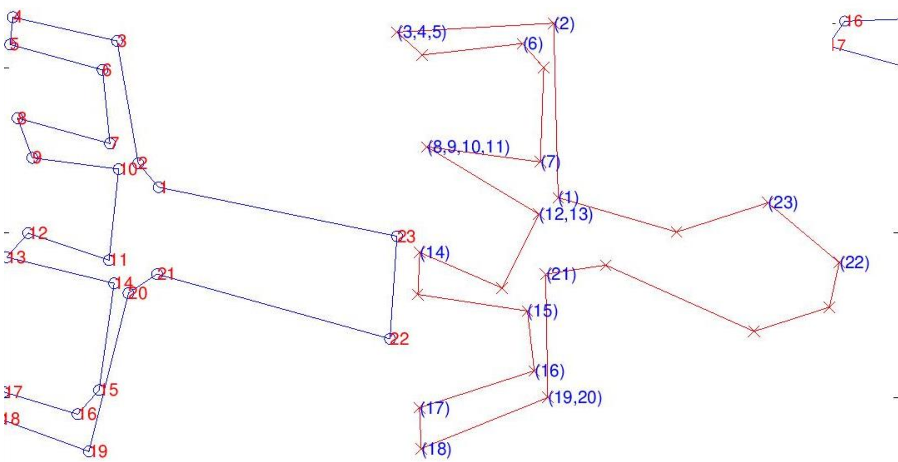
- **Simplification and Matching (216 Binary Shape Database)**



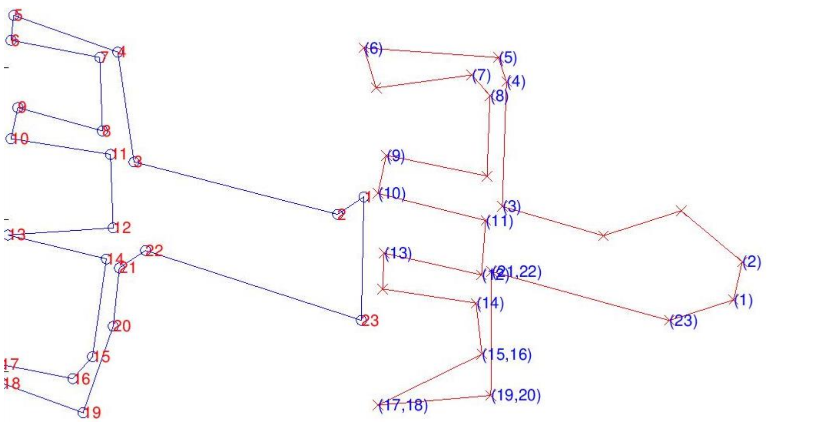
38 points in the first fork
23 points in the second fork



Simplified using MAS:
5 anomalous points



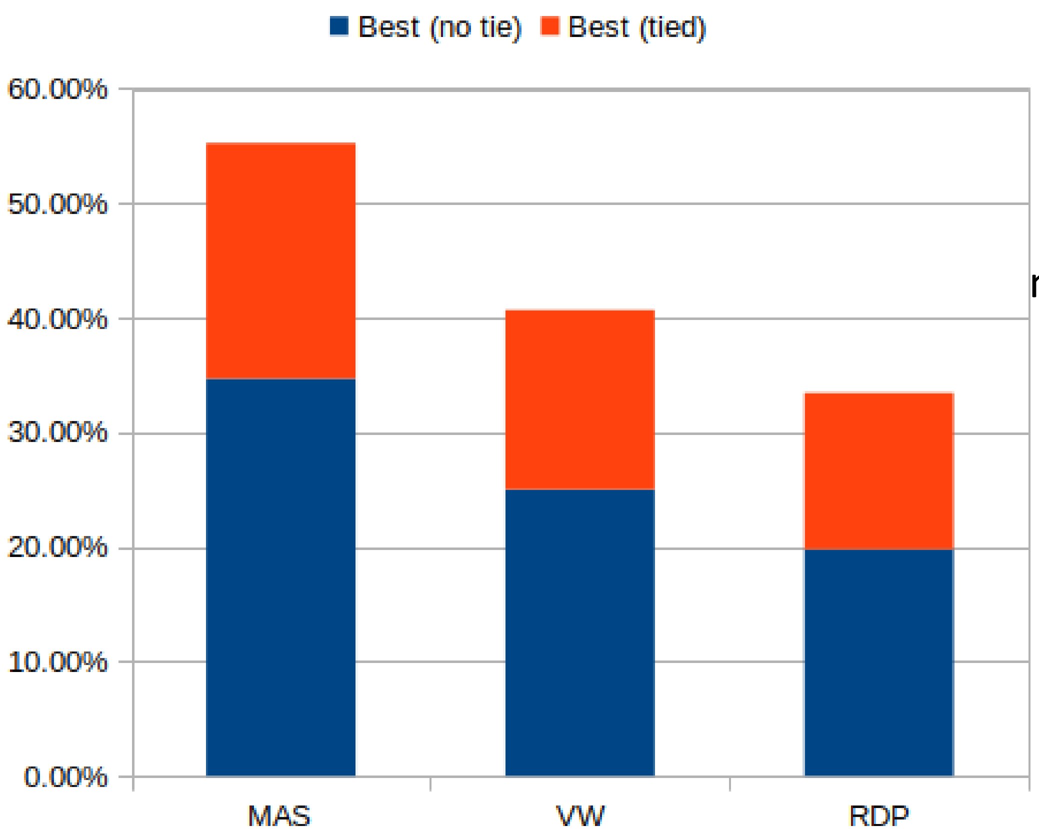
Simplified using Visvalingam-Whyatt algorithm:
12 anomalous points



Simplified using Douglas–Peucker algorithm:
9 anomalous points

Experimental Results

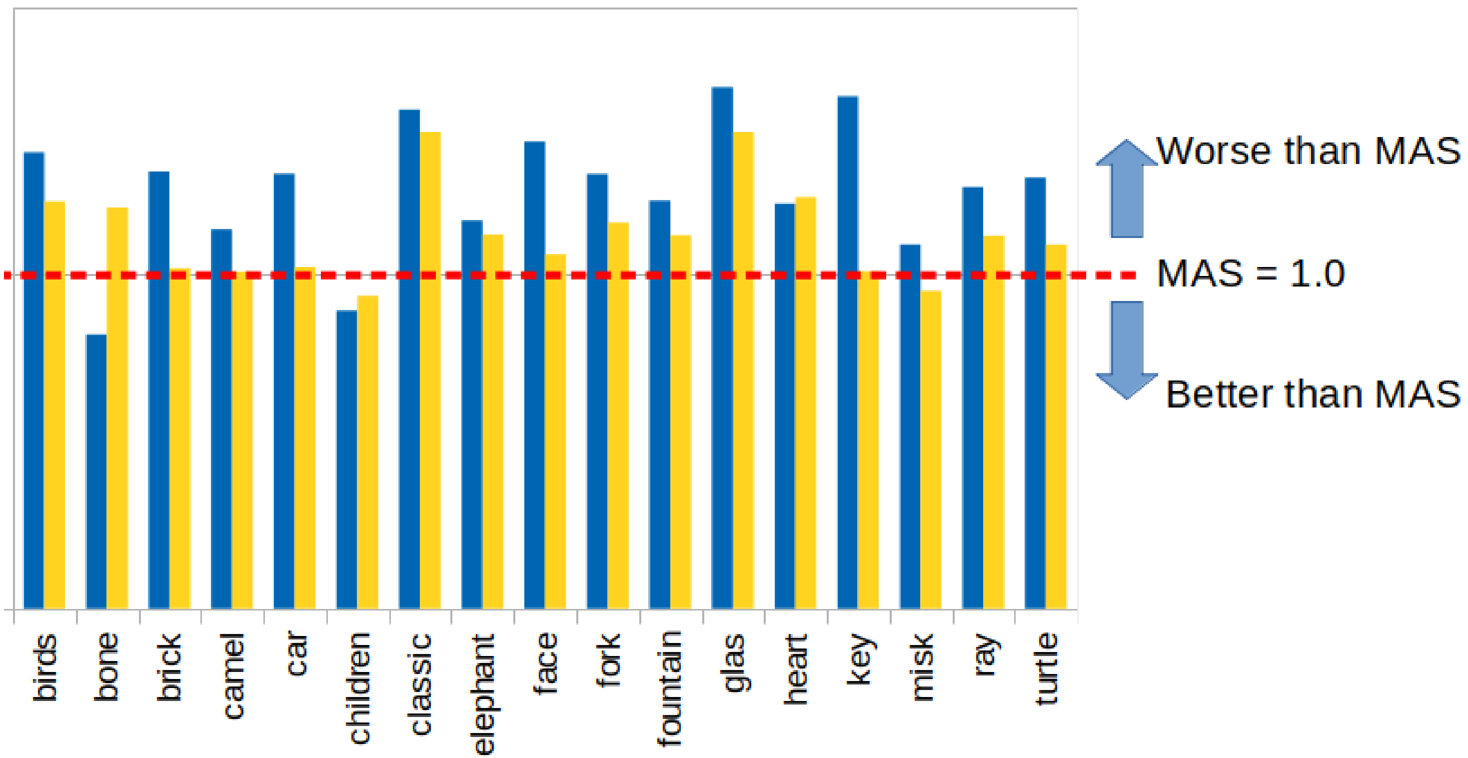
- **Simplification and Matching (216 Binary Shape Database)**



Best is the one that leads to a matching with the lowest number of anomalous points

- **Simplification and Matching (216 Binary Shape Database)**

Gap of performance per class (number of *anomalous* points):
MAS has small defeats and big victories



Conclusions and Future Work

- Matching-Aware Simplification
  - reduces the number of anomalous points during matching
  - can be combined with any matching algorithm
- Future work
  - Study the combined execution of simplification and matching
  - Study the simplification/matching in a wide sequence of polygons (not just two) to generate more natural interpolations
- New datasets (case studies) for moving regions are welcome!

Thank you!

Questions?

Project homepage: http://most.web.ua.pt/