Project Presentation

Gabriel Beauplet

Friday November 2

Introduction

The ScanAllFish project

- Creating a freely available database of scans of all the world's 30,000 fish species
  - Problem 1 : Micro-CT devices are expensive
  - Problem 2 : One scan may last up to 12h
- Solution : Scan several fishes simultaneously

TopoAngler

- Utilizing topological analysis to create selectable candidate features
- Providing an intuitive user-interface
- The ability to export all selected meta-features for further analysis

Level set topology

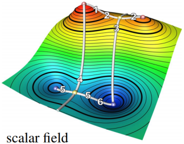
The super level set of a real value a is $\{x \in \mathbb{R}^3 | f(x) \geq a\}$

Join tree

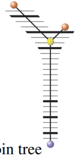
We sweep the 3D space by decreasing the function value f
For each vertex encountered, the topological change caraterize the vertex, it can be :

- regular : The topology of the super-level sets do not change.
- maximum: A new super-level set component is created.
- join saddle: Two super-level set components merging.

A vertex that is not regular is called critical.



scalar field


join tree

Topology-based Segmentation

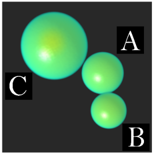
Upper link

For a vertex v, $\{u \in N(v) | f(u) \geq f(v)\}$

Computing the augmented join tree

The algorithm first sorts the vertices of K by decreasing function value

1. If the upper link of v is empty, then create a new component containing v and set v as its head
2. If the upper link of v is not empty, find the components that contain the vertices in the upper link of v. Add an edge between v and the head of each of the components. Next, merge these components and set v as the head of the merged component.



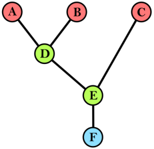


Figure: Left : A level set at a given real value is corresponds to the shown spherical surfaces. A super-level set corresponds to the region inside the three spheres. Right : The join tree of the scalar function

Hierarchical Segmentation

Principle

- Two types of importance measure
  - $|f(c_1) - f(c_2)|$ where (c_1, c_2) is an edge
  - Edge Hypervolume (the integral of the scalar function over the enclosed volume)
- All leaf edges that are incident on a maximum, are first added to a priority queue based on the importance measure
- Then, at each step, the least important leaf edge is removed

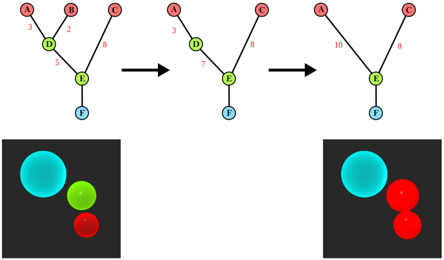


Figure: A hierarchical segmentation of the volume is performed using the join tree. By increasing the simplification level (top), different branches of the tree are joined to form a simplified represtation. This corresponsd to the number of features that are shown in the rendering (bottom).

Hierarchical Segmentation

Computing hierarchical segmentation

- The user select a number n of candidate features
- Only the n most important features are kept, the other are removed

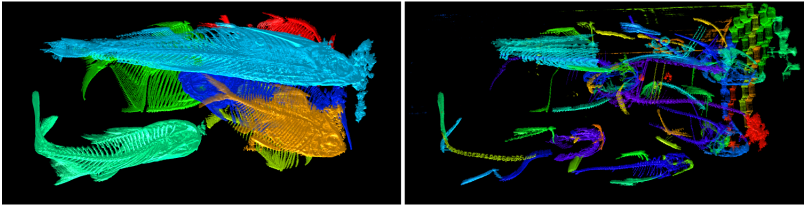


Figure: Selecting a different number of candidate features leads to a change in granularity of detected features. Changing between different simplifications is required by the workflow in order to construct entire fishes with the necessary detail.

My implementation

Make the dataset

- Collect the data : Took 3 fishes on the ScanAllFish project webpage
  - Each fish is contains around 1 billion points
- Downsample each dimension by 8
- Stack the 3 fishes

The algorithm

- Implementation of the H.S. using Python and Cython
  - Compute the join tree
  - Delete all the maximum until n are left

After the algorithm

- Upsample each volume segmented
- Extract the minimum volume containing each volume segmented

Results

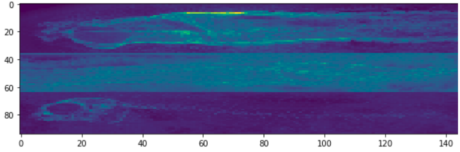


Figure: Superposition of 3 fishes

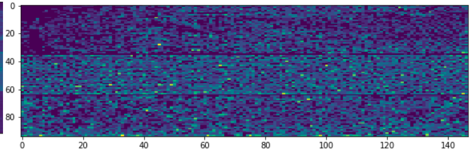


Figure: Separation in 9648 parts

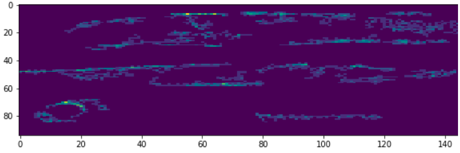


Figure: Separation in 35 parts

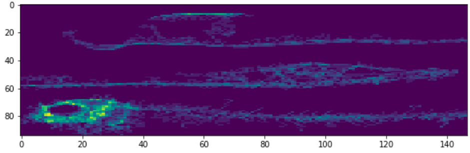


Figure: Separation in 3 parts

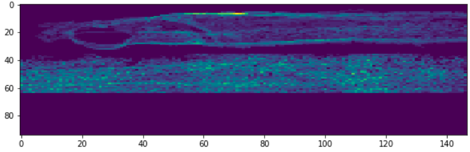


Figure: Separation in 2 parts

Comparison of the importance measure

- Using the sum as importance measure gives better segmentation

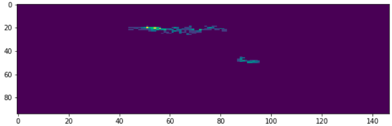


Figure: Separation in 3 parts using the absolute difference between the maximum values of a maximum component and the maximum value of his parent

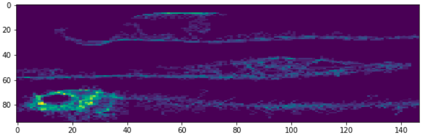


Figure: Separation in 3 parts using the sum of each maximum component

Encountered problems

Quotes from the TopoAngler paper

- *If the upper link of v is not empty, find the components that contain the vertices in the upper link of v*
- *A consistent comparison between vertices is ensured by a simulated perturbation of the function*

Marching Cubes

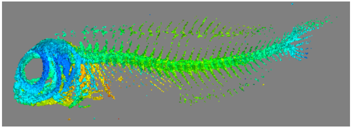


Figure: Mesh of the extracted fish using the marching cube algorithm

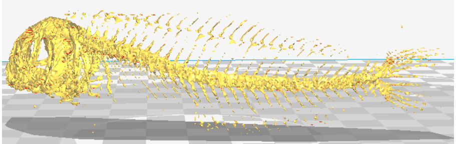


Figure: Mesh of the fish before adding support

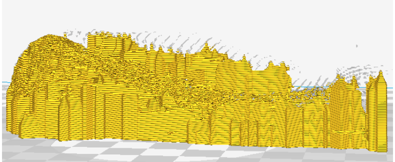


Figure: Mesh of the fish after adding support

Animation

Moving the fin

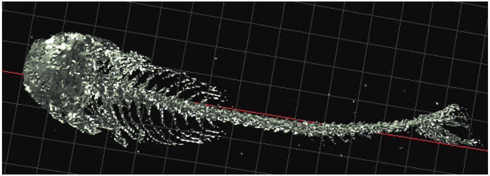


Figure: Image of the fish from top after the extraction

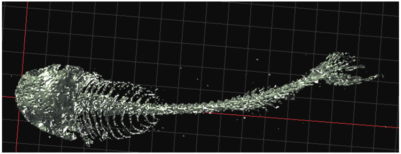


Figure: Start position of the fin using the formula for each point p : $x(p) = 5.10^{-4} z(p)^2$

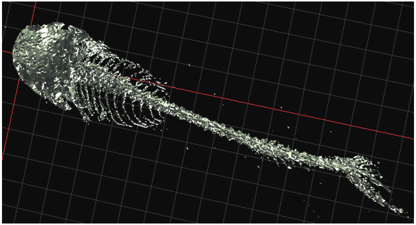


Figure: End position of the fin using the formula for each point p : $x(p) = -7.10^{-4} z(p)^2$

Animation

Bad moving of the fin

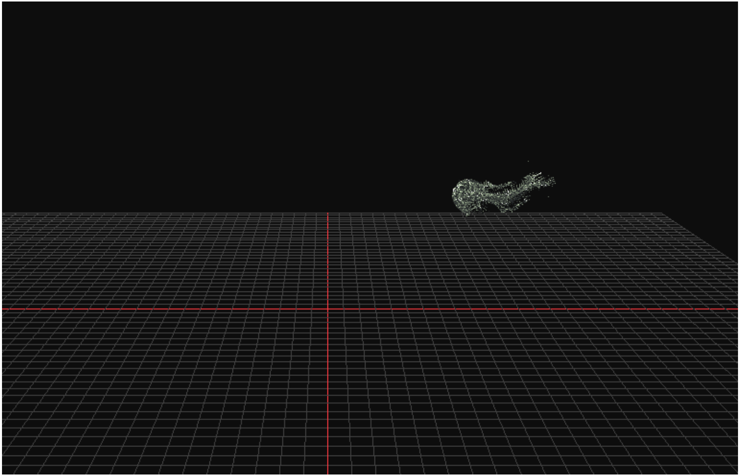


Figure: Example of random swim

Fish Game

What could we do ?

- Move the fish randomly in the 3D space
- A game : The fish Game (analogy with the snake game)
  - Move the fish with the left and right arrow keys
  - Eat the food to get a bigger fish
  - The game stops if you touch the edges with the head of the fish

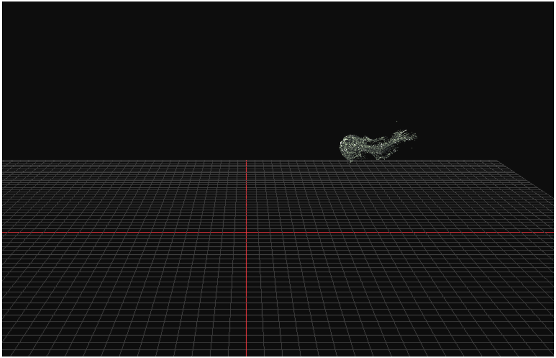


Figure: Example of random swim

Conclusion

Improvement

- Try to improve the hierarchical segmentation with machine learning
- Make the fish game with 3 degrees of liberty
- Make the fish open his mouth
- Add other fishes

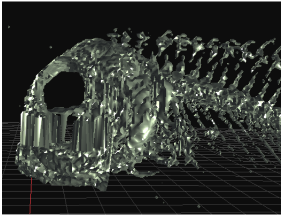


Figure: Result when you lower one part of the fish head

All the code is available on my github page : https://github.com/beaupletga