Mathematical Methods for Image Synthesis - Reading/Project 2

Nicolas Bonneel

In this class you will need to do one project and one article reading that will be presented at the end of the semester. If you take a project with me, you will need to take a reading from Julie's part, and conversely, if you take the reading with me, you'll need to take a project from Julie. A project is expected to take between 15 and 30 hours, and can be implemented with any *imperative* programming language you want. Readings go much more in-depth than the class lectures, and are thus more complex.

1 Readings – weeks 3-4

You can choose between these 3 options :

- Polar factorization and monotone rearrangement of vectorâĂŘvalued functions, Brenier 1991
  `http://www.math.toronto.edu/~mccann/assignments/477/Brenier91.pdf`
- Entropic Metric Alignment for Correspondence Problems, Solomon et al., 2016 `https://hal.archives-ouvertes.fr/hal-01305808`
- Principal Geodesic Analysis for Probability Measures under the Optimal Transport Metric, Seguy et al. 2015
  `http://papers.nips.cc/paper/5680-principal-geodesic-analysis-for-probability-measures-under-the-optimal-trans`

2 Project

For this project, we require you to implement the paper of Aurenhammer et al. "Minkowski-Type Theorems and Least-Squares Clustering" `http://sci-hub.cc/10.1007/PL00009187` to compute a displacement interpolation between two images (one is sampled with diracs, the other one is consi-

dered as a continuous function). You may get help from MÃľrigot's paper "A Multiscale Approach to Optimal Transport" `https://hal.inria.fr/hal-00604684/document` : its writing is more concise and hence clearer, and it also explains the image part, but we do not require you to implement the multiscale aspect.

You may use an existing Voronoi diagram library for your chosen language. In C++, you may use CGAL or other more lightweight libraries for this task, or even use a convex hull library (e.g., QHull) and compute the Voronoi diagram via the parabolic lifting seen in class. In Matlab, you may use the function "voronoi" directly ; in Python you may use SciPy. Any other choice of language or library for Voronoi diagrams in permitted. You may not directly use a Power diagram library.

You will then use the lifting seen in class to transform your Voronoi diagram into a Power diagram. As such, be sure to use a Voronoi library that allows for 3-D Voronoi diagrams (as you will lift 2d points in the image plane to 3d), or 4-d convex hulls (one lifting to go from the convex hull to Voronoi diagram, and another lifting to go from the Voronoi diagram to a Power diagram).

You will then implement the gradient descent, and apply it to images. You will sample the target measure (i.e., the sum of diracs) using Lloyd algorithm on the target image. You will obtain displacement interpolation results similarly to MÃľrigot's approach.

Note : We are fully aware a C++ implementation can be found on Q. MÃľrigot's webpage. Of course, plagiarism is not permitted.