Assignment Five: Geometric Optimal Transport Map

David Gu

Yau Mathematics Science Center
Tsinghua University
Computer Science Department
Stony Brook University

gu@cs.stonybrook.edu

August 14, 2020

Computational Results


Computational Results

Computational Results

How to eliminate mode collapse?

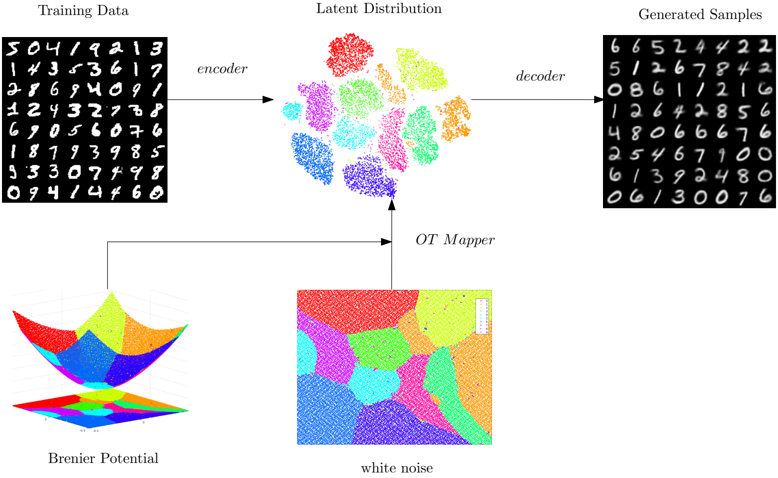


Figure: Geometric Generative Model.

Convex Geometric View

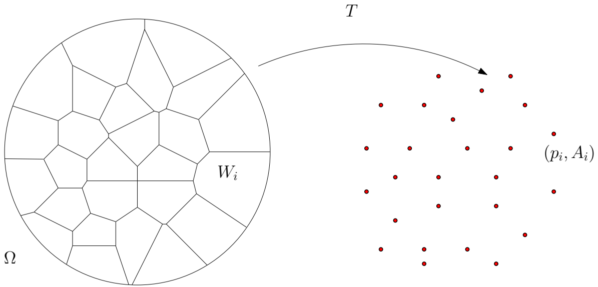
Monge-Ampére Equation

Problem (Brenier)

Given (Ω, μ) *and* (Σ, ν) *and the cost function* $c(x, y) = \frac{1}{2}|x - y|^2$, *the optimal transportation map* $T : \Omega \to \Sigma$ *is the gradient map of the Brenier potential* $u : \Omega \to \mathbb{R}$, *which satisfies the Monge-Ampére equation,*

$$\boxed{det \left(\frac{\partial^2 u(x)}{\partial x_i \partial x_j} \right) = \frac{f(x)}{g \circ \nabla u(x)}}$$

Semi-Discrete Optimal Transportation Problem

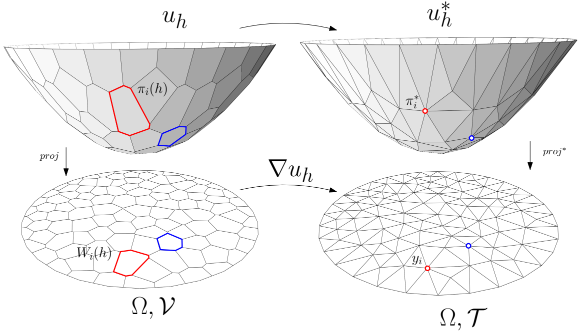


Problem (Semi-discrete OT)

Given a compact convex domain Ω *in* $\mathbb{R}^d$, *and* $p_1, p_2, \cdots, p_k$ *and weights* $w_1, w_2, \cdots, w_k > 0$, *find a transport map* $T : \Omega \to \{p_1, \dots, p_k\}$, *such that* $vol(T^{-1}(p_i)) = w_i$, *so that* T *minimizes the transportation cost:*

$$\mathcal{C}(T) := \frac{1}{2} \int_{\Omega} |x - T(x)|^2 dx$$

Semi-Discrete Optimal Transportation Problem



According to Brenier theorem, there will be a piecewise linear convex function $u : \Omega \to \mathbb{R}$, the gradient map gives the optimal transport map.

Alexandrov Theorem

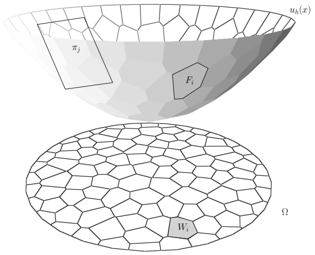
Theorem (Alexandrov 1950)

Given Ω compact convex domain in $\mathbb{R}^n$, $p_1, \cdots, p_k$ distinct in $\mathbb{R}^n$, $A_1, \cdots, A_k > 0$, such that $\sum A_i = Vol(\Omega)$, there exists PL convex function

$$f(\mathbf{x}) := \max\{\langle \mathbf{x}, \mathbf{p}_i \rangle + h_i | i = 1, \cdots, k\}$$

unique up to translation such that

$$Vol(W_i) = Vol(\{\mathbf{x} | \nabla f(\mathbf{x}) = \mathbf{p}_i\}) = A_i.$$



Alexandrov's proof is topological, not variational. It has been open for years to find a constructive proof.

Variational Proof

Theorem (Gu-Luo-Sun-Yau 2013)

Ω is a compact convex domain in $\mathbb{R}^n$, $y_1, \cdots, y_k$ distinct in $\mathbb{R}^n$, μ a positive continuous measure on Ω. For any $\nu_1, \cdots, \nu_k > 0$ with $\sum \nu_i = \mu(\Omega)$, there exists a vector $(h_1, \cdots, h_k)$ so that

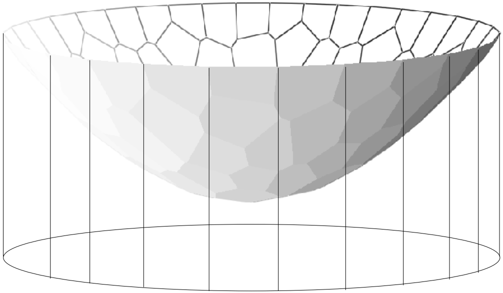
$$u(\mathbf{x}) = \max\{\langle \mathbf{x}, \mathbf{p}_i \rangle + h_i\}$$

satisfies $\mu(W_i \cap \Omega) = \nu_i$, where $W_i = \{\mathbf{x} | \nabla f(\mathbf{x}) = \mathbf{p}_i\}$. Furthermore, $\mathbf{h}$ is the maximum point of the concave function

$$E(\mathbf{h}) = \sum_{i=1}^{k} \nu_i h_i - \int_{\mathbf{0}}^{\mathbf{h}} \sum_{i=1}^{k} w_i(\eta) d\eta_i,$$

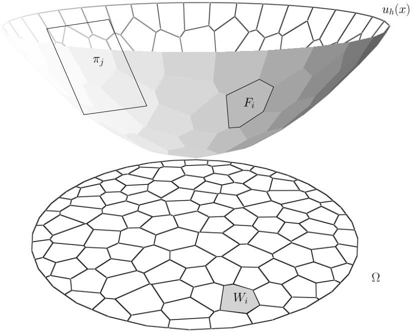
where $w_i(\eta) = \mu(W_i(\eta) \cap \Omega)$ is the μ-volume of the cell.

Geometric Interpretation



One can define a cylinder through $\partial\Omega$, the cylinder is truncated by the xy-plane and the convex polyhedron. The energy term $\int^{\mathbf{h}} \sum w_i(\eta) d\eta_i$ equals to the volume of the truncated cylinder.

Computational Algorithm

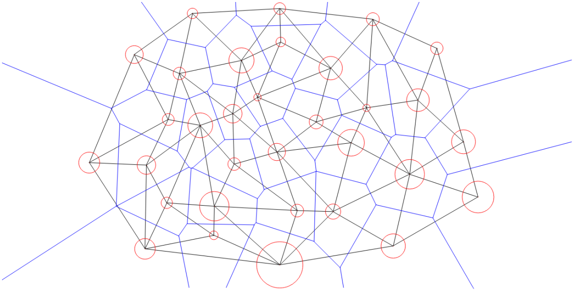


Definition (Alexandrov Potential)

The concave energy is

$$E(h_1, h_2, \cdots, h_k) = \sum_{i=1}^{k} \nu_i h_i - \int_{\mathbf{0}}^{\mathbf{h}} \sum_{j=1}^{k} w_j(\eta) d\eta_j,$$

Computational Algorithm



The Hessian of the energy is the length ratios of edge and dual edges,

$$\frac{\partial w_i}{\partial h_j} = -\frac{|e_{ij}|}{|\bar{e}_{ij}|}$$

Optimal Transport Map

Input: A set of distinct points $P = \{p_1, p_2, \cdots, p_k\}$, and the weights $\{A_1, A_2, \cdots, A_k\}$;A convex domain Ω, $\sum A_j = \mathrm{Vol}(\Omega)$;
Output: The optimal transport map $T : \Omega \to P$

1. Scale and translate P, such that $P \subset \Omega$;
2. Initialize $\mathbf{h}^0 \leftarrow \frac{1}{2}(|p_1|^2, |p_2|^2, \cdots, |p_k|^2)^T$;
3. Compute the Brenier potential $u(\mathbf{h}^k)$ (envelope of π_i's) and its Legendre dual $u^*(\mathbf{h}^k)$ (convex hull of π_i^*'s);
4. Project the Brenier potential and Legendre dual to obtain weighted Delaunay triangulation $\mathcal{T}(\mathbf{h}^k)$ and power diagram $\mathcal{D}(\mathbf{h}^k)$;

5. Compute the gradient of the energy

$$\nabla E(\mathbf{h}) = (A_1 - w_1(\mathbf{h}), A_2 - w_2(\mathbf{h}), \cdots, A_k - w_k(\mathbf{h}))^T.$$

6. If $\|E(\mathbf{h}^k)\|$ is less than ε, then return $T = \nabla u(\mathbf{h}^k)$;
7. Compute the Hessian matrix of the energy

$$\frac{\partial w_i(\mathbf{h})}{\partial h_j} = -\frac{|e_{ij}|}{|\bar{e}_{ij}|}, \quad \frac{\partial w_i}{\partial h_i} = -\sum_j \frac{\partial w_i(\mathbf{h})}{\partial h_j}.$$

8. Solve linear system

$$\nabla E(\mathbf{h}) = \mathrm{Hess}(\mathbf{h}^k)\mathbf{d};$$

Optimal Transport Map

9. Set the step length $\lambda \leftarrow 1$;
10. Construct the convex hull $\mathrm{Conv}(\mathbf{h}^k + \lambda \mathbf{d})$;
11. if there is any empty power cell, $\lambda \leftarrow \frac{1}{2}\lambda$, repeat step 3 and 4, until all power cells are non-empty;
12. set $\mathbf{h}^{k+1} \leftarrow \mathbf{h}^k + \lambda \mathbf{d}$;
13. Repeat step 3 through 14.

Optimal Transportation Map

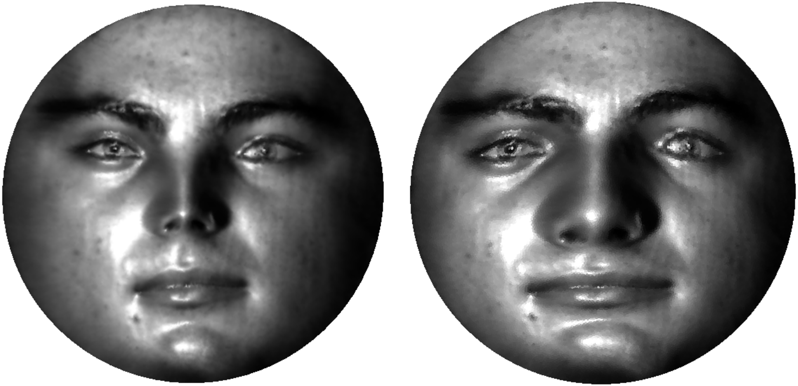


Figure: Optimal transportation map.

Optimal Transportation Map

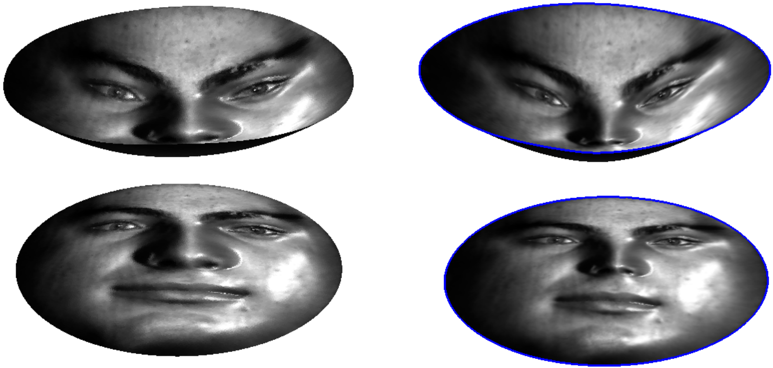


Figure: Optimal transportation map.

Instruction

Dependencies

1. 'detri2', a mesh generation library, written by Dr. Hang Si.
2. ‘MeshLib‘, a mesh library based on halfedge data structure.
3. ‘freeglut‘, a free-software/open-source alternative to the OpenGL Utility Toolkit (GLUT) library.

Directory Structure

- ot_2d/include, the header files for optimal transport;
- ot_2d/src, the source files for optimal transport.
- data,Some models.
- CMakeLists.txt, CMake configuration file.
- resources, Some resources needed.
- 3rdparty, MeshLib and freeglut libraries.

Configuration

Before you start, read README.md carefully, then go three the following procedures, step by step.

1. Install [CMake](https://cmake.org/download/).
2. Download the source code of the C++ framework.
3. Configure and generate the project for Visual Studio.
4. Open the .sln using Visual Studio, and complie the solution.
5. Finish your code in your IDE.
6. Run the executable program.

3. Configure and generate the project

1. open a command window
2. cd Assignment_6_skeleton
3. mkdir build
4. cd build
5. cmake ..
6. open CCGHomework.sln inside the build directory.

5. Finish your code in your IDE

- You need to modify the file: OT.cpp and CDomainOptimalTransport.cpp
- search for comments "insert your code here"
- Modify functions:
  1. CDomainTransport::_newton(COMTMesh * pInput, COMTMesh * pOutput)
  2. CBaseOT::_update_direction(COMTMesh* pMesh)
  3. CBaseOT::_compute_hessian_matrix(COMTMesh& mesh,Eigen::SparseMatrix& hessian)

6. Run the executable program

Dynamic Linking Libraries

Copy detri2.dll and detri2d.dll from 3rdparty/detri2/lib/windows to build/ot_2d/; Libraries and dlls for Linux and MAC are also available.

Command

Command line:

OT2d.exe girl.m

All the data files are in the data folder, all the texture images are in the textures folder.