Learning Low-dimensional Representations of Shape Data Sets

Project of Approches Géométriques en Apprentissage Statistique

Tong ZHAO

Overview

Introduction

Algorithm

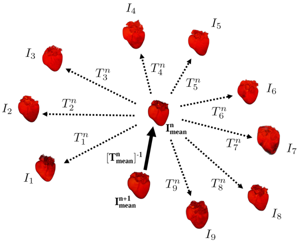
Experiment

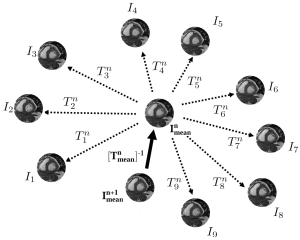
Conclusion

Task

Given a set of shapes, how to find the average?

Given the mean shape, how to find the diffeomorphism from the mean to each shape?





Fréchet Mean

$$m = \text{argmin}_{p \in C} \sum_{i=1}^{N} \int_M d(p, x_i)^2 dV$$

where:

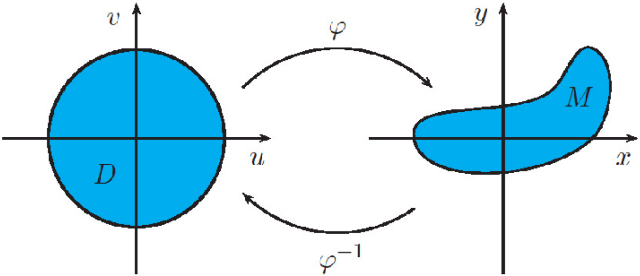
- C is the space containing all shapes
- M is the space where a shape lies in
- N is the number of shapes in the set
- d is a defined metric on C
- $x_1, \cdots, x_N \in$ C

Related Methods - Fréchet mean

- **Point distribution model** (PDM): linear model (Euclidean space + Euclidean metric)
- **Kernel-based Method**: choice of kernel is not trivial (Euclidean space + Non-linear metric)
- **Connectivity-Independent Method**: large non-convex optimization (Euclidean space + Non-linear metric)
- **Lie-Algebra Based Method**: SE(3) (Non-Euclidean space)
- **Deformable Model**: large non-convex optimization

Diffeomorphism

Given two manifolds D and M, a differental map φ is called a diffeomorphism if it is a bijection and its inverse is differentiable as well.

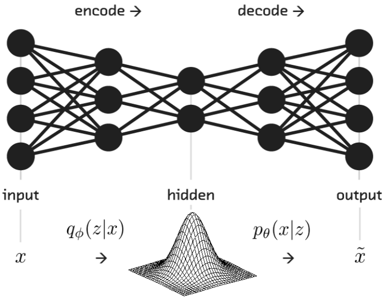


Related Methods - Diffeomorphism

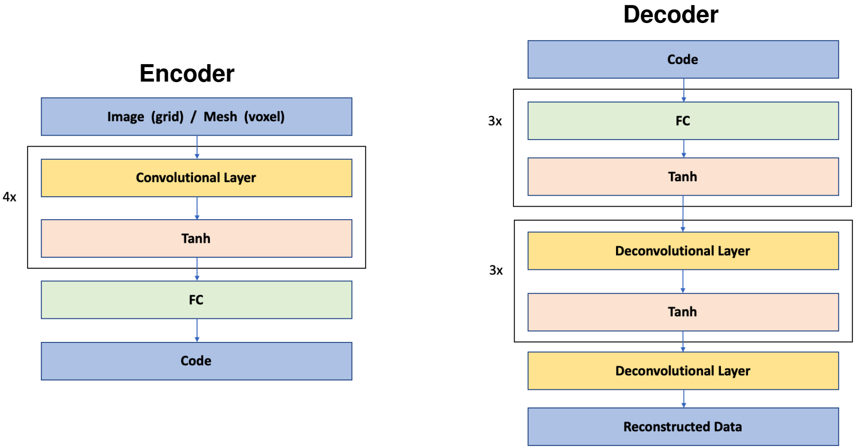
- **Rigid Transformation**: few degrees of liberty
- **Optimal Transport**: don't have any smoothness guarentee
  - **SVF**: stationary velocity fields
  - **LDDMM**: dynamic velocity fields
- <u>**Deep Learning Method**</u>

Basic Idea

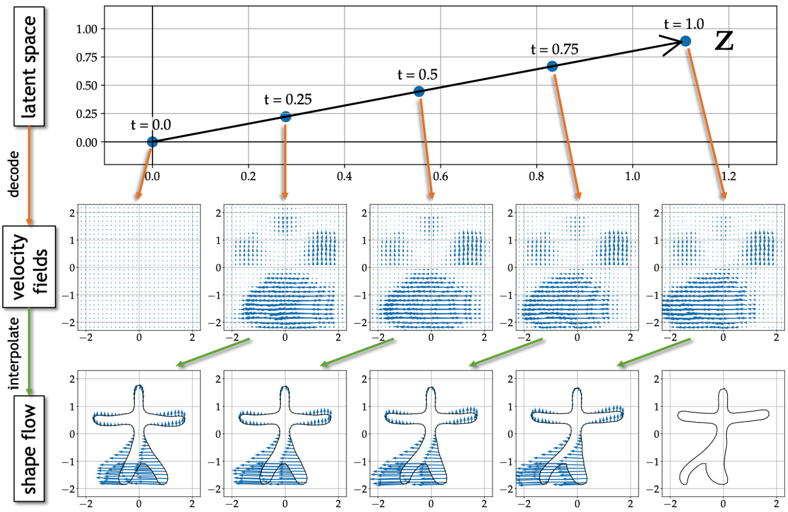
An LDDMM-based deep learning method to construct diffeomorphism.



Net Structure - VAE



Diffeomorphism



Reprinted from the article

Loss Function

- Cross entropy: estimated by Monte-Carlo Method

$$\mathcal{A}(y_i;\theta,\nu,y_0) = -\int \log p(y_i|z_i;\theta,y_0)\cdot q(z_i|y_i;\nu)\cdot dz_i$$
$$\approx -\frac{1}{L}\sum_{i=1}^{L}\log p(y_i|z_i^{(l)};\theta,y_0) = \mathcal{A}'$$

- KL divergence

$$\mathcal{R}_{kl}(y_i;\theta,\nu) = KL[q(z_i|y_i;\nu)||p(z_i)]$$

- S-Sobolev norm

$$\mathcal{R}_s(\theta,\nu;y_i) = \lambda\cdot\int_{t\in[0,1]}\int_{z_i\in\mathbb{R}^n}\|D_\theta(z_i\cdot t)\|_S^2\cdot q(z_i|y_i;\nu)\cdot dz_i\cdot dt$$
$$\approx \frac{\lambda}{T\cdot L}\sum_{t=1}^{T}\sum_{l=1}^{L}\|D(z_i^{(l)}\cdot\frac{t-1}{T-1})\|_S^2 = \mathcal{R}_s'(\theta,\nu;y_i)$$

Optimization

While the network is optimized by the framework directly, the mean shape y_0 is optimized by a smoothed gradient.

The gradient $\frac{\partial L'}{\partial y_0}$ is smoothed by a gaussien kernel of standard deviation σ_y, then it is backpropogated as usual.

Current-splatting Layer

Given a mesh with K triangles whose centers are $\{c_1, \cdots, c_K\}$ and whose normals are $\{n_1, \cdots, n_K\}$, we have for $\forall x \in \mathbb{R}^3$:

$$S_{y_i}(x) = \sum_{i=1}^{K} \exp\left[- \|x - c_k\|^2 / \sigma_S^2 \right] \cdot n_k$$

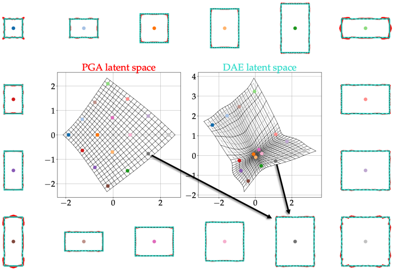
Test 1: Learn Latent Space with Rectangles

Dataset: Simulated rectangle meshes with point correspondence.

Number: training meshes 441, testing meshes 400.

Model:

- A PGA model with 2 components
- A DAE model initialized by PGA results (?)



Reprinted from the article

Test 2: Hippocampi Meshes

Dataset: Right hippocampi meshes segmented from MR images in ADNI dataset

Number: training meshes 162, testing meshes 162.

Model:

- A PGA model with 10 components (?)
- A DAE model with 10 latent variables

	Data noise	PGA	DAE	DAE+
Reconstruction	85.2 ± 40.1	66.7 ± 11.5	$\mathbf{32.6} \pm \mathbf{6.0}$	-
Generalization	85.2 ± 40.1	$\mathbf{67.7} \pm \mathbf{12.6}$	116.8 ± 20.0	74.7 ± 16.1

Reprinted from the article

Test 3: Classification with MR Images

Dataset: Brain T1-wieghted MR images from ADNI dataset

Number: 54 - CN, 53 - MCI, 53 - AD

Model:

- A PGA model with 3 components
- A DAE model with 3 latent variables

	CN/MCI/AD	CN/AD	CN/MCI	MCI/AD
PGA	58.8 %	84.1 %	67.3 %	**71.7 %**
DAE	**61.3 %**	**85.0 %**	67.3 %	68.9 %

Reprinted from the article

Advantage

- **An end-to-end framework** which learns at the same time the mean shape y_0 and the diffeomorphism.
- It can be used to **generate new data**, which helps us to enlarge the dataset
- **A novel idea** to apply deep learning models in diffeomorphism
- The method can be applied both with or **without point correspondence**.

Disadvantage

- **Overfitting issue**: The DAE can be esaily overfitted by a small amount of data, which leads to a large generalization error.
- **Regularization term for meshes**: Given that the results from DAE have always a sharp curvature, we may add a regularization term (eg. laplacian loss) to get a smoother result. It also helps to reduce noises.
- **Re-parameterization**: facilitate the calculation of KL divergence and help to sample new data.
- **Ensure the differentiability**: as a regularization term, it is hard to now if the differentiability of the mapping is ensured or not.