Deep Manifold Prior

UMASS AMHERST

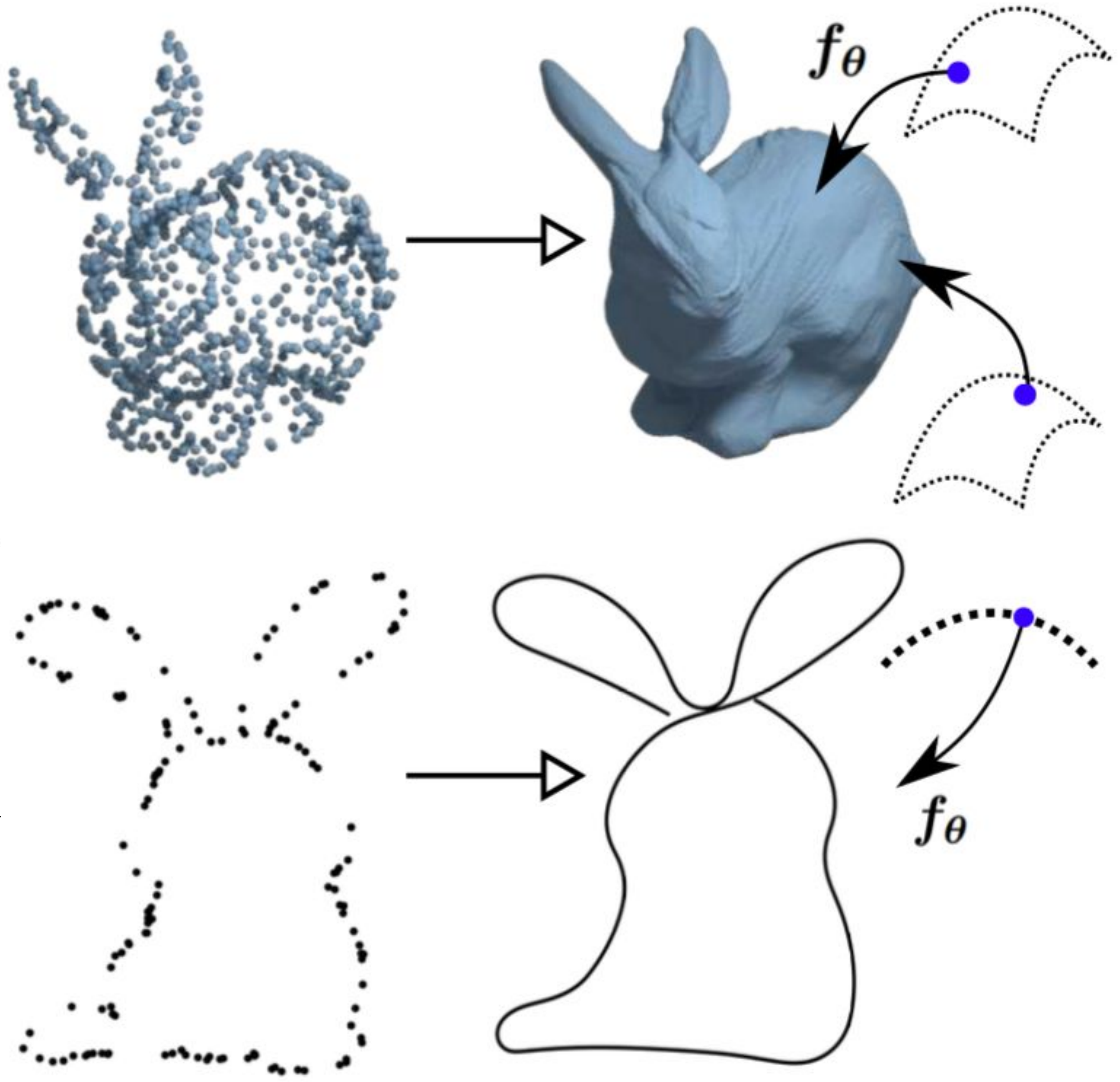
Matheus Gadelha
mgadelha@cs.umass.edu

Rui Wang
rui.wang@cs.umass.edu

Subhransu Maji
smaji@cs.umass.edu

mgadelha.me/dmp

Scan for page and code!



We present a prior for manifold structured data, such as surfaces of 3D shapes, where deep neural networks are adopted to reconstruct a target shape using gradient descent starting from a random initialization. We show that surfaces generated this way are smooth, with limiting behavior characterized by Gaussian processes.

Overview

A n-manifold is a topological space $\mathcal{M}$ for which every point in $\mathcal{M}$ has a neighborhood homeomorphic to the Euclidean space $\mathbb{R}^n$

$$\mathcal{U} \subset \mathcal{M} \qquad \mathcal{V} \subset \mathbb{R}^n$$

Point cloud: $P \subset \mathcal{M}$

n-manifold chart:

$$\phi : \mathcal{U} \rightarrow \mathcal{V}, \phi(u) = (x_1(u), x_2(u), ..., x_n(u))$$

Manifold parametrization:

$$f_\theta = \zeta = \phi^{-1}$$

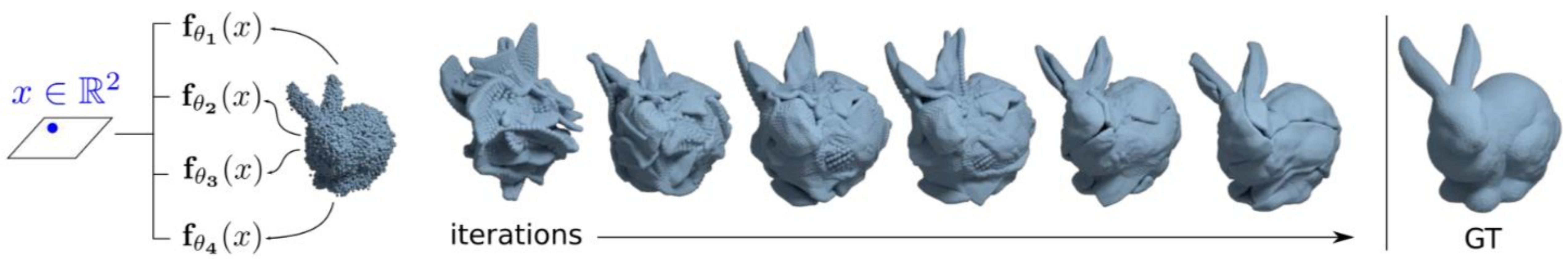
neural network

Chamfer distance between point clouds P_1 and P_2:

$$\mathcal{L}_C(P_1, P_2) = \sum_{p_1 \in P_1} \min_{p_2 \in P_2} \|p_1 - p_2\|_2^2 + \sum_{p_2 \in P_2} \min_{p_1 \in P_1} \|p_1 - p_2\|_2^2$$

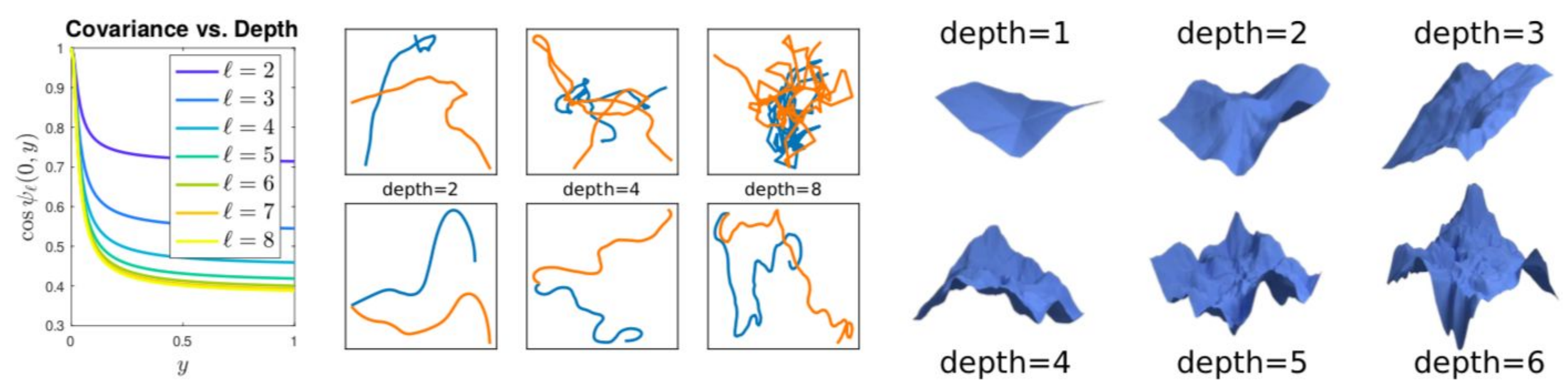
Optimize parametrizations through gradient descent:

$$\theta_1^*, \theta_2^*, ...\theta_k^* = \underset{\theta_1, \theta_2, ...\theta_k}{\arg\min}\, \mathcal{L}_C\Big(\bigcup_{i=1}^{k} f_{\theta_i}(x), P\Big)$$



Manifold reconstruction pipeline. Manifold parametrizations are encoded by neural networks and trained to minimize the reconstruction error wrt. the noisy target. Prior induced by the neural networks makes the generated surface much closer to the ground-truth, *without ever seeing any additional training data*.

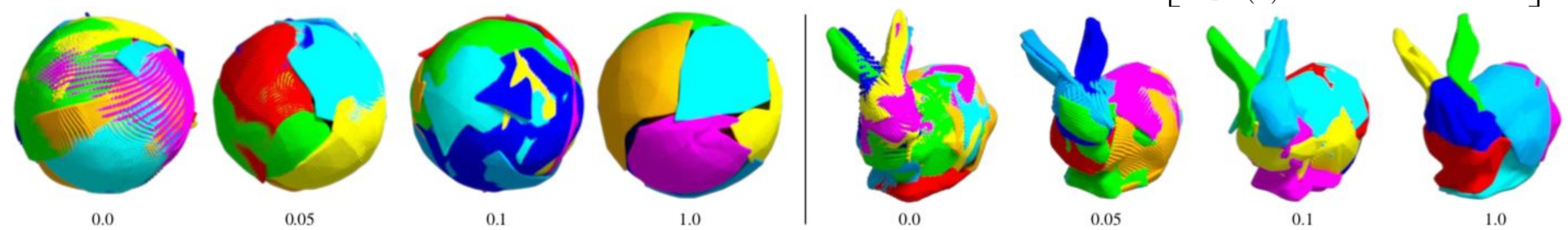
Limiting GP for the Deep Manifold Prior



(left) a plot demonstrating the relationship between the network depth and the covariance function for the limiting GP. **(middle)** Random curves generated by the coordinate (top rows) and arc-length (bottom rows) parametrizations using deep networks with varying depths. **(right)** Random surfaces generated by deep networks of varying depths.

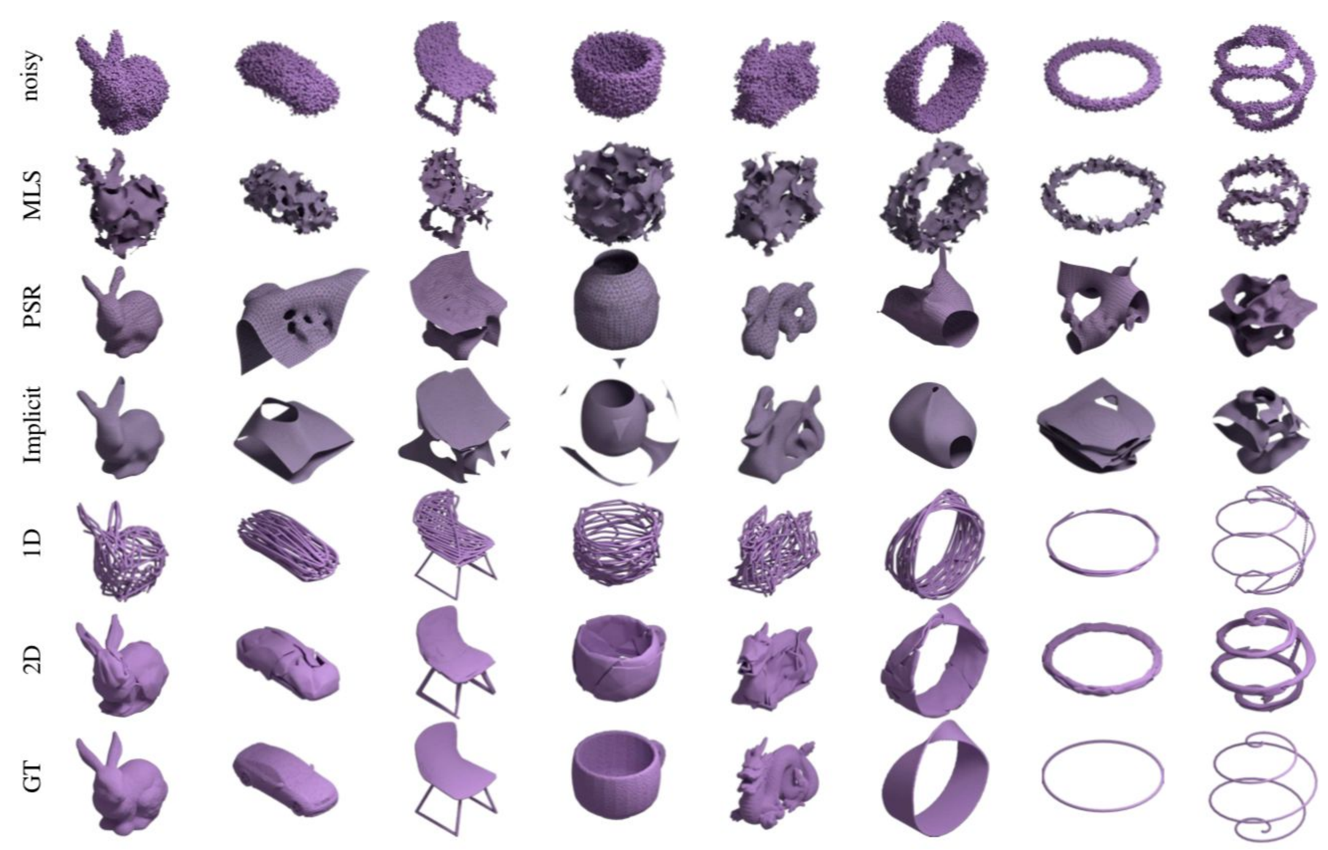
Stretch Regularization

$$\mathcal{L}_S(\theta) = \mathbb{E}_{x \sim [0,1]^n} \left[\sum_{x' \in \mathcal{N}(x)} \|f_\theta(x) - f_\theta(x')\|_2^2 \right]$$



Effect of the regularization weight on the reconstructed manifold. For this experiment, we use our method to reconstruct a sphere using an atlas with 8 charts and render each one with a different color.

Denoising



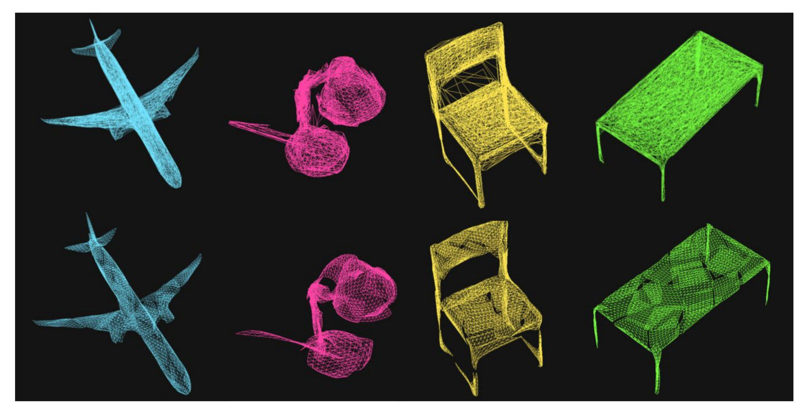
	Surface	Contour	Implicit	RIMLS [23]	SPSR [16]
bunny	**2.71E-04**	6.64E-04	5.52E-04	1.43E-03	3.96E-04
dragon	**4.18E-04**	6.12E-04	1.20E-03	1.65E-03	1.46E-02
car	**2.73E-04**	4.57E-04	6.83E-02	1.50E-03	2.10E-03
cup	**2.59E-04**	5.80E-04	2.64E-02	1.74E-03	1.00E-02
mobius	**3.51E-04**	4.95E-04	3.26E-03	1.96E-03	1.89E-02
chair	**3.95E-04**	4.22E-04	7.32E-03	2.09E-03	2.58E-02
spiral	1.05E-03	**7.31E-04**	1.64E-02	2.98E-03	7.90E-02
ring	5.69E-04	**5.54E-04**	4.81E-02	2.46E-03	3.76E-02
avg.	**4.48E-04**	5.65E-04	2.13E-02	1.98E-03	2.36E-02

Quantitative results for point cloud denoising. Surface, Contour and Implicit represent different deep manifold priors based on a 2-manifold, 1-manifold and level-set paramertizations.

Ablation studies:

	S1R	S8R	S1	S8	C1R	C8R	C1	C8	RIMLS [23]	SPSR [16]
avg.	4.48E-03	**4.48E-04**	2.75E-03	1.35E-03	1.08E-03	5.77E-04	1.00E-03	5.82E-04	1.98E-03	2.36E-02

Learning from Data



Auto-encoder w/ stretch reg. Vanilla AtlasNet(top) trained with stretch reg. (bottom).

Architecture	mean/cat.	mean/inst.	#params.
MRTNet	4.80	4.26	81.6M
AtlasNet	4.74	4.38	42.6M
ConvAtlas	**4.53**	**4.00**	**14.5M**

SVR w/ convolutional parametrizations. Mean Chamfer metric (scaled by 10^3).