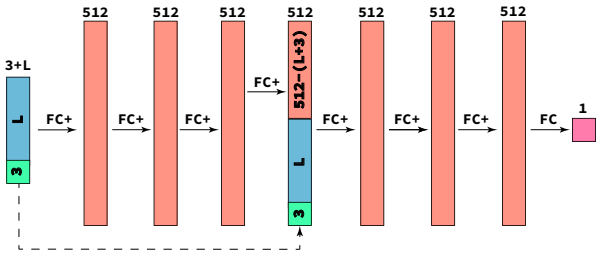
# Implicit Geometric Regularization for Learning Shapes Supplementary Material

## A. Additional Implementation Details

### A.1. Network Architecture.

We used Auto-Decoder network architecture proposed in (Park et al., 2019), as described in sections 5 and 7.2:




where $\mathbf{FC}$ is a fully connected linear layer and $\mathbf{FC}+$ is $\mathbf{FC}$ followed by $\mathbf{softplus}$ activation; a smooth approximation of $\mathbf{ReLU}$: $x \mapsto \frac{1}{\beta}\ln(1+e^{\beta x})$. We used $\beta = 100$. The dashed line connecting the input to the 4th layer indicates a skip connection. $\mathbf{L}$ is the latent vector's size. For shape reconstruction application we take $\mathbf{L} = 0$; for the shape space experiment we used $\mathbf{L} = 256$.

### A.2. Training Details.

**Shape Reconstruction.** Training was done on a single Nvidia V-100 GPU with PYTORCH deep learning framework (Paszke et al., 2017). We used ADAM optimizer (Kingma & Ba, 2014) for 100k iterations with constant learning rate of $0.0001$. In each iteration we sampled uniformly at random $128^2$ points from of the input point cloud.

**Shape Space Learning.** Training was done on 4 Nvidia V-100 GPUs, with PYTORCH deep learning framework (Paszke et al., 2017). We used ADAM optimizer (Kingma & Ba, 2014) for 1k epochs with initial learning rate of $0.0005$ scheduled to decrease by a factor of 2 every $500$ epochs. We divided the training set into mini-batches: a batch contains 32 different shapes, where each shape is freshly sampled uniformly at random to produce $128^2$ points.

## B. Additional Results

### B.1. Shape Space Learning

As mentioned in 7.2 we present additional results from the shape space learning experiment in Figure 2. We provide reconstruction results of both training and test (i.e., unseen point clouds) sets, with the random train-test split. These results are discussed in Section 7.2.

## C. Theory

### C.1. Plane Reproduction using Liapunov Function

In this section we suggest an alternative, self-contained proof for the plane reproduction property of our model in the non-noisy data case, i.e., $\mathcal{X} = \{\boldsymbol{x}_i\}_{i\in I}$ span some $d-1$ dimension hyperplane $\mathcal{H} \subset \mathbb{R}^d$ that contains the origin.

We present a simple argument that, with random initialization, the gradient flow in equation 1 converges, with probability one, to one of the two global minima corresponding to the signed distance function to $\mathcal{H}$ characterized in Theorem 1. We work in the transformed coordinate space and consider the gradient flow

$$\frac{d\boldsymbol{q}}{dt} = -\nabla_{\boldsymbol{q}}\ell(\boldsymbol{q}), \tag{1}$$

with $\ell(\boldsymbol{q})$ as in equation 7.

**Theorem 3.** *When initializing the gradient flow in equation 1 randomly, then with probability one the solution convergences*

$$\boldsymbol{q}(t) \xrightarrow{t\to\infty} \boldsymbol{q}^*,$$

*where $\boldsymbol{q}^*$ is one of the global minima of the loss in equation 7, i.e., $\pm\boldsymbol{q}$. Therefore, the limit model, $f(\boldsymbol{x};\boldsymbol{q}^*)$, approximates the signed distance function to $\boldsymbol{e}_1^{\perp}$ (i.e., $\mathcal{H}$ in the transformed coordinates).*

We prove Theorem 3 using a certain *Liapunov function* (explained shortly). By random initialization we mean $\boldsymbol{q}^0$ is drawn from some continuous probability distribution in $\mathbb{R}^d$ (i.e., with a density function). Note that with probability one $\boldsymbol{q}^0$ is not orthogonal to $\boldsymbol{e}_1$. Let $\boldsymbol{v}$ be one of $\pm\boldsymbol{q} = \pm\sqrt{1-\frac{\lambda_1}{2\lambda}}\boldsymbol{e}_1$ from Theorem 1 so that $\boldsymbol{v}^T\boldsymbol{q}^0 > 0$.

**Liapunov function.** To show that $\boldsymbol{q}(t)$ converges to $\boldsymbol{v}$ we will introduce a *Liapunov function*; the existence of such a function implies the desired convergence using standard stability results from the theory of dynamical systems (Wiggins, 2003; Teschl, 2012). Consider the domain $\Omega = \left\{\boldsymbol{q} \in \mathbb{R}^d | \boldsymbol{e}_1^T \boldsymbol{q} > 0\right\}$. $h : \Omega \to \mathbb{R}$ is a *Liapunov function* if it is $C^1$ and satisfies the following conditions:

1. *Energy:* $h(\boldsymbol{v}) = 0$ and $h(\boldsymbol{q}) > 0$ for all $\boldsymbol{q} \in \Omega \setminus \{\boldsymbol{v}\}$.
2. *Decreasing:* $\nabla h(\boldsymbol{q}) \cdot \frac{d\boldsymbol{q}}{dt}(\boldsymbol{q}) < 0$ for all $\boldsymbol{w} \in \Omega \setminus \{\boldsymbol{v}\}$.
3. *Bounded:* The level-sets $\{\boldsymbol{q} | h(\boldsymbol{q}) = c\}$ are bounded.

Intuitively, a Liapunov function can be imagined as a sort of an energy function (i.e., non-negative) that vanishes only at $\boldsymbol{v}$ and that the flow defined by equation 1 strictly decreases its value at every point, except at the fixed point $\boldsymbol{v}$. These conditions imply that if a flow (i.e., integral curve) starting at $\boldsymbol{q}_0 \in \Omega$ stays bounded it has to converge to $\boldsymbol{v}$. See for example Theorem 6.14 in (Teschl, 2012). Now, consider

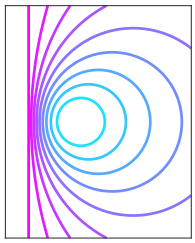


*Figure 1.* Level-sets of $h$.

$$h(\boldsymbol{q}) = \frac{\|\boldsymbol{q} - \boldsymbol{v}\|^2}{1 + \|\boldsymbol{q}\|^2}. \tag{2}$$

We will prove that $h$ is Liapunov for our problem. First it clearly satisfies the *energy* condition. The *bounded* condition can be seen by noting that $h(\boldsymbol{q}) \in [0, 1)$ for all $\boldsymbol{q} \in \Omega$ and that in the quadratic equation $h(\boldsymbol{q}) = c$ the quadratic term has the form $(1 - c) \|\boldsymbol{w}\|^2$ and since $(1 - c) > 0$ the level-sets of $h$ are all finite-radius circles, see Figure 1.

To prove the *decreasing* property a direct computation shows that for $\boldsymbol{q} \in \Omega$

$$\nabla h \cdot \frac{d\boldsymbol{q}}{dt} = \frac{-8\boldsymbol{v}^T \boldsymbol{q}}{(1 + \|\boldsymbol{q}\|^2)^2} \left( \boldsymbol{q}^T D \boldsymbol{q} + \lambda \left( \|\boldsymbol{q}\|^2 - 1 \right)^2 \right) \leq 0$$

where in the last inequality we used the fact that $\boldsymbol{v}^T \boldsymbol{q} = q_1 > 0$, and $D$ is a positive semi-definite matrix, i.e., $\lambda_i \geq 0$, $i \in [d]$. Furthermore, if the r.h.s. equals zero then $\boldsymbol{q}^T D \boldsymbol{q} = 0$ and $\|\boldsymbol{q}\| = 1$; this implies that $\boldsymbol{q} = \boldsymbol{v}$. Therefore for all $\boldsymbol{q} \in \Omega \setminus \{\boldsymbol{q}\}$ we have $\nabla h \cdot \frac{d\boldsymbol{q}}{dt} < 0$. □

**Relation to Theorem 2.** Although this seems as a special case of Theorem 2, note that it works for the continuous gradient flow. This is in contrast to the proof of Theorem 2 that uses the result of (Lee et al., 2016) building upon the discrete nature of gradient descent iterations. Furthermore, we believe a simple self-contained convergence proof that does not rely on previous work could be of merit.

## References


Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lee, J. D., Simchowitz, M., Jordan, M. I., and Recht, B. Gradient descent converges to minimizers. *arXiv preprint arXiv:1602.04915*, 2016.

Park, J. J., Florence, P., Straub, J., Newcombe, R., and Lovegrove, S. Deepsdf: Learning continuous signed distance functions for shape representation. *arXiv preprint arXiv:1901.05103*, 2019.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. 2017.

Teschl, G. *Ordinary differential equations and dynamical systems*, volume 140. American Mathematical Soc., 2012.

Wiggins, S. *Introduction to applied nonlinear dynamical systems and chaos*, volume 2. Springer Science & Business Media, 2003.

*Figure 2.* Additional results from D-Faust shape space experiment (see Section 7.2 in main paper). Left - train results, right - test results. In each row (left to right): Registration (not used), raw scan (source of input point clouds), our result, and SAL result. Back-faces are colored in magenta.