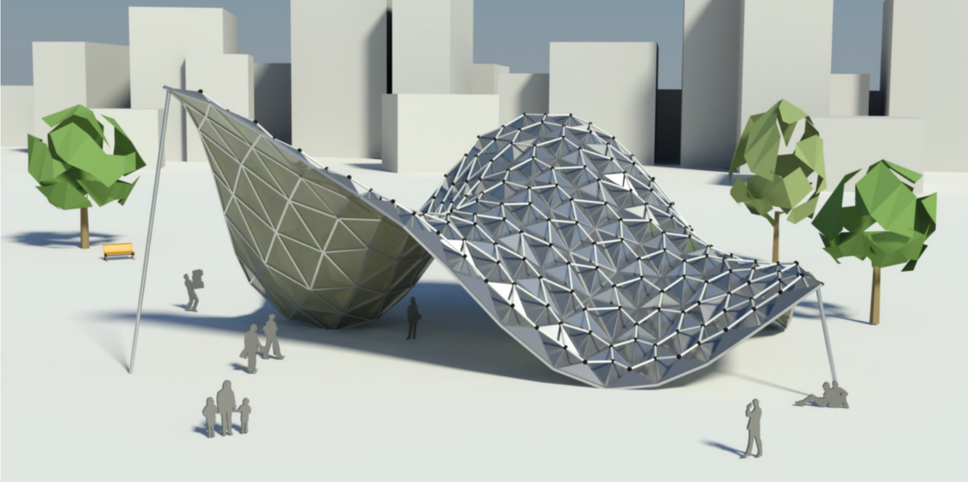
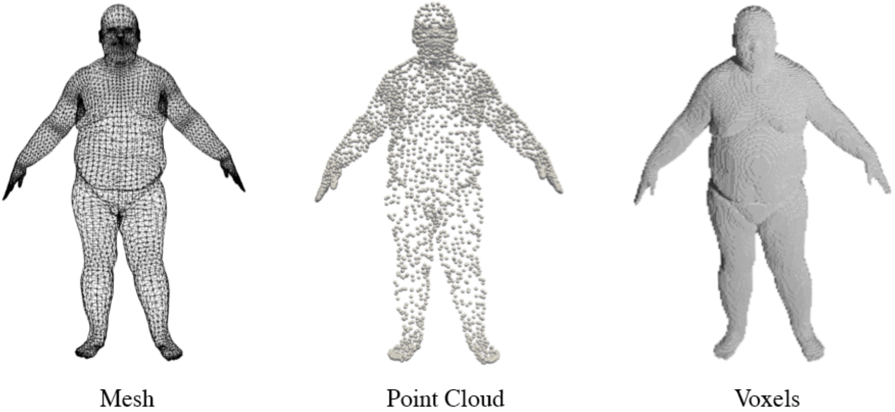


# Sign Agnostic Learning with Derivatives of 3D Geometry from Raw Data

Debabrata Ghosh

Visual Computing Institute | RWTH AACHEN UNIVERSITY

# Representations of 3D shapes



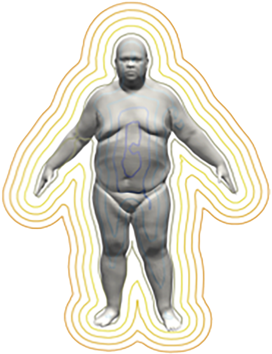

Discrete Representation of 3D Shape [1]

# Representations of 3D shapes

Surface represented as zero level-set: $\mathcal{S} = \left\{ \boldsymbol{x} \in \mathbb{R}^3 \mid f(\boldsymbol{x}) = 0 \right\}$

# Representations of 3D shapes

Surface represented as zero level-set: $\mathcal{S} = \{\boldsymbol{x} \in \mathbb{R}^3 \mid f(\boldsymbol{x}) = 0\}$



Continuous (Implicit) Representation of 3D Shape [1]

# Implicit Neural Representation

## Why Neural Networks

- flexibility and approximation power [4]

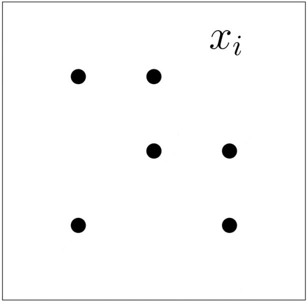
# Implicit Neural Representation

## Why Neural Networks

- flexibility and approximation power [4]
- optimization and generalization properties

# Learning Implicit Neural Representation

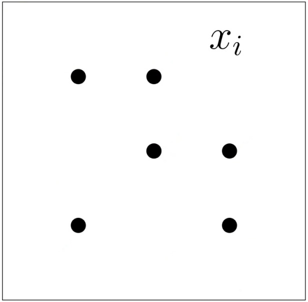
## Methods



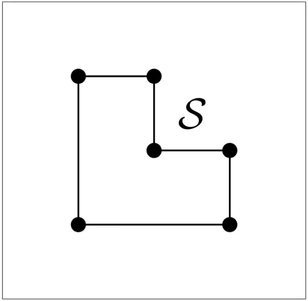

Input data [7]

# Learning Implicit Neural Representation

## Methods



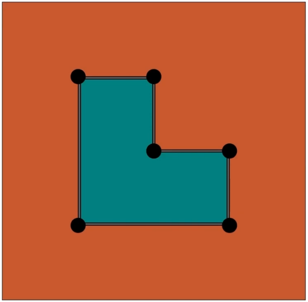

Input data [7]




Output shape [7]

# Learning Implicit Neural Representation

## Methods

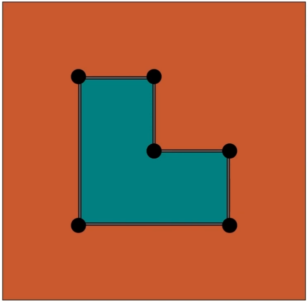


Occupancy function

$$\Phi_{ind}(x) = \begin{cases} 0 & x \text{ outside} \\ 1 & x \text{ inside} \end{cases} \quad [7]$$

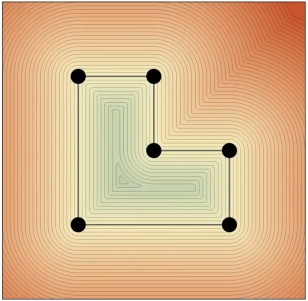
# Learning Implicit Neural Representation

## Methods



Occupancy function

$$\Phi_{ind}(x) = \begin{cases} 0 & x \text{ outside} \\ 1 & x \text{ inside} \end{cases} \text{[7]}$$

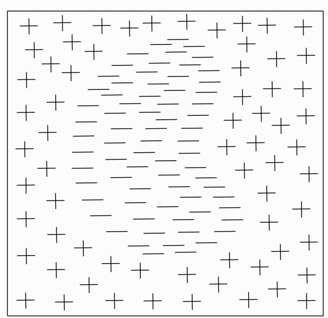


Signed distance function

$$\Phi_{sdf}(x) = (-1)^{\Phi_{ind}(x)} \min_{\boldsymbol{z}\in\mathcal{X}} \|\boldsymbol{x}-\boldsymbol{z}\|_2 \text{ [7]}$$

# Learning Implicit Neural Representation

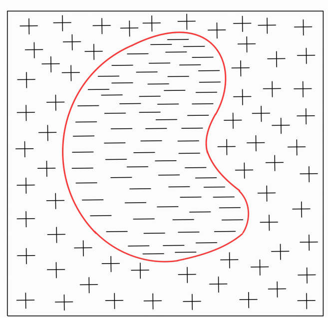
## Challenges



Data available for supervised learning [2]

# Learning Implicit Neural Representation

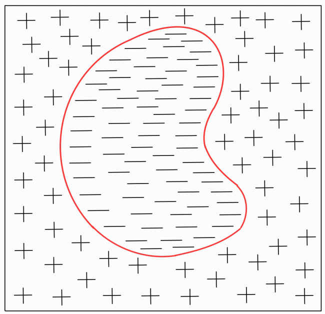
## Challenges



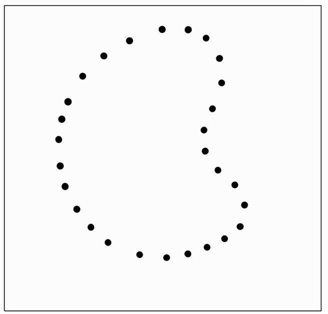
Data available for supervised learning [2]

# Learning Implicit Neural Representation

## Challenges



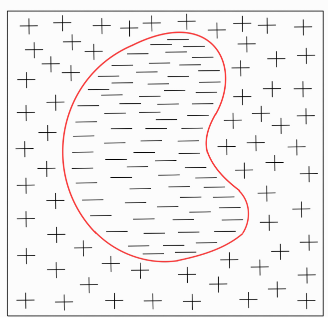
Data available for supervised learning [2]



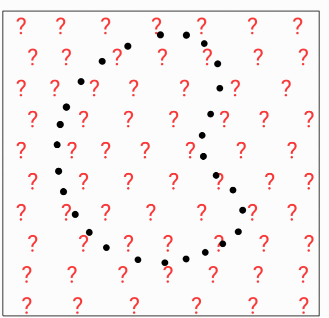
Raw data [2]

# Learning Implicit Neural Representation

## Challenges



Data available for supervised learning [2]

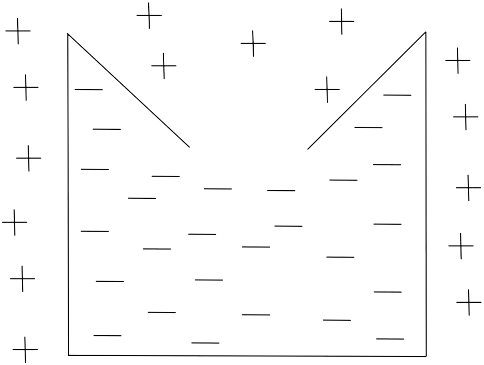


Raw data [2]


Debabrata Ghosh

# Learning Implicit Neural Representation

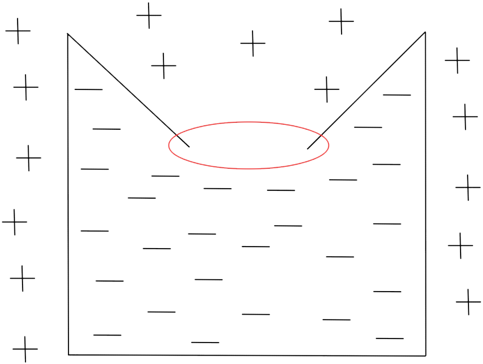
## Challenges



Learning missing parts [2]

# Learning Implicit Neural Representation

## Challenges

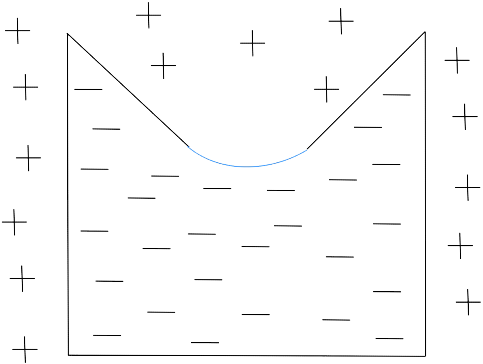


Learning missing parts [2]


Debabrata Ghosh

# Learning Implicit Neural Representation

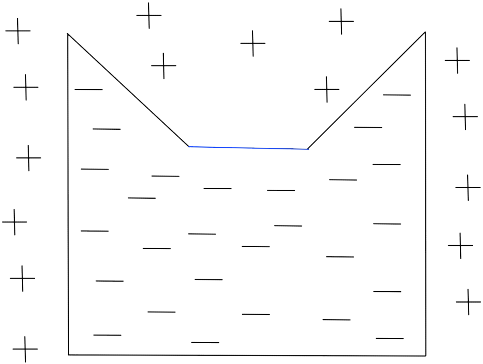
## Challenges



Learning missing parts [2]

# Learning Implicit Neural Representation

## Challenges



Learning missing parts [2]

# Background: Sign Agnostic Learning (SAL)

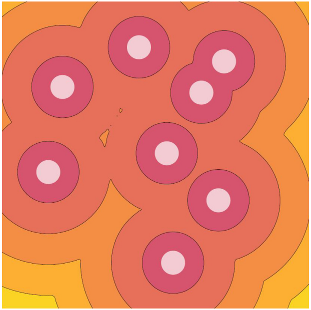
## Unsigned Distance Function

$L^0$ distance, $h_0(\boldsymbol{z}) = \begin{cases} 0 & \boldsymbol{z} \in \mathcal{X} \\ 1 & \boldsymbol{z} \notin \mathcal{X} \end{cases}$ [2]

# Background: Sign Agnostic Learning (SAL)

## Unsigned Distance Function

$L^0$ distance, $h_0(\boldsymbol{z}) = \begin{cases} 0 & \boldsymbol{z} \in \mathcal{X} \\ 1 & \boldsymbol{z} \notin \mathcal{X} \end{cases}$ [2]



$L^2$ (Euclidean) distance,
$h_2(\boldsymbol{z}) = \min_{\boldsymbol{x} \in \mathcal{X}} \|\boldsymbol{z} - \boldsymbol{x}\|_2$ [2]

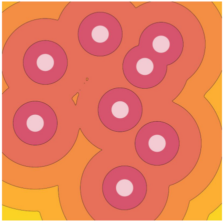
# Background: Sign Agnostic Learning (SAL)

## SAL Loss Function

$$\mathrm{loss}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}} \ell \left( f(\boldsymbol{x}; \boldsymbol{\theta}), h(\boldsymbol{x}) \right)$$

Loss function

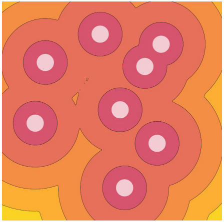
Unsigned distance



# Background: Sign Agnostic Learning (SAL)

## SAL Loss Function

$$\mathrm{loss}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}} \left| f(\boldsymbol{x};\boldsymbol{\theta}), h(\boldsymbol{x}) \right|$$
$$\mathrm{loss}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}} \left( f(\boldsymbol{x};\boldsymbol{\theta}), h(\boldsymbol{x}) \right)^2$$

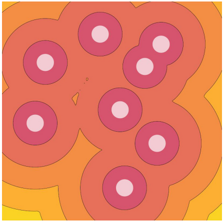


# Background: Sign Agnostic Learning (SAL)

## SAL Loss Function

$$\mathrm{loss}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}} \left| f(\boldsymbol{x};\boldsymbol{\theta}), h(\boldsymbol{x}) \right| \quad \times$$

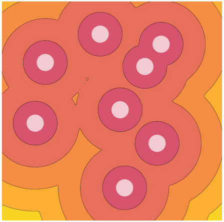
$$\mathrm{loss}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}} \left( f(\boldsymbol{x};\boldsymbol{\theta}), h(\boldsymbol{x}) \right)^2 \quad \times$$



# Background: Sign Agnostic Learning (SAL)

## SAL Loss Function

$$\mathrm{loss}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}} \underset{\substack{\downarrow \\ \text{Sign agnostic dissimilarity}}}{\tau} \left(f(\boldsymbol{x};\boldsymbol{\theta}), h(\boldsymbol{x})\right)$$



# Background: Sign Agnostic Learning (SAL)

## SAL Loss Function

$$\mathrm{loss}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}} \underbrace{\tau}_{\text{Sign agnostic dissimilarity}} \left( f(\boldsymbol{x}; \boldsymbol{\theta}), h(\boldsymbol{x}) \right)$$

Sign agnostic dissimilarity

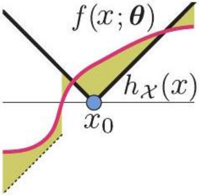
$$\tau(a, b) = ||a| - b|$$

# Background: Sign Agnostic Learning (SAL)

## SAL Loss Function

$$\mathrm{loss}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}}\tau\left(f(\boldsymbol{x};\boldsymbol{\theta}), h(\boldsymbol{x})\right)$$

$\tau(a,b) = ||a| - b|$ $h(x) = |x - x_0|$

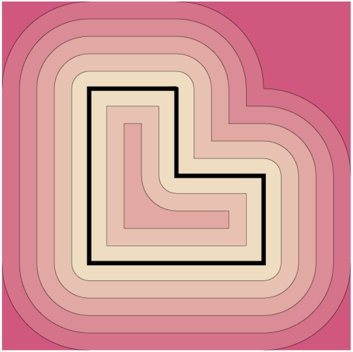



Sign agnostic learning in 1-D case [2]


Debabrata Ghosh

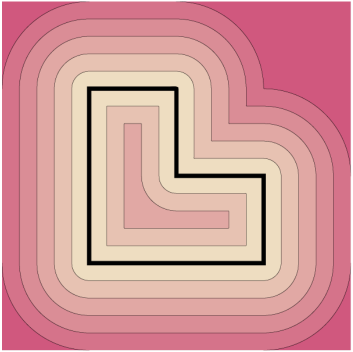
Visual Computing Institute
RWTH AACHEN UNIVERSITY

# Background: Limitations of SAL

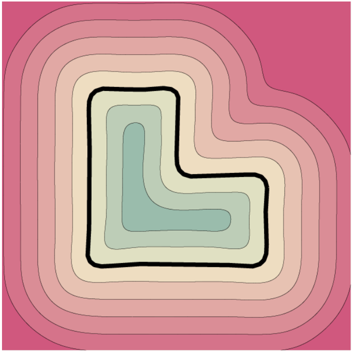


Unsigned distance [3]

# Background: Limitations of SAL

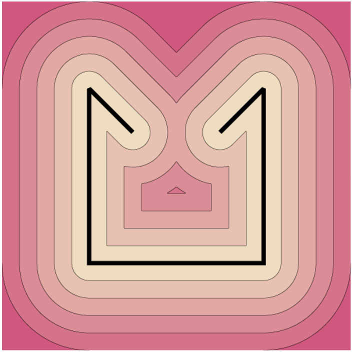


Unsigned distance [3]



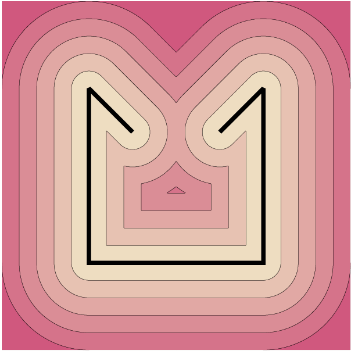
Level sets after SAL training [3]

# Background: Limitations of SAL

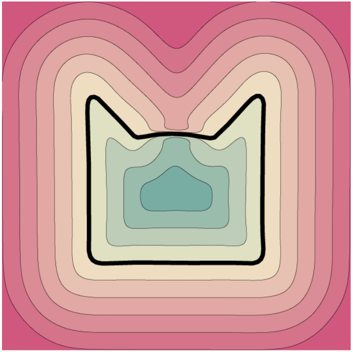


Unsigned distance [3]

# Background: Limitations of SAL



Unsigned distance [3]



Level sets after SAL training [3]

# Background: Sobolev Training

## Adding Derivatives

- Target function: $f\left(x; \boldsymbol{\theta}\right) = \max\{ax, bx\} + c$ [8]

# Background: Sobolev Training

## Adding Derivatives

- Target function: $f\left(x;\boldsymbol{\theta}\right) = \max\{ax, bx\} + c$ [8]
- Learned function: NN with one hidden layer [8]

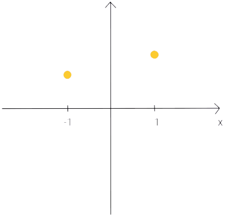
# Background: Sobolev Training

## Adding Derivatives

- Target function: $f\left(x;\boldsymbol{\theta}\right) = \max\{ax, bx\} + c$ [8]
- Learned function: NN with one hidden layer [8]
- Data given for $x = \{-1, 1\}$ [8]

## Adding Derivatives

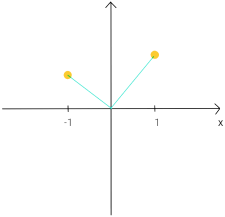
- Target function: $f\left(x;\boldsymbol{\theta}\right) = \max\{ax, bx\} + c$ [8]
- Learned function: NN with one hidden layer [8]
- Data given for $x = \{-1, 1\}$ [8]



# Background: Sobolev Training

## Adding Derivatives

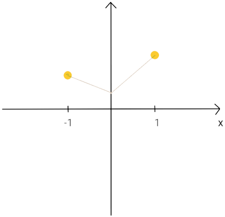
- Target function: $f\left(x;\boldsymbol{\theta}\right) = \max\{ax, bx\} + c$ [8]
- Learned function: NN with one hidden layer [8]
- Data given for $x = \{-1, 1\}$ [8]



# Background: Sobolev Training

## Adding Derivatives

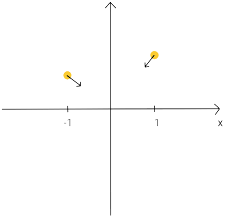
- Target function: $f\,(x;\boldsymbol{\theta}) = \max\{ax, bx\} + c$ [8]
- Learned function: NN with one hidden layer [8]
- Data given for $x = \{-1, 1\}$ [8]



# Background: Sobolev Training

## Adding Derivatives

- Target function: $f\left(x;\boldsymbol{\theta}\right) = \max\{ax, bx\} + c$ [8]
- Learned function: NN with one hidden layer [8]
- Data given for $x = \{-1, 1\}$ [8]





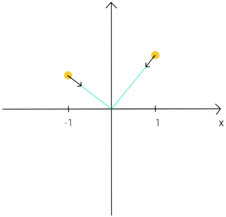
Debabrata Ghosh

Visual Computing Institute
RWTH AACHEN UNIVERSITY

# Background: Sobolev Training

## Adding Derivatives

- Target function: $f(x;\boldsymbol{\theta}) = \max\{ax, bx\} + c$ [8]
- Learned function: NN with one hidden layer [8]
- Data given for $x = \{-1, 1\}$ [8]





Visual Computing Institute | RWTH AACHEN UNIVERSITY

## Extension of SAL Loss

$$\mathrm{loss}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}} \tau \left( f(\boldsymbol{x}; \boldsymbol{\theta}), h(\boldsymbol{x}) \right)$$

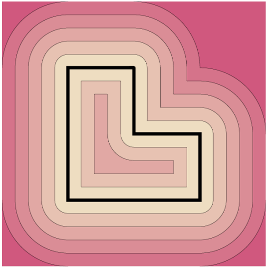
# Sign Agnostic Learning with Derivatives (SALD)

## Extension of SAL Loss

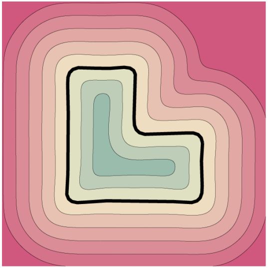
$$\text{loss}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}} \tau \left( f(\boldsymbol{x}; \boldsymbol{\theta}), h(\boldsymbol{x}) \right) + \mathbb{E}_{\boldsymbol{x}} \tau \left( \nabla_{\boldsymbol{x}} f(\boldsymbol{x}; \boldsymbol{\theta}), \nabla_{\boldsymbol{x}} h(\boldsymbol{x}) \right)$$

# Sign Agnostic Learning with Derivatives (SALD)

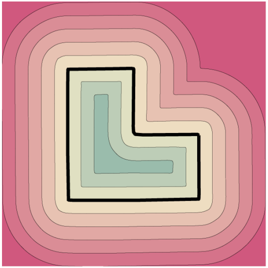
## Results in 2D



Unsigned distance [3]



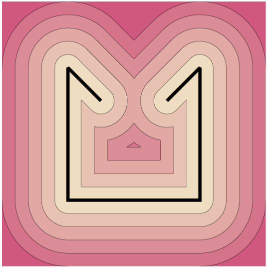
Level sets after SAL training [3]



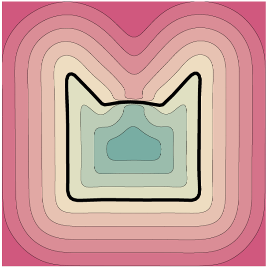
Level sets after SALD training [3]

# Sign Agnostic Learning with Derivatives (SALD)

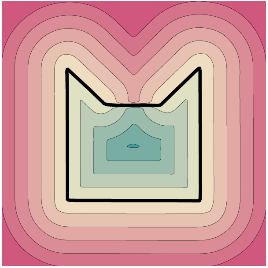
## Results in 2D



Unsigned distance [3]



Level sets after SAL training [3]



Level sets after SALD training [3]



Debabrata Ghosh


Visual Computing Institute
RWTH AACHEN UNIVERSITY

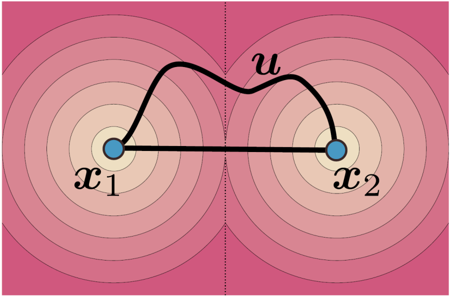
## Minimal Surface Property

Minimizes the surface area of missing parts [13, 3].

# Sign Agnostic Learning with Derivatives (SALD)

## Minimal Surface Property

Minimizes the surface area of missing parts [13, 3].

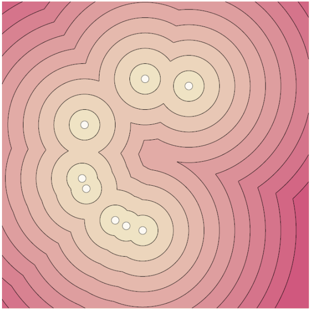



Minimal Surface Property in 2D [3]

# Sign Agnostic Learning with Derivatives (SALD)

## Minimal Surface Property

Minimizes the surface area of missing parts [13, 3].



Unsigned distance [3]

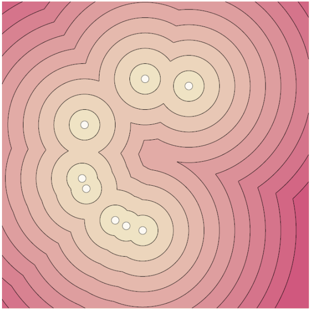

Debabrata Ghosh

Visual Computing Institute
RWTH AACHEN UNIVERSITY

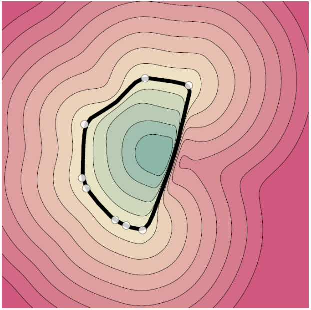
# Sign Agnostic Learning with Derivatives (SALD)

## Minimal Surface Property

Minimizes the surface area of missing parts [13, 3].

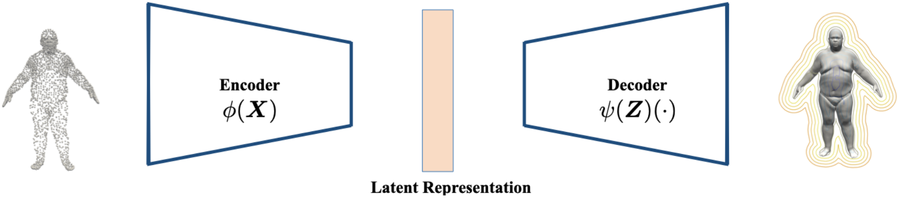


Unsigned distance [3]



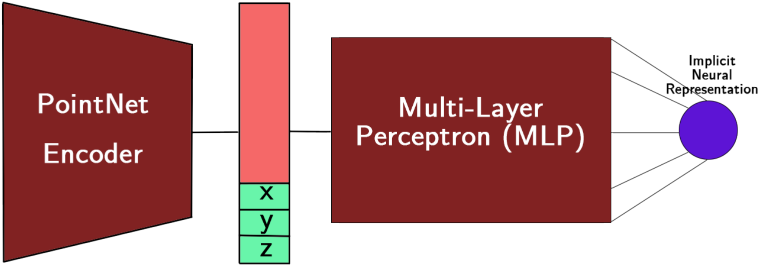
Level sets after SALD training [3]

# Neural Network Architecture



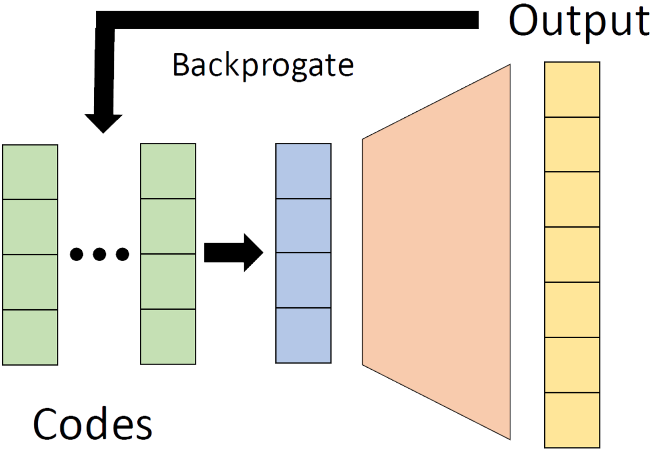

Variational Auto Encoder [10, 1]

# Neural Network Architecture




Variational Auto Encoder with PointNet Encoder [10, 6]

# Neural Network Architecture




Auto Decoder [11, 3]

## Evaluation Metrics

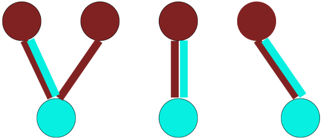
Chamfer distance: $$\mathrm{d_C}\left(\mathcal{X}_1, \mathcal{X}_2\right) = \frac{1}{2}\left(\mathrm{d}_{\overrightarrow{\mathrm{C}}}\left(\mathcal{X}_1, \mathcal{X}_2\right) + \mathrm{d}_{\overrightarrow{\mathrm{C}}}\left(\mathcal{X}_2, \mathcal{X}_1\right)\right),$$

$$\mathrm{d}_{\overrightarrow{\mathrm{C}}}\left(\mathcal{X}_1, \mathcal{X}_2\right) = \frac{1}{\left|\mathcal{X}_1\right|} \sum_{\boldsymbol{x}_1 \in \mathcal{X}_1} \min_{\boldsymbol{x}_2 \in \mathcal{X}_2} \left\|\boldsymbol{x}_1 - \boldsymbol{x}_2\right\|$$

## Evaluation Metrics

Chamfer distance: $$\mathrm{d_C}\left(\mathcal{X}_1, \mathcal{X}_2\right) = \frac{1}{2}\left(\mathrm{d}_{\mathrm{C}}^{\rightarrow}\left(\mathcal{X}_1, \mathcal{X}_2\right) + \mathrm{d}_{\mathrm{C}}^{\rightarrow}\left(\mathcal{X}_2, \mathcal{X}_1\right)\right),$$

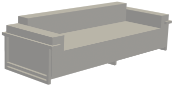
$$\mathrm{d}_{\mathrm{C}}^{\rightarrow}\left(\mathcal{X}_1, \mathcal{X}_2\right) = \frac{1}{\left|\mathcal{X}_1\right|} \sum_{\boldsymbol{x}_1 \in \mathcal{X}_1} \min_{\boldsymbol{x}_2 \in \mathcal{X}_2} \left\|\boldsymbol{x}_1 - \boldsymbol{x}_2\right\|$$

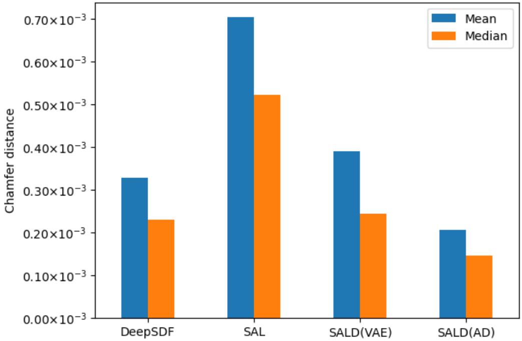


Chamfer distance between 2 point clouds

# Experiments and Results

## ShapeNet

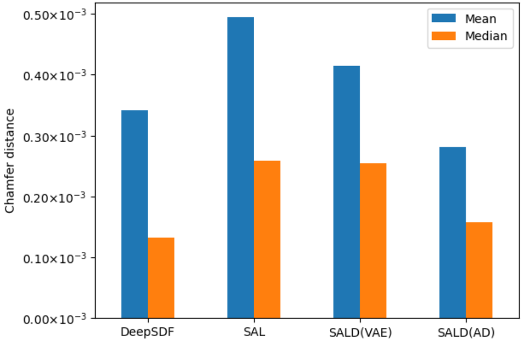





ShapeNet quantitative results [3]

# Experiments and Results

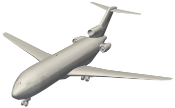
## ShapeNet

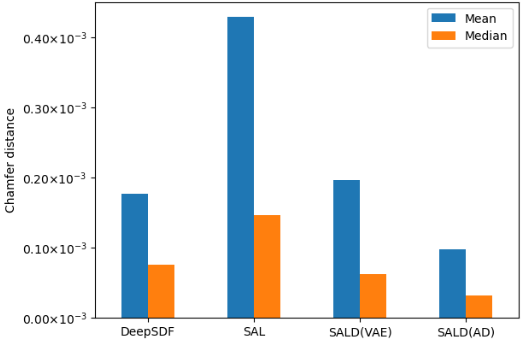



ShapeNet quantitative results [3]

# Experiments and Results

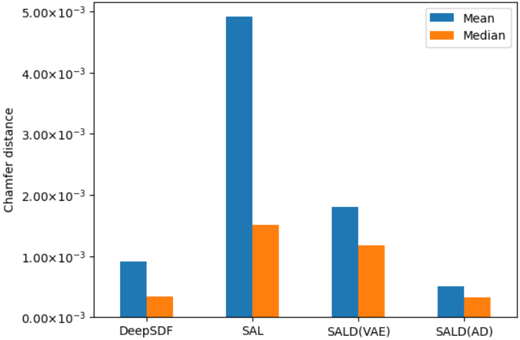
## ShapeNet






ShapeNet quantitative results [3]

## ShapeNet




ShapeNet quantitative results [3]

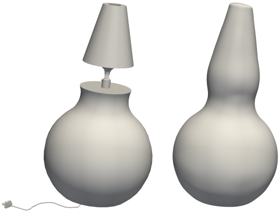
# Experiments and Results

## ShapeNet

Actual SAL DeepSDF SALD

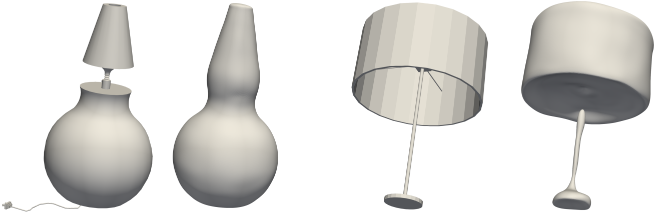
Actual SAL DeepSDF SALD [3]

# Shortcomings



Original SALD [3]

# Shortcomings

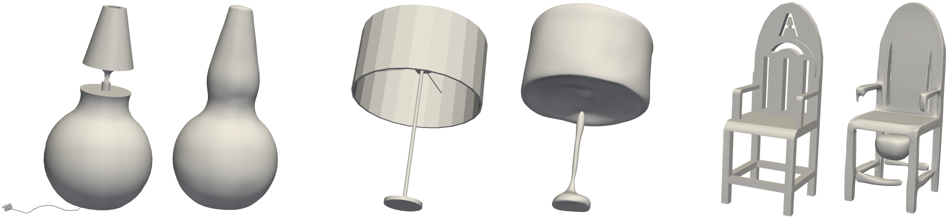


Original SALD [3] Original SALD [3]


Debabrata Ghosh

Visual Computing Institute
RWTH AACHEN UNIVERSITY

# Shortcomings



Original SALD [3] Original SALD [3] Original SALD [3]

# Conclusion and Further Works

- Adding derivatives can improve learned 3D geometry significantly


Debabrata Ghosh

Visual Computing Institute
RWTH AACHEN UNIVERSITY

# Conclusion and Further Works

- Adding derivatives can improve learned 3D geometry significantly
- Sharp features in the reconstructed shapes

# Conclusion and Further Works

- Adding derivatives can improve learned 3D geometry significantly
- Sharp features in the reconstructed shapes
- Favourable minimal surface property

# Conclusion and Further Works

SAL

LightSAL

Model efficiency: point-cloud reconstruction after 500 training epochs [5]

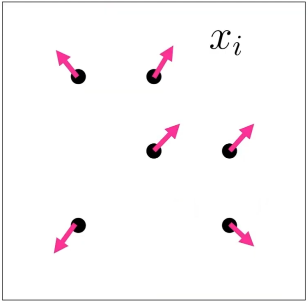


Debabrata Ghosh



Visual Computing Institute
RWTH AACHEN UNIVERSITY

# Conclusion and Further Works

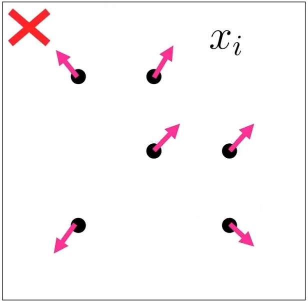



Input data [7]

DiGS: Divergence guided shape implicit neural representation for unoriented point clouds


Debabrata Ghosh

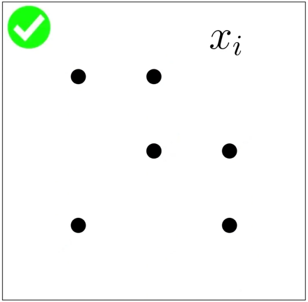
# Conclusion and Further Works




Input data [7]

DiGS: Divergence guided shape implicit neural representation for unoriented point clouds

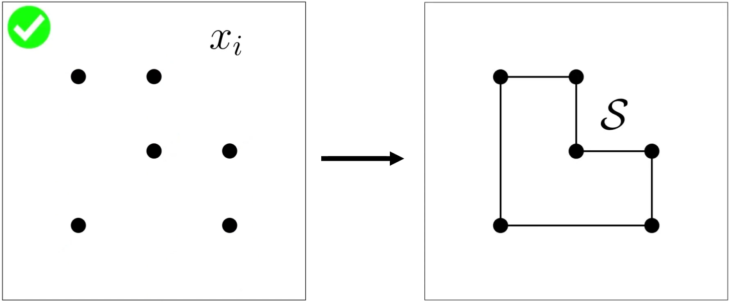
# Conclusion and Further Works




Input data [7]

DiGS: Divergence guided shape implicit neural representation for unoriented point clouds

# Conclusion and Further Works




Input data [7] Output shape [7]

DiGS: Divergence guided shape implicit neural representation for unoriented point clouds

# Conclusion and Further Works

- Combining sign agnostic learning with positional encoding (Fourier features) [3, 12]

# Conclusion and Further Works

- Combining sign agnostic learning with positional encoding (Fourier features) [3, 12]
- Combining the sign-agnostic loss with gradient regularization [3, 9]

# Feedback

Questions?

# References


[1] Matan Atzmon. “Learning Algorithms for Shape Analysis and Shape Synthesis”. PhD thesis. 2022.

[2] Matan Atzmon and Yaron Lipman. “SAL: Sign Agnostic Learning of Shapes From Raw Data”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020.

[3] Matan Atzmon and Yaron Lipman. “SALD: Sign Agnostic Learning with Derivatives”. In: *9th International Conference on Learning Representations, ICLR 2021*. 2021.

[4] Matan Atzmon et al. “Controlling neural level sets”. In: *Advances in Neural Information Processing Systems*. 2019, pp. 2032–2041.

[5] Abol Basher, Muhammad Sarmad, and Jani Boutellier. “LightSAL: Lightweight Sign Agnostic Learning for Implicit Surface Representation”. In: *CoRR* abs/2103.14273 (2021). arXiv: `2103.14273`.

# References II


[6] R. Charles et al. "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, July 2017, pp. 77–85. DOI: 10.1109/CVPR.2017.16.

[7] *CVPR 2022 paper: Divergence guided shape implicit neural representation for unoriented point clouds*. May 2022. URL: https://www.youtube.com/watch?v=bQWpRyM9wYM.

[8] Wojciech M. Czarnecki et al. "Sobolev Training for Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017.

[9] Amos Gropp et al. "Implicit Geometric Regularization for Learning Shapes". In: *Proceedings of Machine Learning and Systems 2020*. 2020, pp. 3569–3579.

# References III


[10] Diederik P. Kingma and Max Welling. "Auto-Encoding Variational Bayes". In: *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2014.

[11] Jeong Joon Park et al. "DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2019.

[12] Matthew Tancik et al. "Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains". In: *NeurIPS* (2020).

[13] Hong-Kai Zhao, S. Osher, and R. Fedkiw. "Fast surface reconstruction using the level set method". In: *Proceedings IEEE Workshop on Variational and Level Set Methods in Computer Vision*. 2001, pp. 194–201. DOI: `10.1109/VLSM.2001.938900`.