Microsoft

CVPR

微软亚洲研究院创研论坛

CVPR 2020 论文分享会

SSRNet: Scalable 3D Surface Reconstruction Network

密振兴，罗一鸣，陶文兵*

华中科技大学人工智能与自动化学院

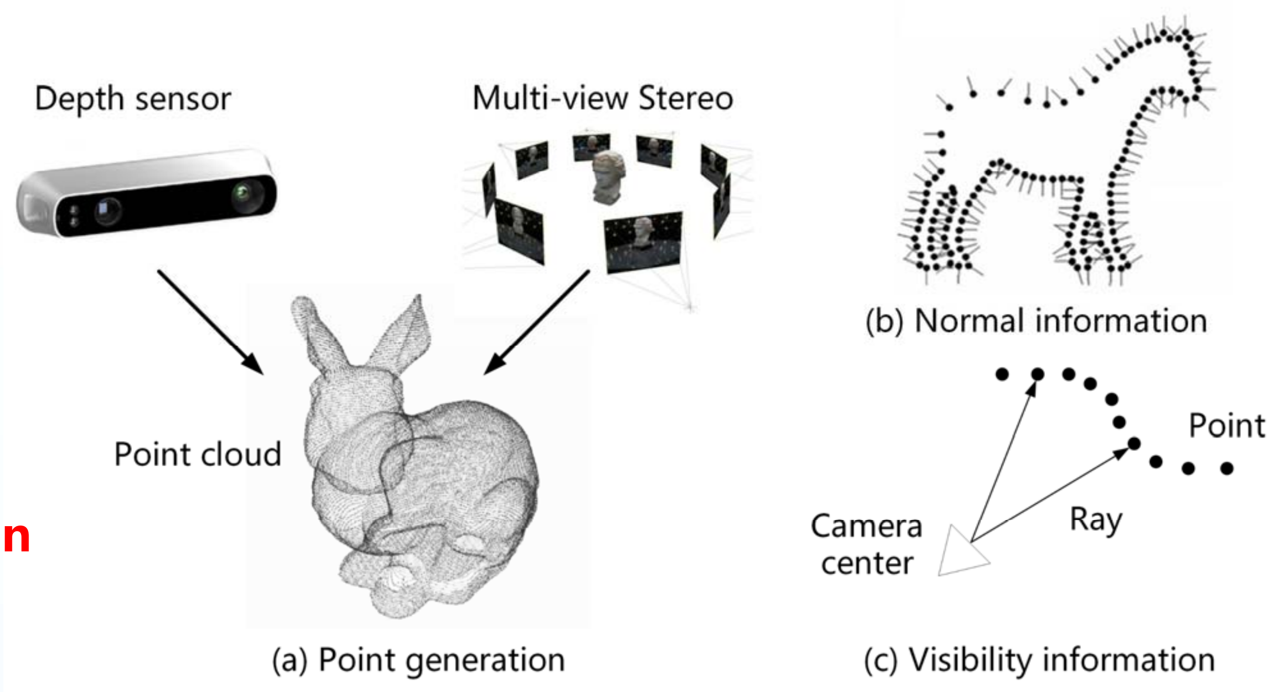
一 Background

01 Point cloud

A set of 3D points sampling from object surfaces

May be with normal or visibility information

- **Pros**
  - **Flexible and sparse**
  - **Easy to be produced by scans and Multi-view Stereo**
- **Cons**
  - **Not easy for manipulation and rendering**

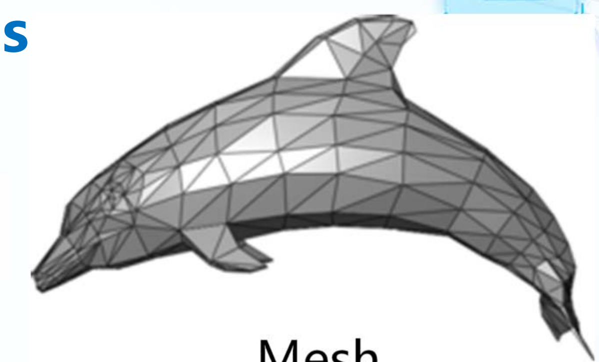


02 Triangular Mesh

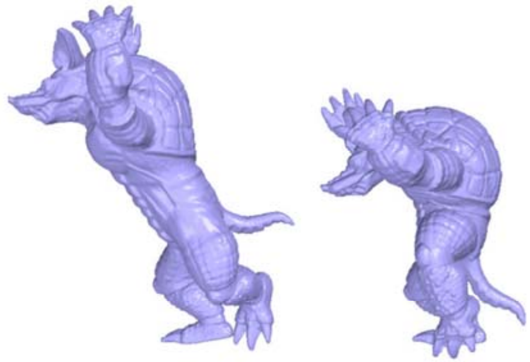
A set of connected triangles representing object surfaces

Consisting of vertices, edges and triangle facets

- **Pros**
  - Supporting efficient traversal, manipulation and rendering
  - Extensively used in computer graphics
- **Cons**
  - Not easy to be directly produced by scans



Mesh

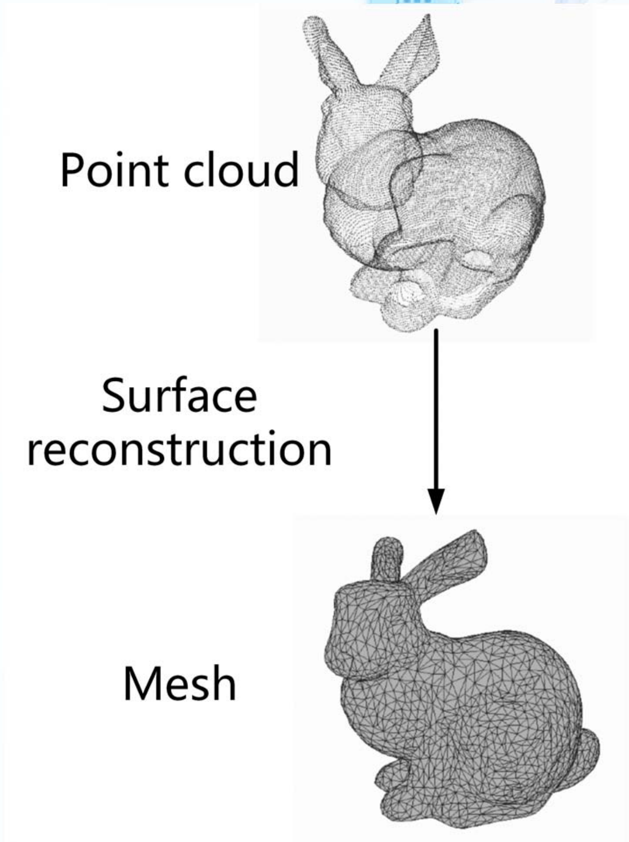


Deformation

Rendering

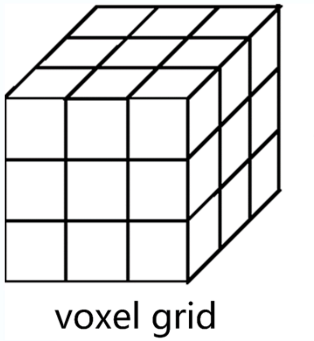
03 Surface Reconstruction

- **Definition**
  - **From point cloud to triangular mesh**
- **Problems**
  - **Non-uniform sampling density of point cloud**
  - **Noise in point cloud**
  - **Outliers far from the true surface**
  - **Missing data due to limited sensor range, high light absorption and occlusions**

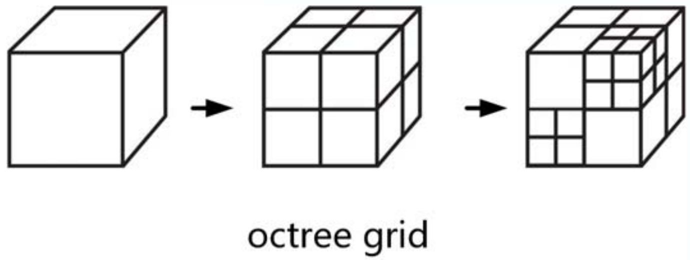


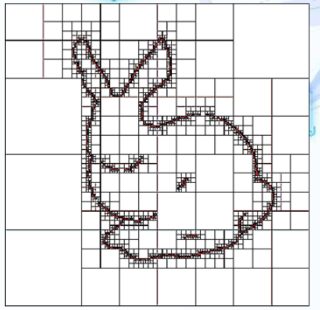
04 3D Space Discretization

- **Voxel Grid**
  - **Simple, uniform but facing complexity issue**

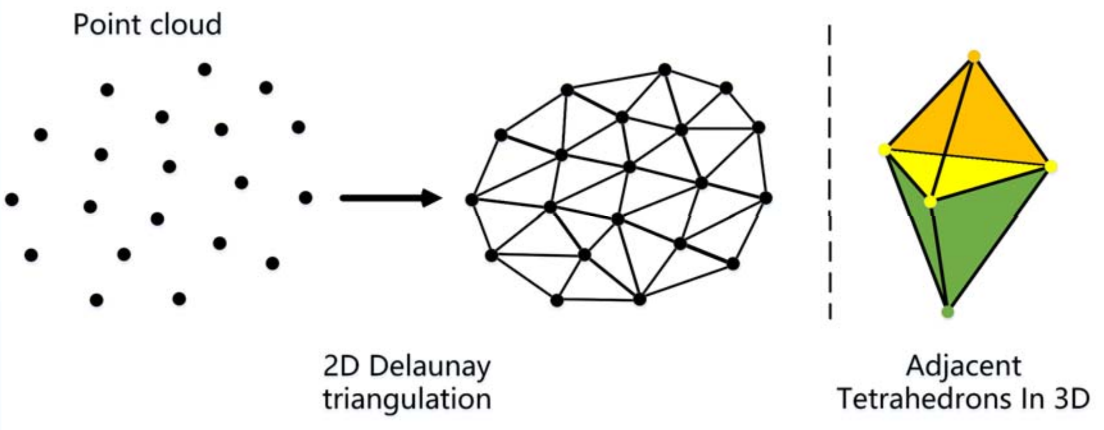


- **Octree Grid**
  - **Flexible and higher resolution**





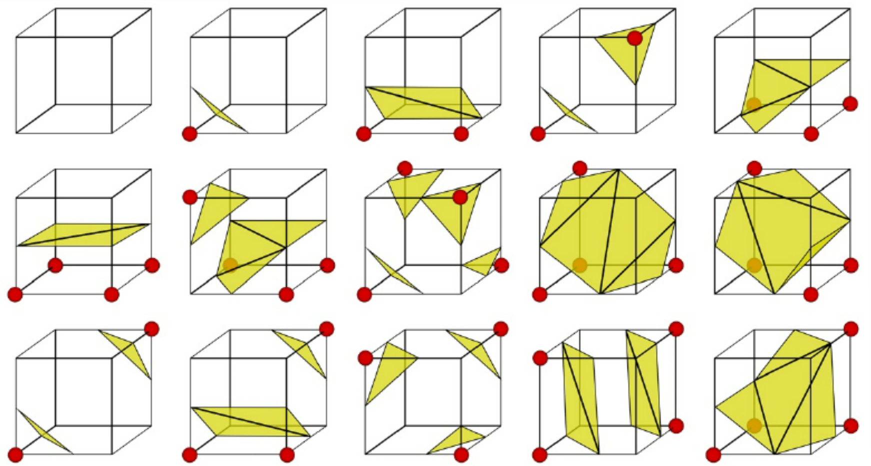
- **Delaunay Triangulation**
  - **Adaptive to density but irregular**
  - **Yielding a graph**

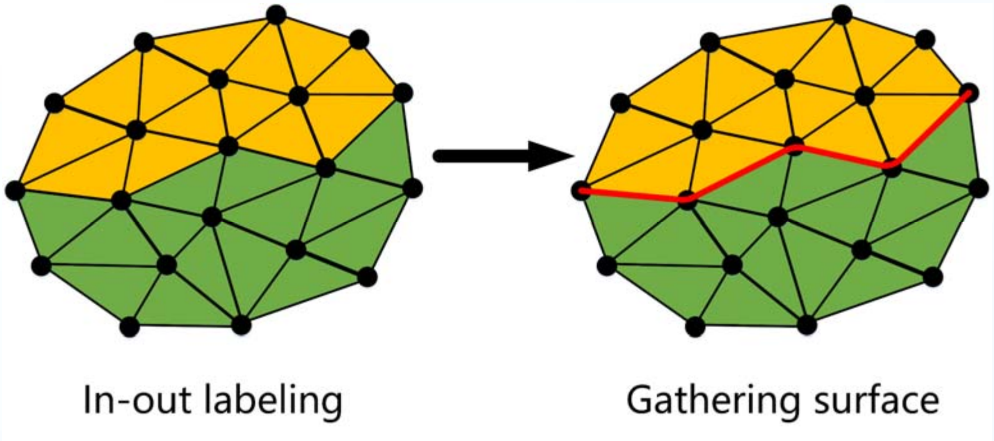


05 How to Get a Triangular Mesh

- **Marching Cubes**
  - **Classify samples as inside or outside the surface by implicit function**
  - **Interpolating intersections on cube edges and extracting triangles**

- **Delaunay triangulation**
  - **Classify tetrahedrons as inside or outside the surface.**
  - **Gathering triangles from adjacent tetrahedrons of different labels**





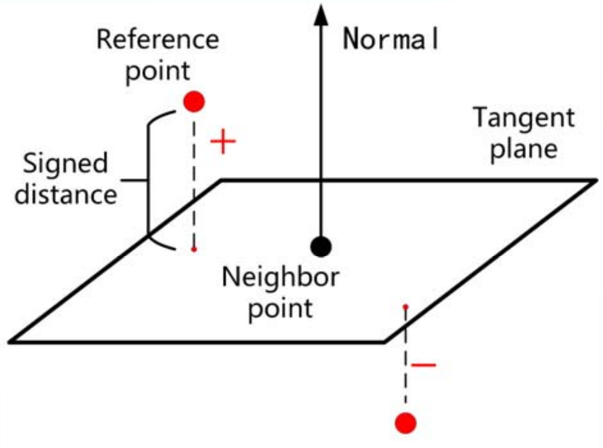
- **William E. Lorensen et al. Siggraph 1987**

二 Related work

01 Local geometric methods

➢ Signed distance Function

- ➢ **Computing for each vertex the signed projection distance onto the tangent plane of its closest point**

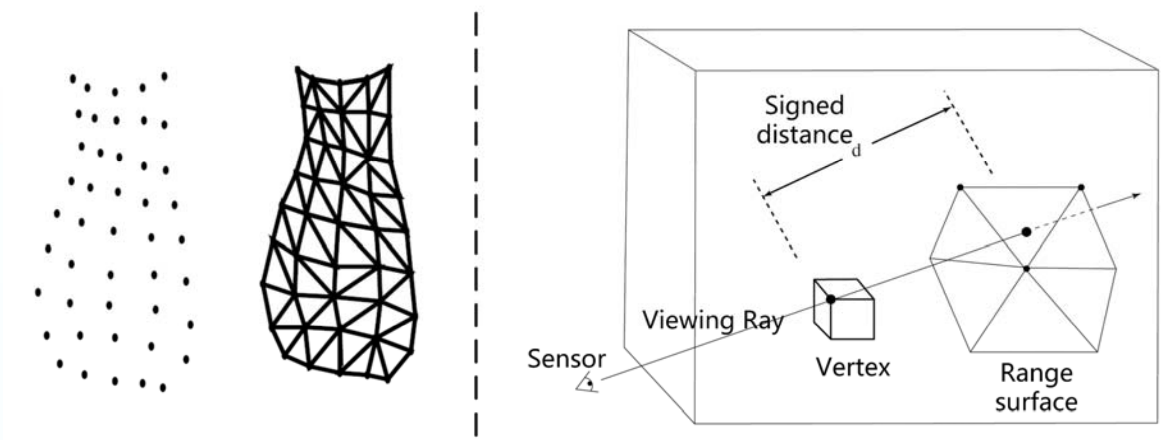


Signed distance

- ■ **Hugues Hoppe et al. 1992**

➢ Truncated Signed distance Function (TSDF)

- ➢ **Computing TSDF from range images for each vertex**



(a) Range image triangulation (b) Computing TSDF

- ■ **Brian Curless et al. 1996**

01 Local geometric methods

➢ Moving least squares (MLS)

- ➢ A class of methods approximating the input points by **spatially varying low-degree** polynomials.

➢ Levin method:

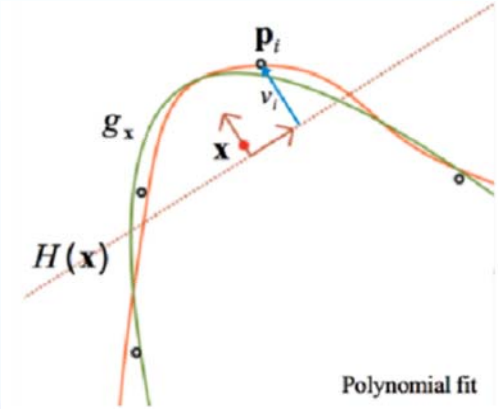
Defining a **height field** using weighted PCA and approximate it by a low-degree **bivariate polynomial.**

➢ Planar fitting:

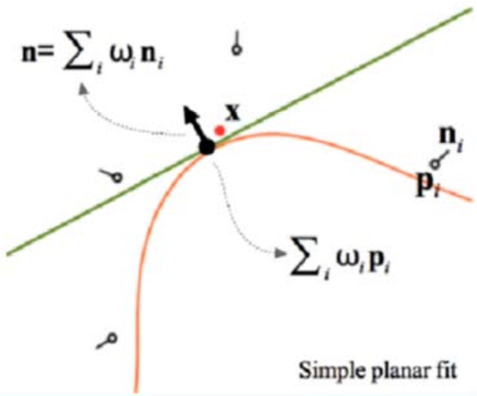
Fitting the **distance** between the evaluation point and the **best fitted plane.**

➢ Spherical fitting:

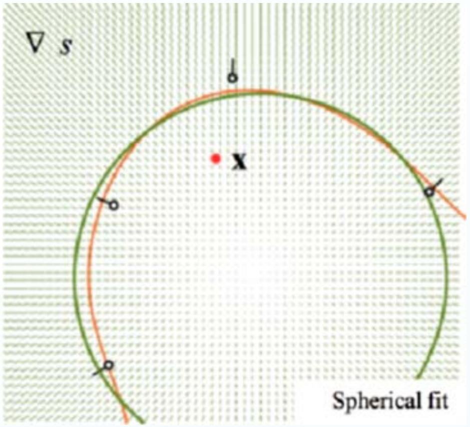
Fitting a **gradient field** of the algebraic **spheres** to the **input normals.**



- Levin et al. 2003



- Amenta et al. 2004



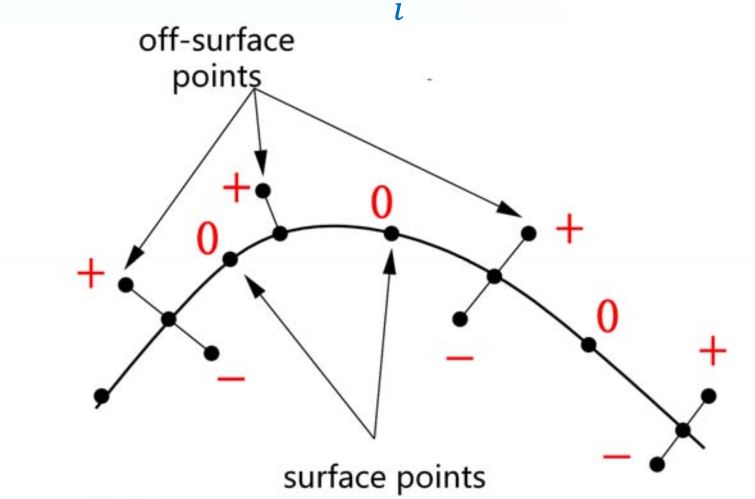
- Guennebaud et al. 2007

02 Global geometric methods

➢ Radial basis functions

➢ **Constructing a signed-distance function and fitting with RBFs**

$$f(\mathbf{x}) = g(\mathbf{x}) + \sum_i \lambda_i \, \phi(\|\mathbf{x} - \mathbf{p}_i\|)$$

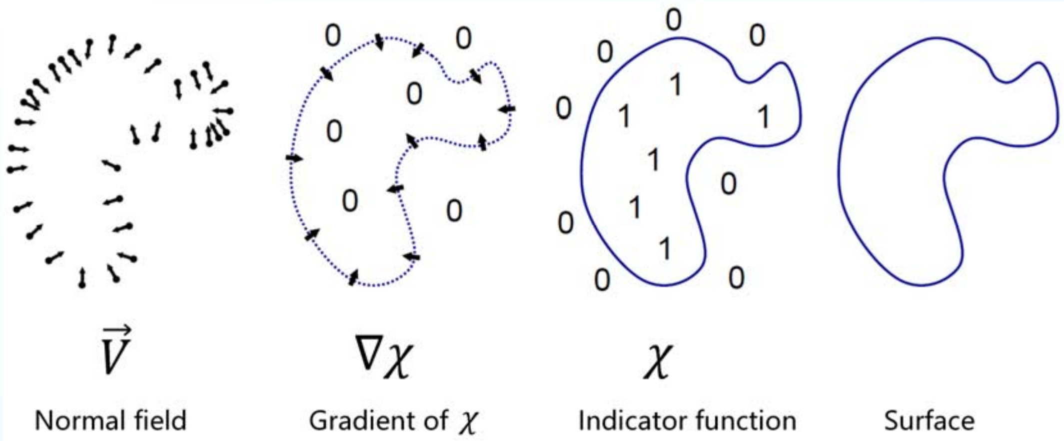


■ J. C. Carr et al. 2001

➢ (Screened) Poisson Surface Reconstruction

➢ **Fitting normal field with gradient of an indicator function and solving a poisson problem**

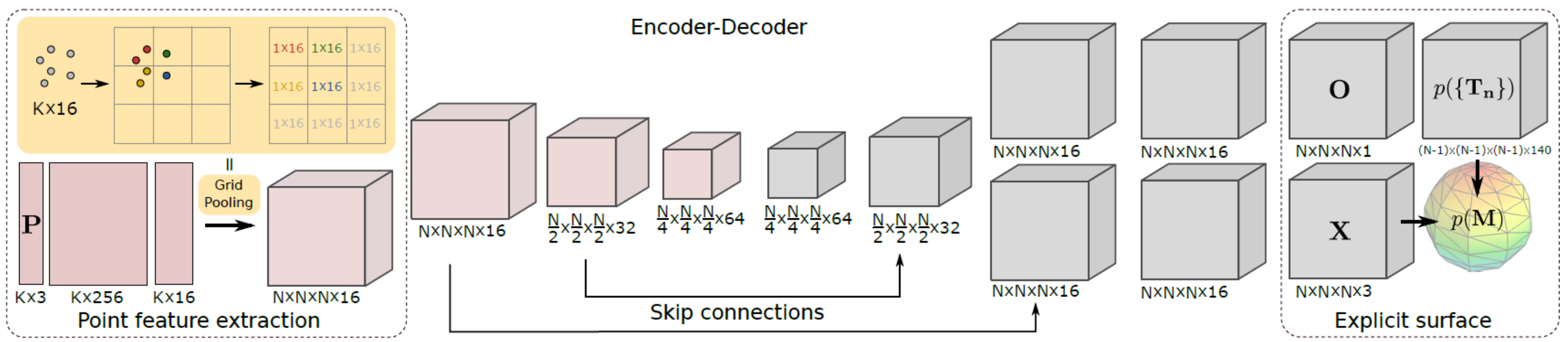
$$\Delta\chi = \nabla \cdot \vec{V}$$



■ Michael Kazhdan et al. ESGP2006, ToG2013

03 Learning-based methods on girds

- **Deep Marching Cubes: Voxel**
  - Converting point features into voxel grids by Grid Pooling
  - Encoding and decoding voxel features via 3D convolution
  - Directly predicting surface by Differentiable Marching Cubes Layer (DMCL)

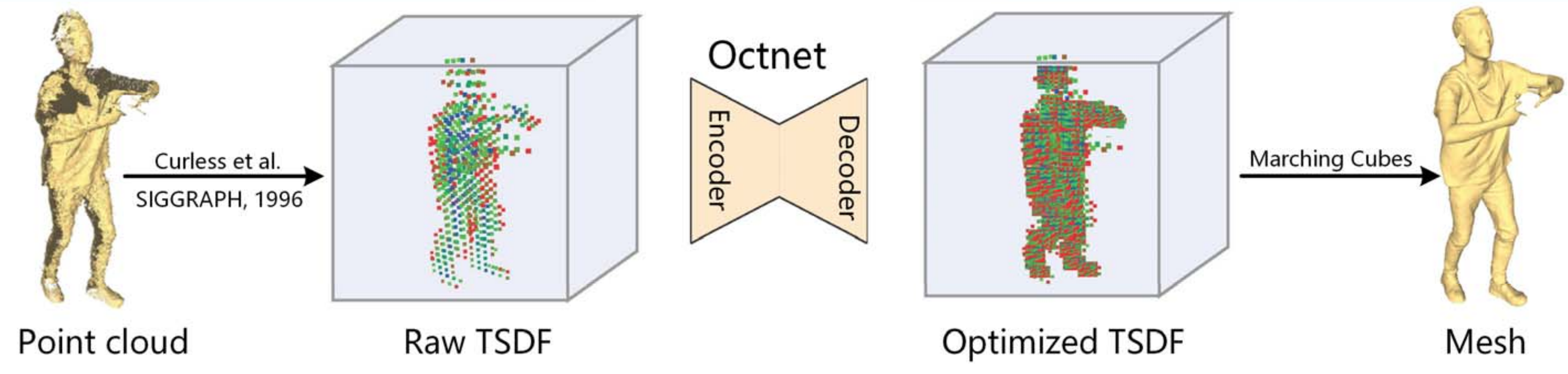


- Yiyi Liao et al. CVPR2018

03 Learning-based methods on girds

➢ OctNetFusion: Octree

- ➢ Converting point cloud into TSDF in an octree
- ➢ Appling convolution, pooling and unpooling operations defined in Octnet to the octree
- ➢ Predicting optimized TSDF



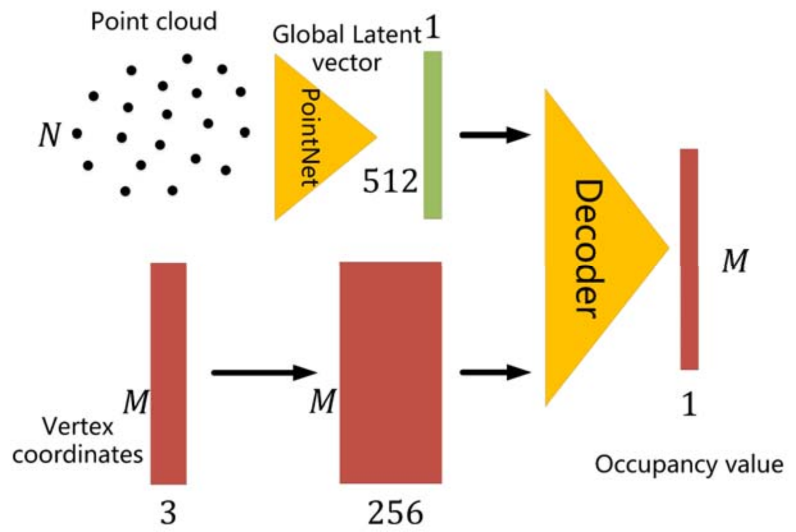
■ Gernot Riegler et al. 3DV2017

04 Learning-based methods for continuous function

➢ Occupancy networks (ONet)

- ➢ Encoding point cloud into **a latent vector**
- ➢ Learning a **occupancy probability** function for 3D locations, trained by **binary classification**

$$f_\theta: \mathbb{R}^3 \times \mathcal{X} \rightarrow [0,1]$$

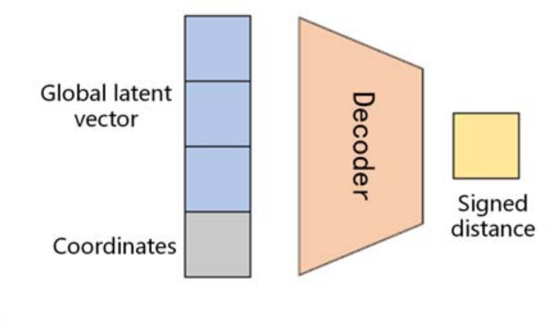


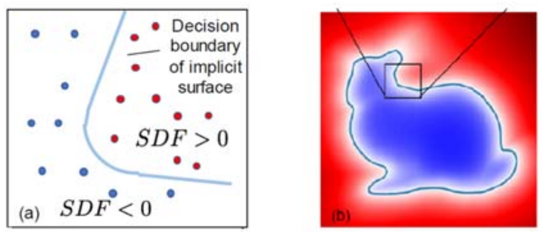
■ Lars Mescheder et al. CVPR2019

➢ Deepsdf

- ➢ Learning **signed distance function** for 3D locations

$$SDF(\mathbf{x}) = s: \mathbf{x} \in \mathbb{R}^3, s \in \mathbb{R}$$





■ Jeong Joon Park et al. CVPR2019

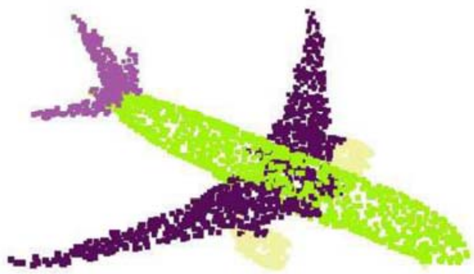
SSRNet

01 Problems and challenges

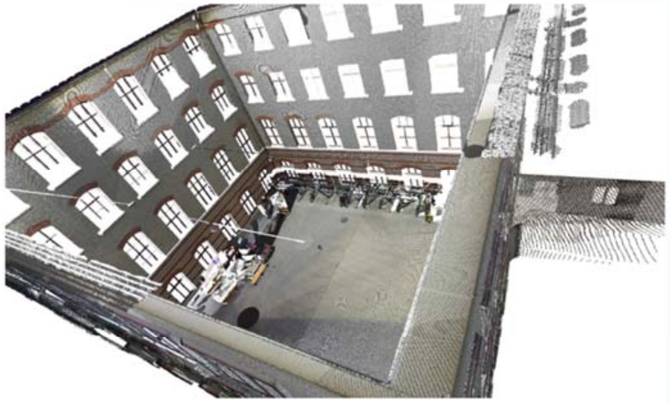
➢ Scalability

- ➢ Voxel network: complexity issue
- ➢ Octree network: dependent on octree structures, complexity and efficiency issue
- ➢ ONet & Deepsdf: global latent vector

- ➢ Traditional methods，Poisson and Delaunay tetrahedrons+Graph cuts



Small object

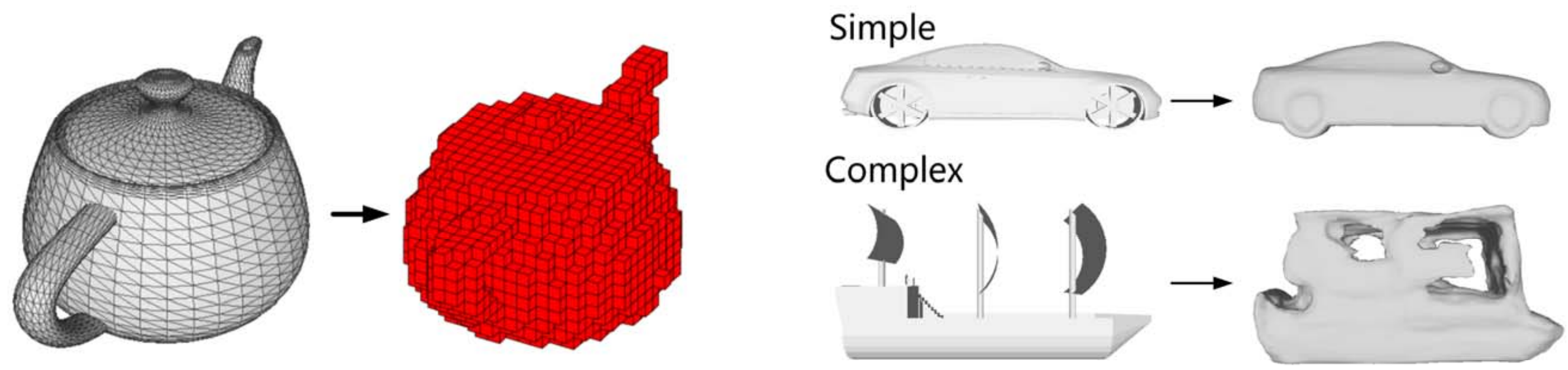


Large-scale scene

01 Problems and challenges

➢ Geometry details

- ➢ Voxel network: information loss in voxelization
- ➢ ONet & Deepsdf: information loss in short latent vector, no local context

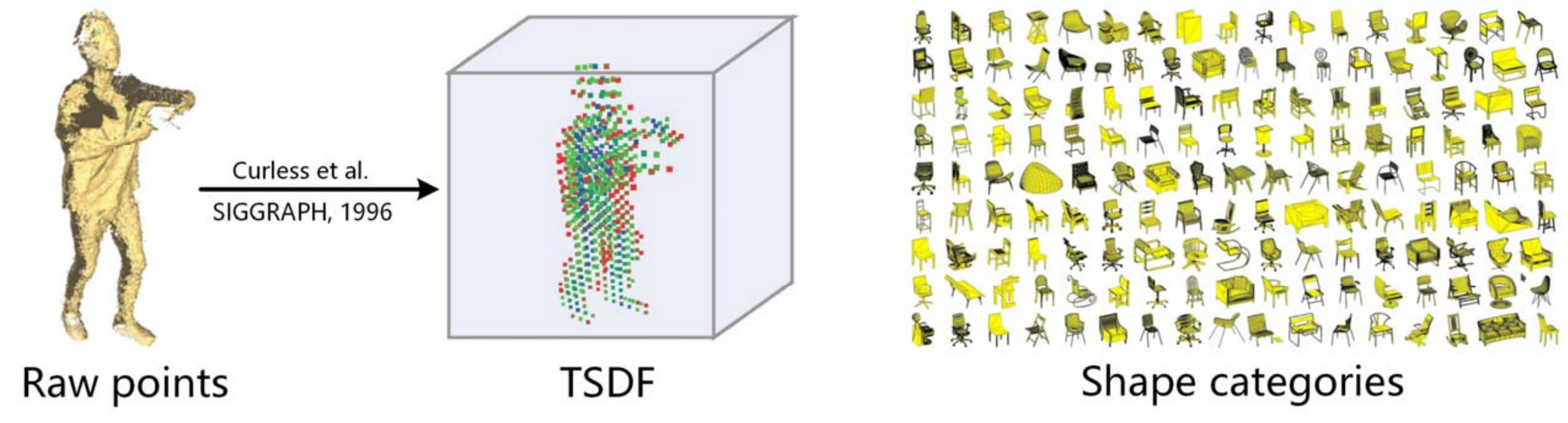


Voxelization

Latent vector

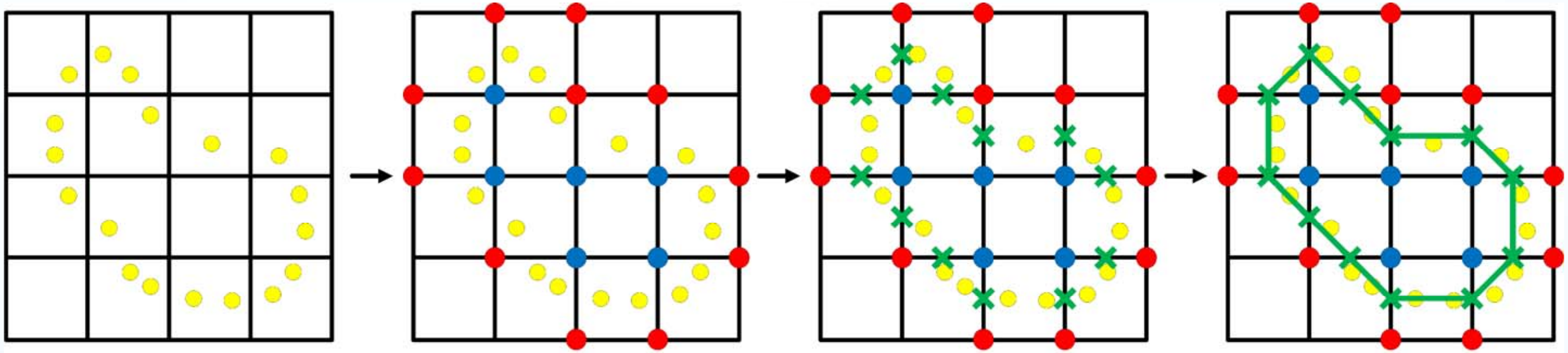
01 Problems and challenges

- Training and generalization
  - Voxel & Octree network: input **TSDF** depends on **traditional** methods
  - Need **large portion** of training data
  - ONet & Deepsdf: dependent on **absolute coordinate**, hard to generalize to **unseen** shape category



02 SSRNet: Scalable 3D Surface Reconstruction Network

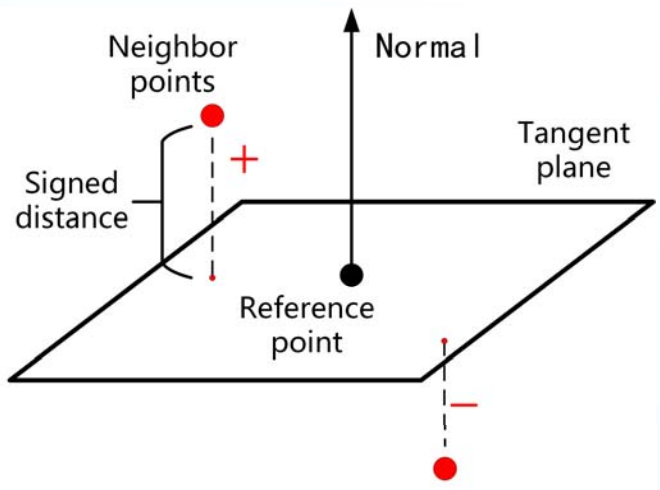
- **Binary classification**
  - Classifying octree vertices as in front or at back of the implicit surface and extracting surface by Marching Cubes
- **Important**
  - How to design octree vertices features for accurate and scalable predication?



(a) (b) (c) (d)

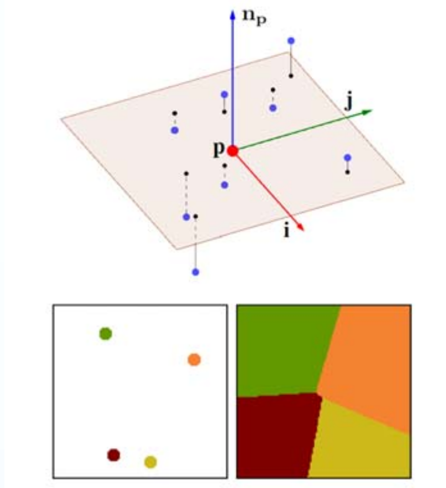
03 Geometry-aware Vertex Feature: Ideas

- **From geometric method**
  - **Idea: signed distances from normal directions are critical, local 3D-2D fitting**
  - **Drawback: global fitting non-scalable, local fitting functions too simple**



Signed distance

- **From tangent convolution**
  - **Idea: local 3D-2D learning for point cloud segmentation**
  - **Drawback: ambiguous normal directions from PCA**

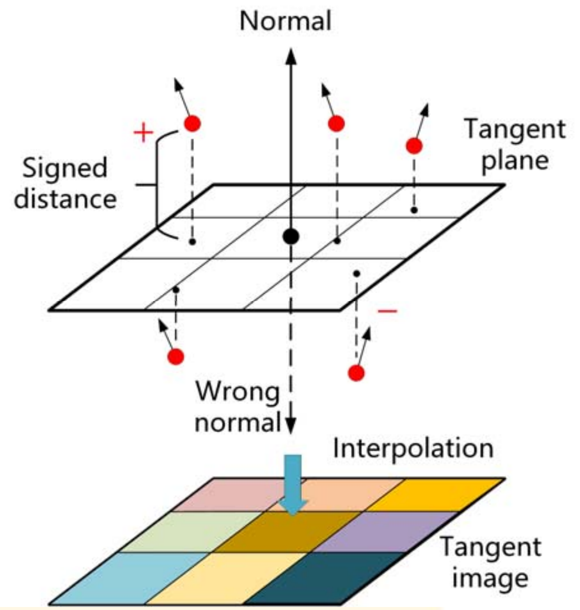


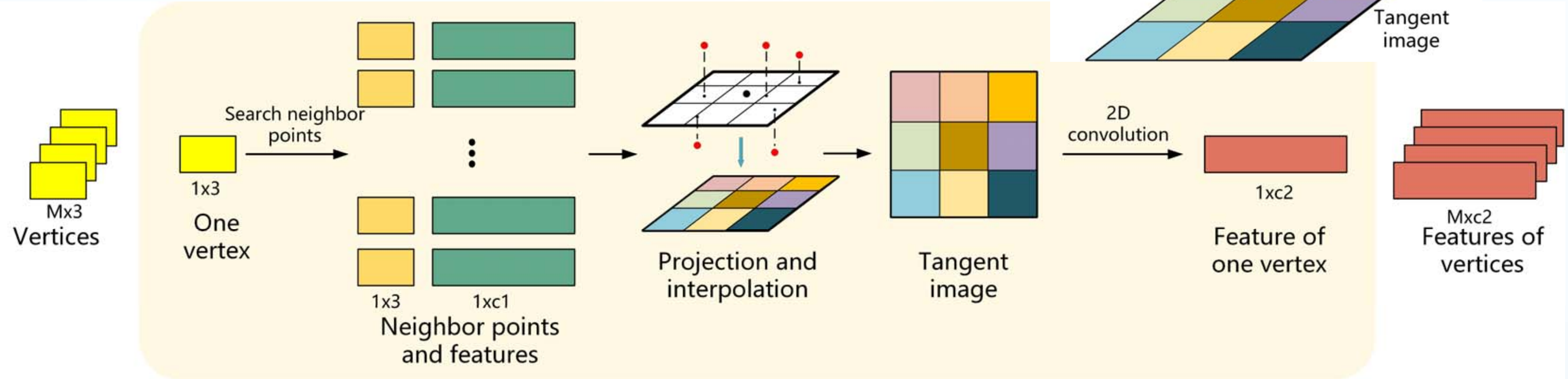
Tangent convolution

- **Maxim Tatarchenko et al. CVPR2018**

03 Geometry-aware Vertex Feature: Design

- Construction
  - Constraining normal of tangent plane by **average neighbor normal**
  - Constructing **signed distance, normal** and **other feature images** for each vertex through **local projection**
  - Applying **2D convolution** on 2D images

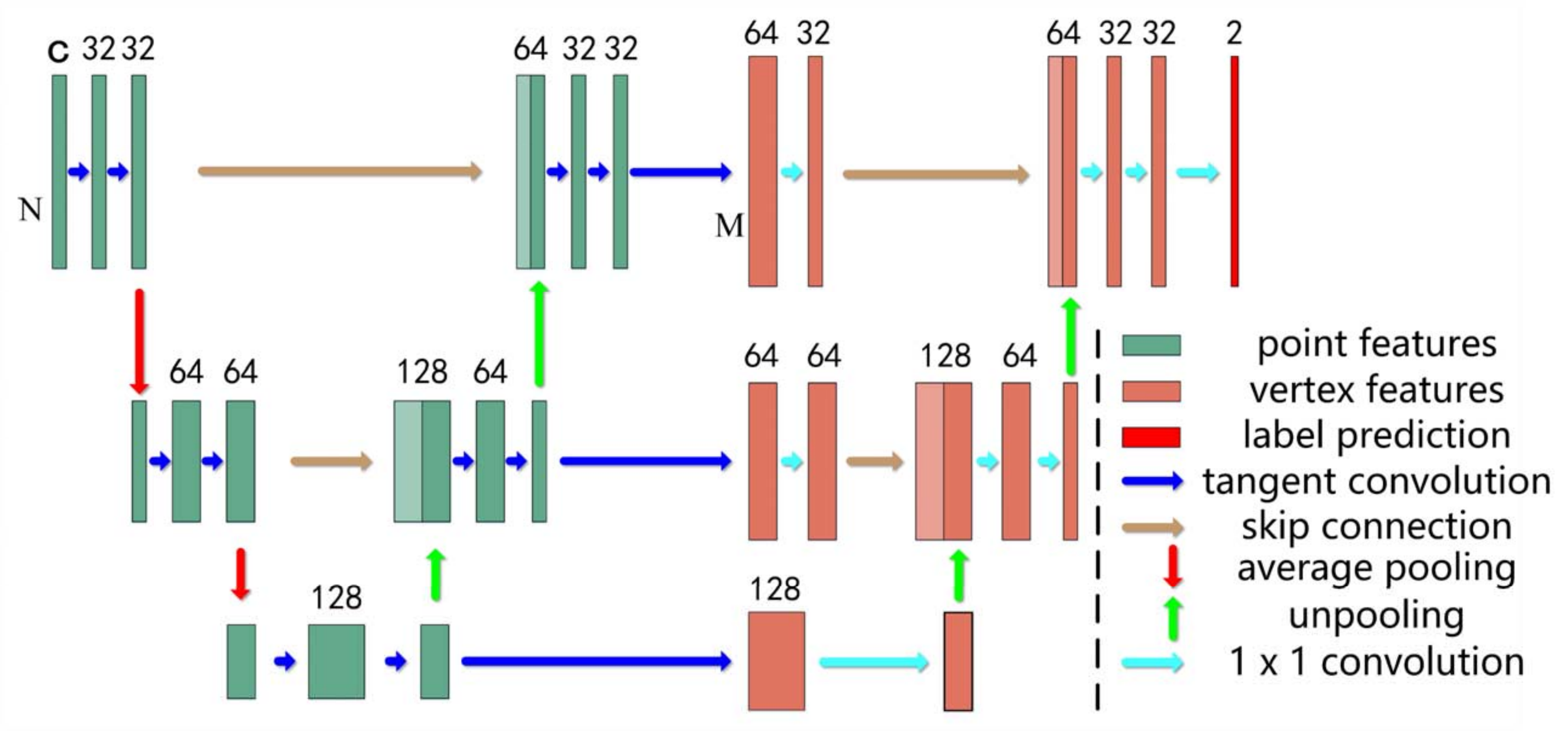




03 Geometry-aware Vertex Feature: Design

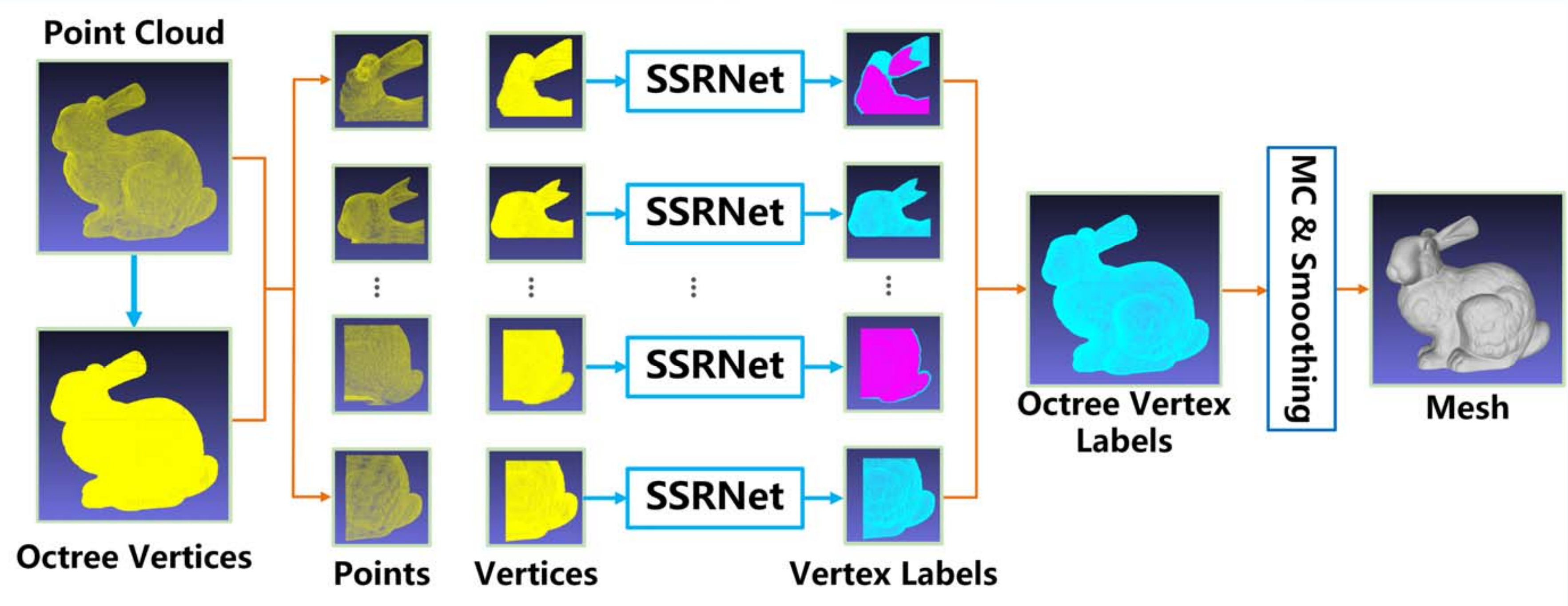
- Advantages
  - Directly learning local surface feature, more accurate
  - Performed in a fixed-size local region by ball query for neighbor points, allowing subdividing input data, more scalable.
  - Independent to octree structure
  - Indices for tangent images can be precomputed, making the network of more efficient
  - Parameters of 2D convolution are shared, making the network less complex

04 SSRNet Architecture



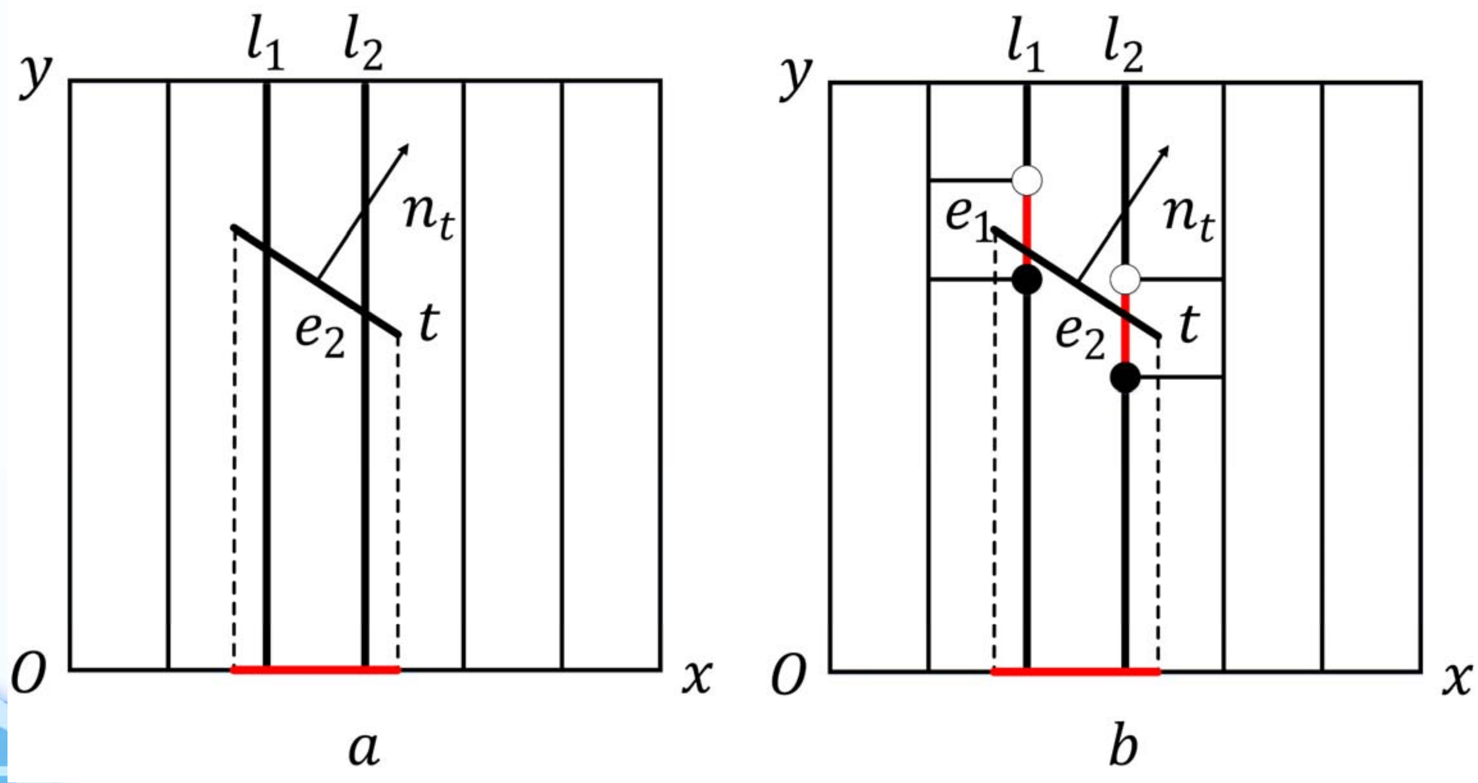
05 SSRNet Pipeline

➢ **Subdividing** points and vertices, **predicting** labels using SSRNet and **extracting** surface by Marching Cubes



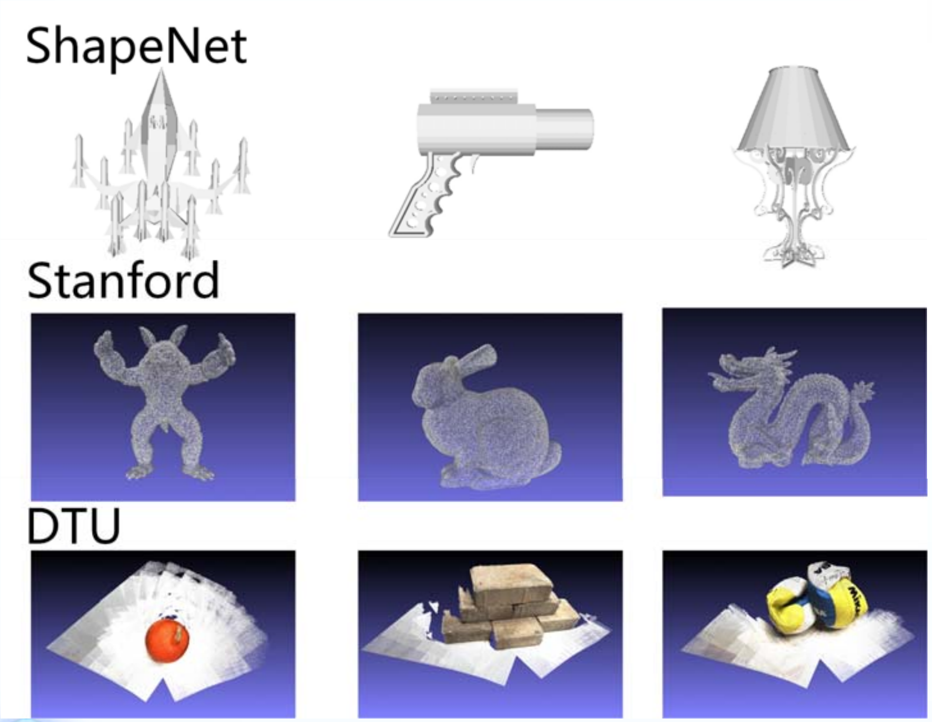
06 Training data generation

- Projecting triangles and record intersections with grid lines
- Labeling vertices on the octree edges containing intersections
- Labeling other vertices through nearest neighbor search



07 Experiments

➢ Datasets



■ Evaluation Metrics

➢ **ShapeNet**

- **IoU** (higher is better),
- **Chamfer-L_1 distance** (lower is better)
- **Normal Consistency** (NC， higher is better)
- **Classification accuracy**

➢ **DTU**

- **DTU Accuracy**
- **DTU Completeness**
- **Chamfer distance** (CD) (All lower is better)
- **Classification accuracy**

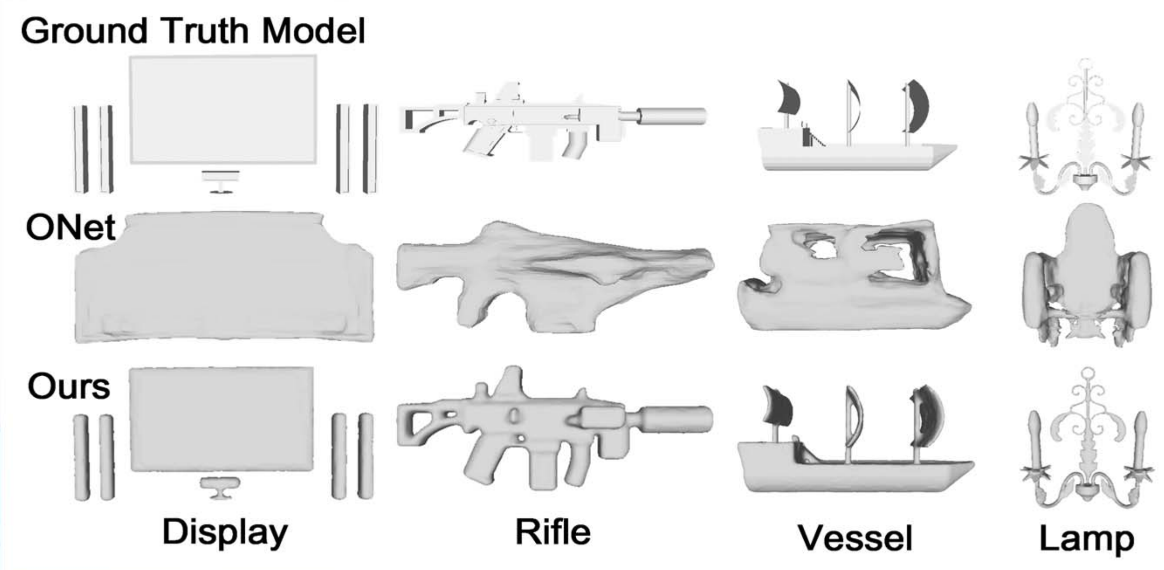
➢ **Stanford**

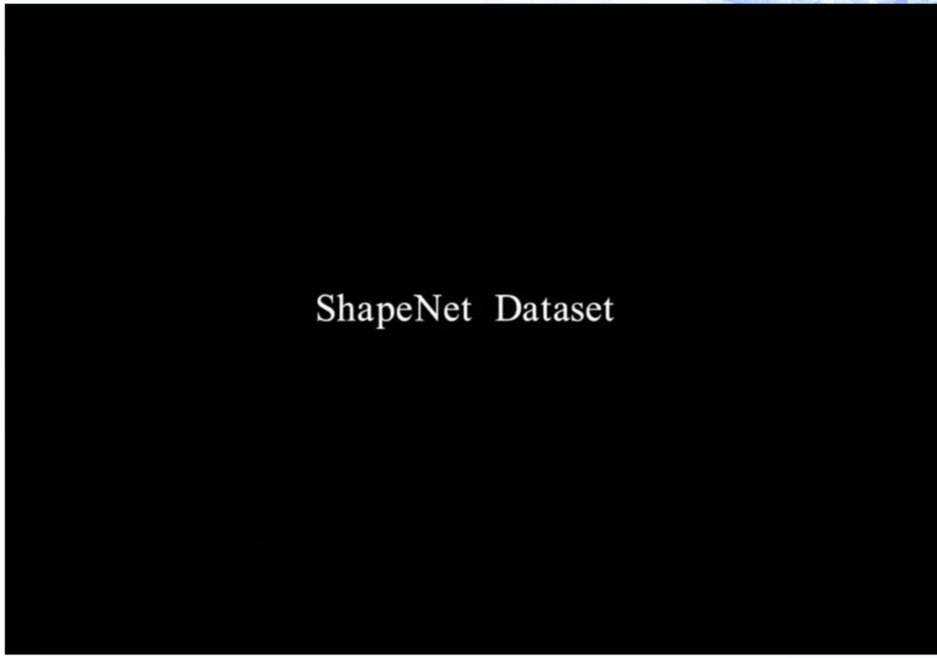
- **Chamfer distance**
- **Classification accuracy**

07 Experiments on ShapeNet

Comparison with learning-based 3D methods:

- 3D-R2N2, PSGN, DMC and Onet
- Classification accuracy is 97.6
- Training data: Onet 4/5, ours 4/50
- Parameters: Onet 13.4M, ours 0.49M



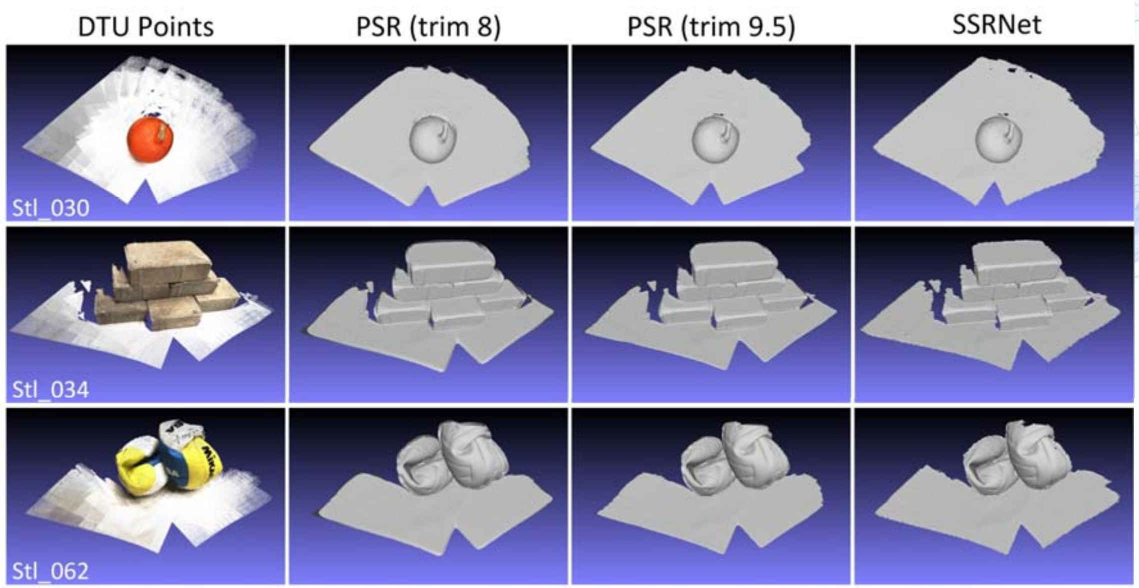


	IoU	Chamfer-L_1	NC
3D-R2N2	0.565	0.169	0.719
PSGN	—	0.202	—
DMC	0.647	0.117	0.848
ONet	0.778	0.079	0.895
	+17.9%	-0.055	+0.072
SSRNet	**0.957**	**0.024**	**0.967**

- 3D-R2N2, Christopher et al. ECCV2016
- DMC, Yiyi Liao, et al. CVPR2018
- PSGN, Haoqiang Fan et al. CVPR2017
- ONet, Lars Mescheder et al. CVPR2019

07 Experiments on DTU

- Results on DTU scans of **large scales.**
- Each point cloud has **millions** of points.
- SSRNet achieves **comparable quality** with SOTA geometric method **PSR.**
- **6/100** for training
- Classification accuracy is **95.7**



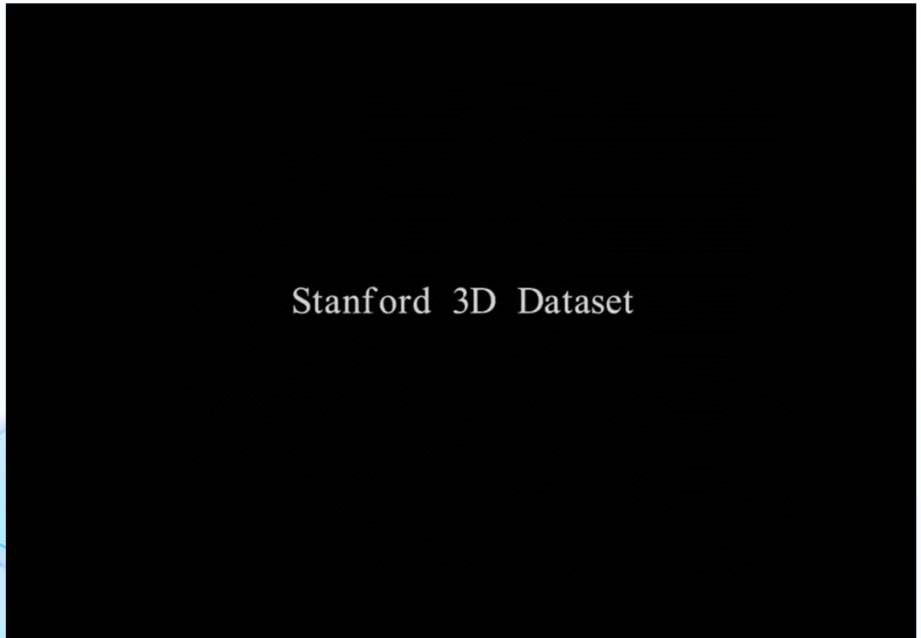
Method	DA		DC		CD	
	Mean	Var.	Mean	Var.	Mean	RMS
PSR(trim 8)	0.473	1.33	0.327	0.220	3.16	12.5
PSR(trim 9.5)	0.330	0.441	0.345	0.438	**1.17**	4.49
Ours	**0.321**	**0.285**	**0.304**	**0.0888**	1.46	**4.42**

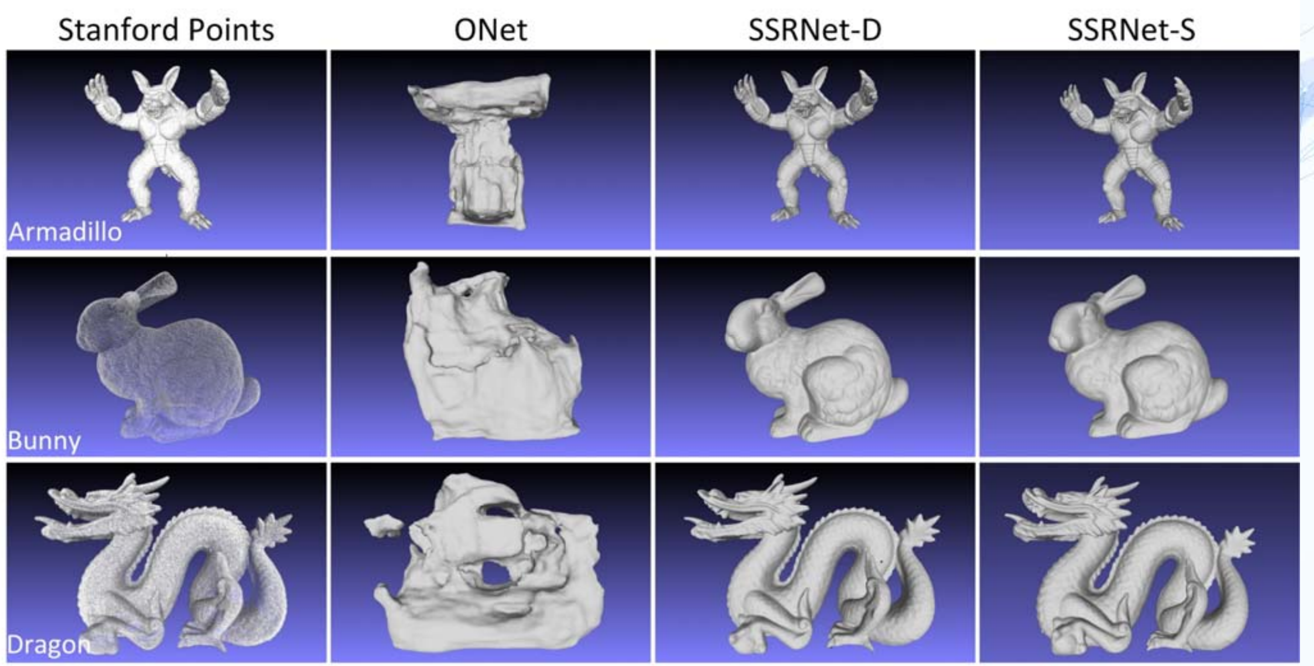
- PSR, Michael Kazhdan et al. ESGP2006, ToG2013

07 Experiments on Stanford for generalization

Applying two trained SSRNet models to Stanford dataset :

- **SSRNet-D** trained on DTU
- **SSRNet-S** trained on ShapeNet
- **Generalization Capability Test**





	Accuracy		CD Mean		CD RMS	
Dataset	ShapeNet-Model	DTU-Model	Ours	PSR	Ours	PSR
Stanford	98.1	98.2	4.21	**3.74**	**9.75**	13.3

07 Experiments about efficiency

Time efficiency on point clouds with millions of points. SSRNet can make use of multiple GPUs for acceleration.

Number	Armadillo	Bunny	Dragon	stl_030	stl_034	stl_062
Point /M	2.16	0.361	1.71	2.43	2.01	2.19
Vertex /M	3.29	3.62	3.07	1.07	0.766	0.922
Triangle /M	1.18	1.52	1.16	0.42	0.31	0.36
Batch	109	73	77	59	88	86

Time /s	Armadillo	Bunny	Dragon	stl_030	stl_034	stl_062
Prep	19.4	8.86	16.6	10.9	9.34	9.99
Pred (1 GPU)	133	82.2	110.5	63.5	84.5	87.9
Pred (4 GPUs)	**50.5**	**27.3**	**38.2**	**30.4**	**31.1**	**34.3**
Total (1 GPU)	153	92.0	128	75.4	94.5	98.6
Total (4 GPUs)	**70.6**	**37.1**	**55.7**	**42.3**	**41.1**	**45.0**

07 SSRNet vs ONet (Lars Mescheder et al. CVPR2019)

- **Common:** Binary classification for octree vertices
- **Differences**

SSRNet	ONet
Learning local geometry feature	Encoding global feature
Capturing local geometry details	Losing information due to global feature
Independent on global coordinates	Dependent on global coordinates
Independent on shape categories	Dependent on shape categories
Allow dividing large input points	Actually downsampling points
Less training data (4K in ShapNet)	More training data (40K in ShapNet, 4/5)
Generalizing well	Hard to generalize to unseen categories
0.49M parameters	13.4M parameters
Open&Close surface	Close surface

08 Contributions

- **Scalability**——allowing subdividing input data
- **Local geometry-aware feature**——high accuracy and quality
- **Training**——Less training data Independent on shape
- **Generalization**——Model on one data directly used to another
- **High efficiency**——Easy to parallel

Thank you！

Q&A