Learning-based Sampling over 3D Point Clouds

presented by

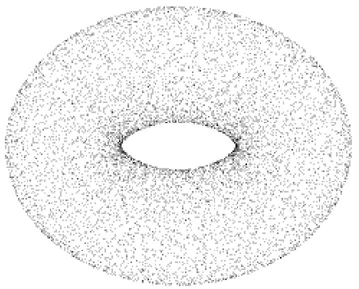
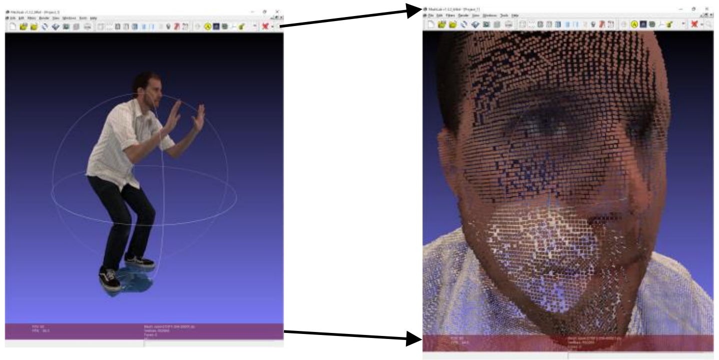
Dr. HOU Junhui, *Assistant Professor*
Department of Computer Science, City University of Hong Kong
Email: jh.hou@cityu.edu.hk
https://sites.google.com/site/junhuihoushomepage/home

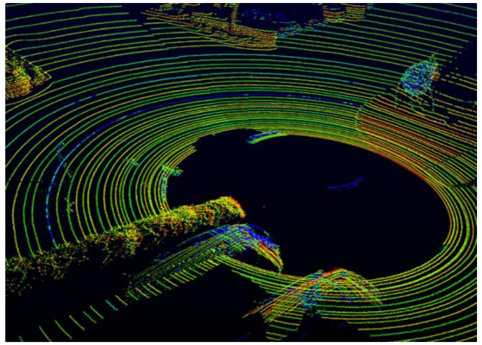
Acknowledgements: my PhD student, Miss Yue Qian
the collaborator, Prof. Ying He, NTU, Sg

References
Y. Qian, J. Hou, *et al.* PUGeo-Net: A Geometry-centric Network for 3D Point Cloud Up-sampling, ECCV, 2020, 1-17
Y. Qian, J. Hou, *et al.* MOPS-Net: A Matrix Optimization-driven Network for Task-Oriented 3D Point Cloud Down-sampling, https://arxiv.org/abs/2005.00383

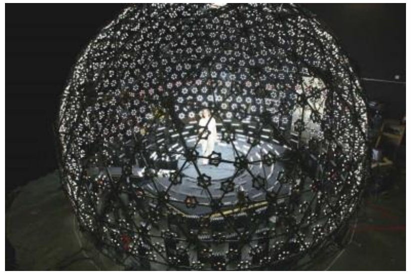
3D Point Cloud Data

- **Unstructured set of 3D point samples**
  - ➢Each point consists of geometry information (x, y, z) and optional attributes , e.g., color (r, g, b) and normal (n_x, n_y, n_z)

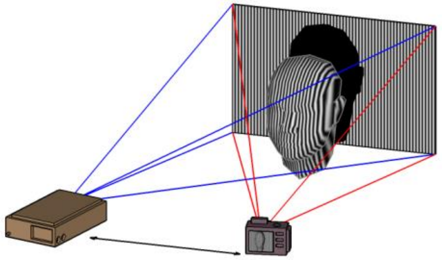


- **Acquisition devices**



Multiview-based

Laser-based

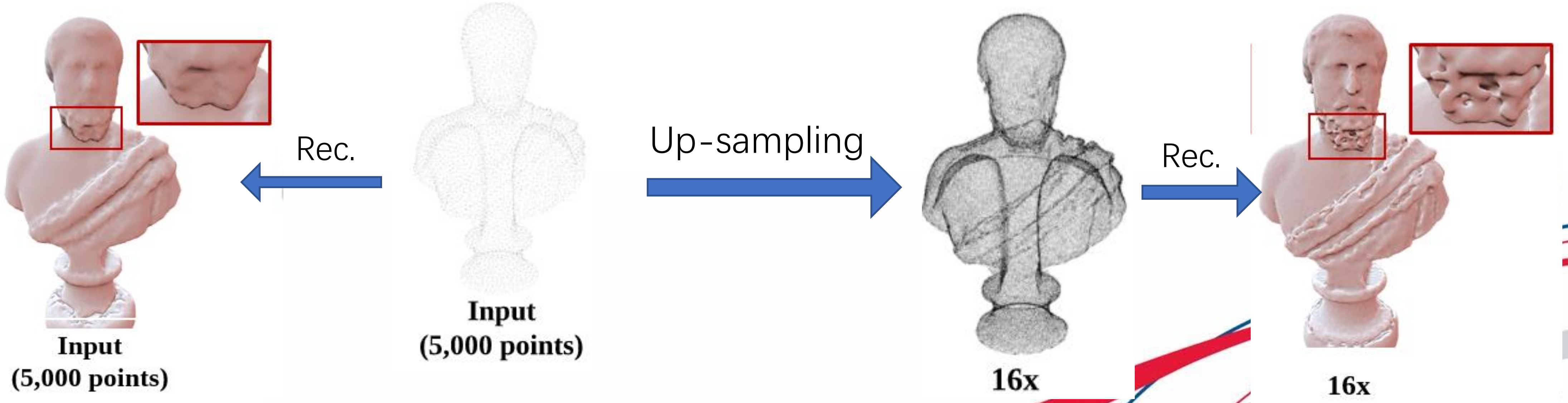


Structured light-based

Realsense

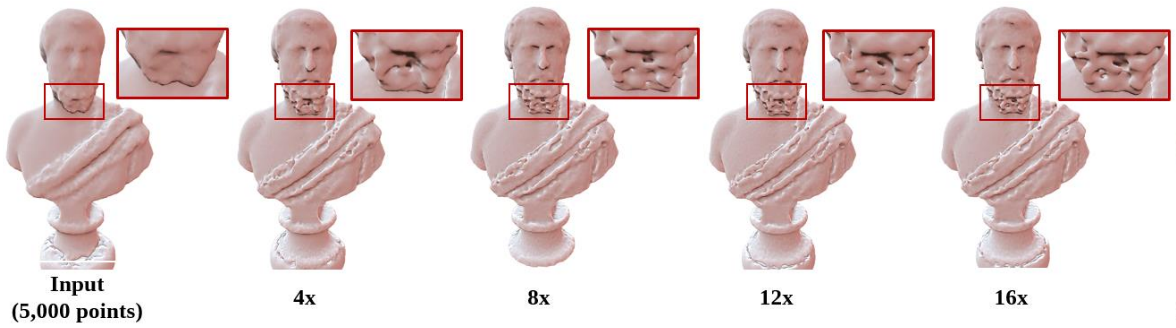
Up-sampling over 3D Point Clouds

- **Given a sparse point cloud with N points, generate a dense point cloud with M points ($M > N$) via a typical computational method to represent objects/scenes.**
  - It is costly and time-consuming to obtain such highly detailed data from hardware.
  - High resolution point clouds are beneficial to subsequent applications, e.g. surface reconstruction, object detection.



Point Cloud Up-sampling vs. Image Up-sampling

香港城市大學
City University of Hong Kong



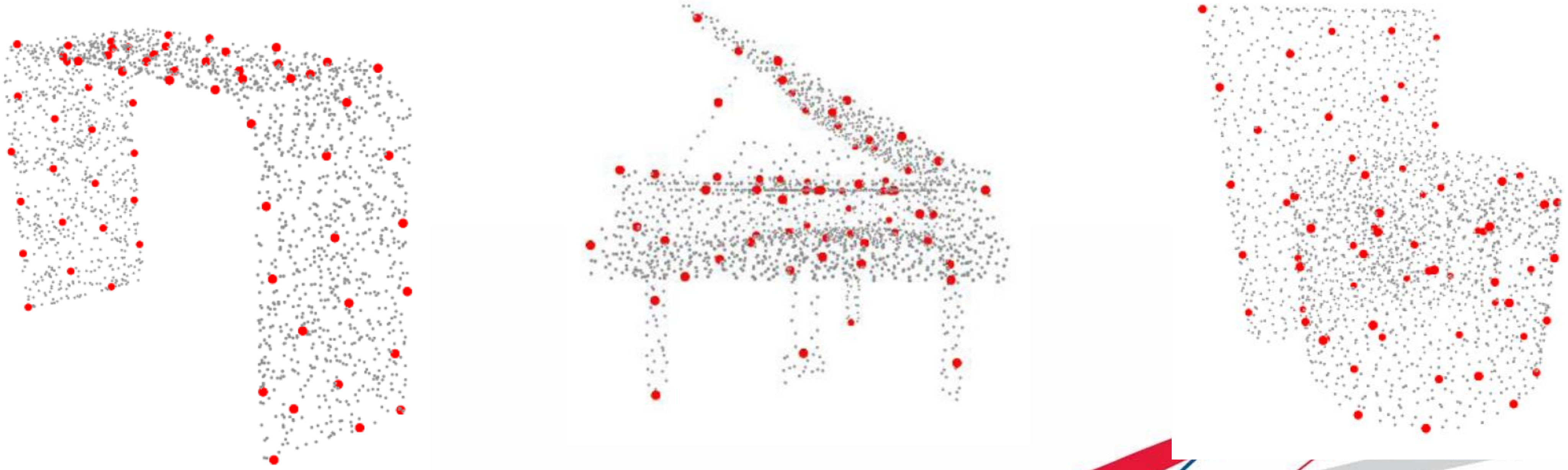
- **3D geometry information**
- **Irregular and unordered (non-Euclidean space)**
- **How to design feature/point expansion?**

More Challenging

- **Illumination (color) information**
- **Regular structure (Euclidean space)**
- **Deconvolution/transposed layer to expand features**

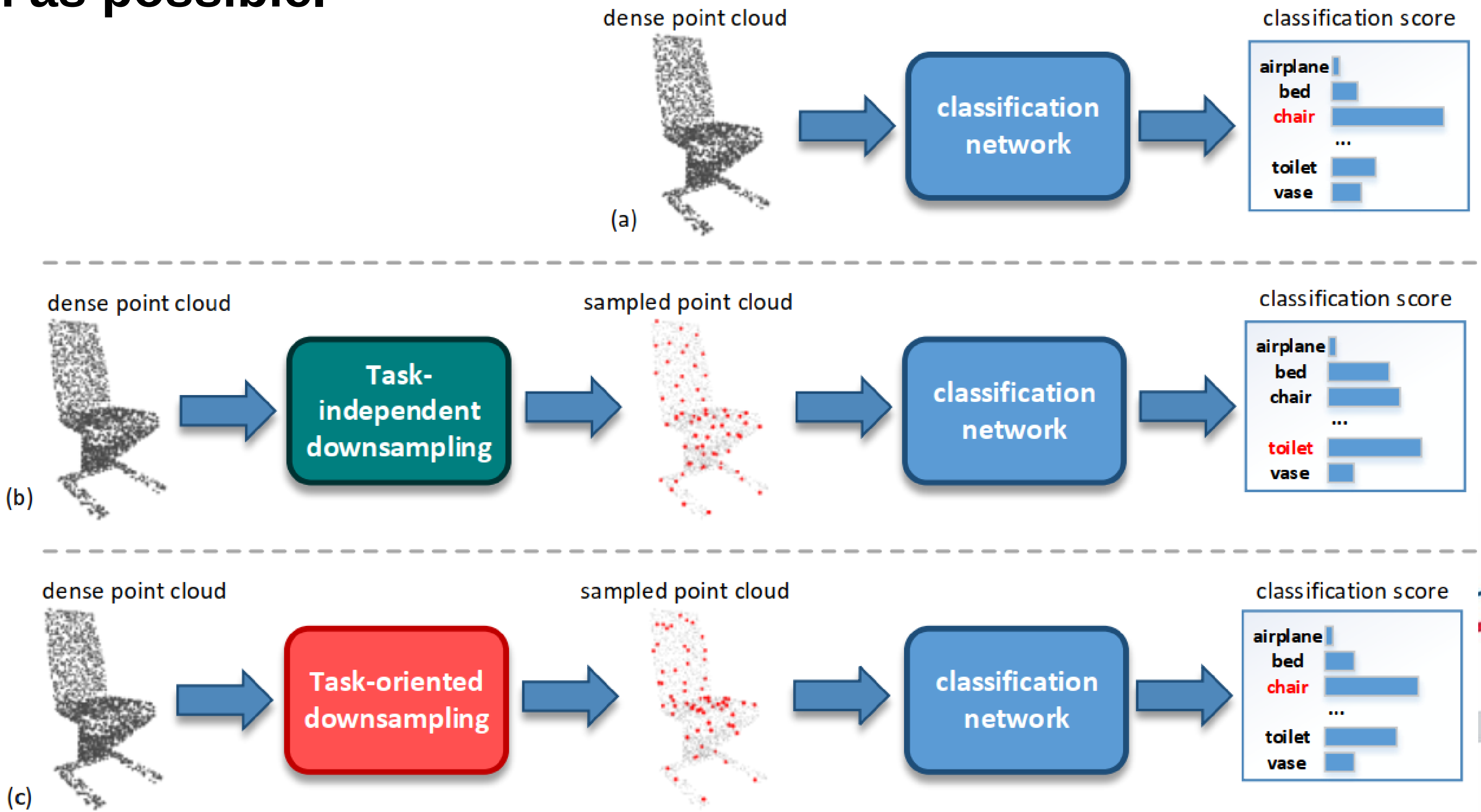
Down-sampling over 3D Point Clouds

- **Given a point cloud with n points, generate a sparse point cloud with m points ($m < n$) distributed in the same space to represent the original object/scene.**
  - ➢ Reduce information redundancy, thus more efficient running time, saving storage space and transmission bandwidth.



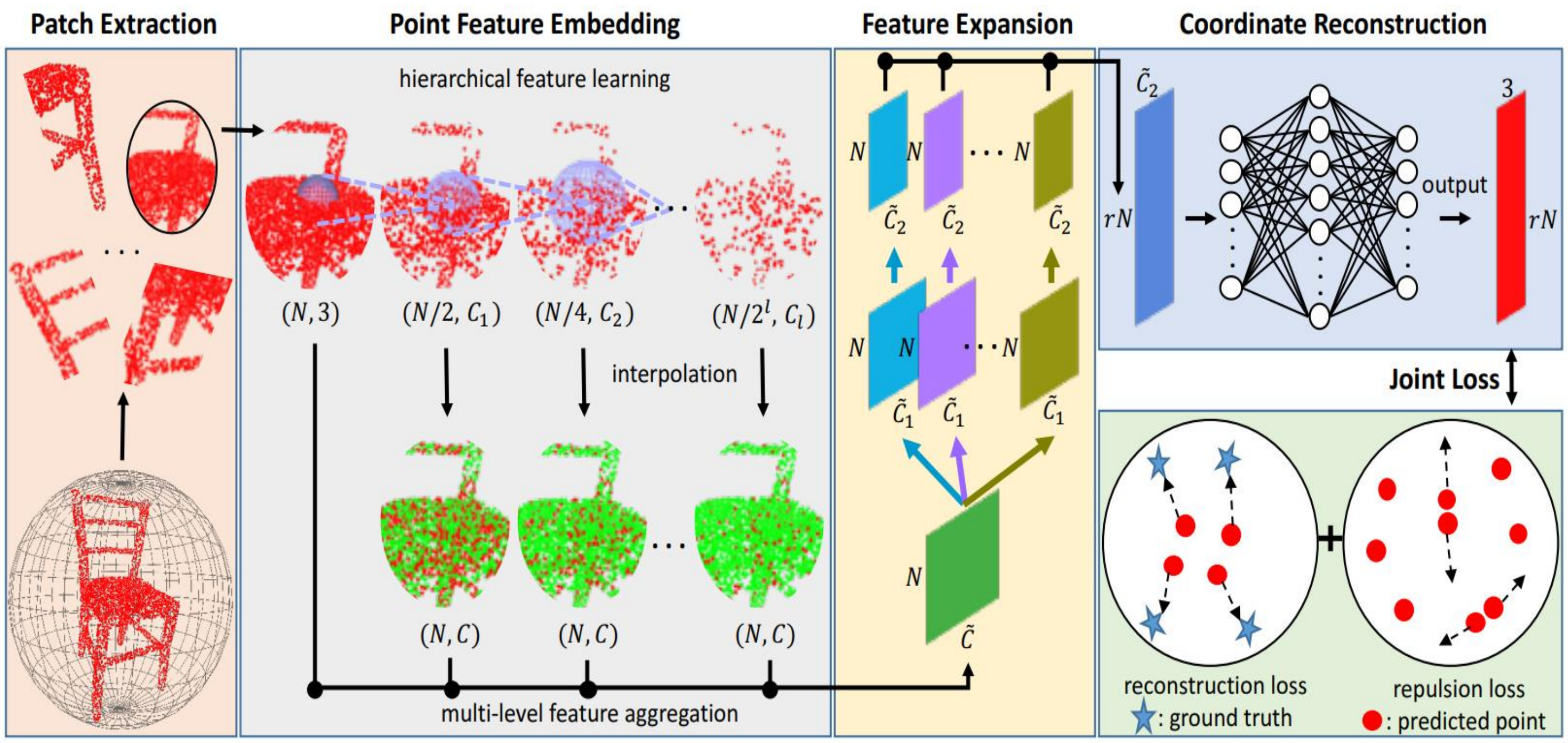
Down-sampling over 3D Point Clouds

- **Our goal: task-oriented point cloud down-sampling, i.e., the down-sampled sparse point clouds will maintain the task performance as much as possible.**



Related Works

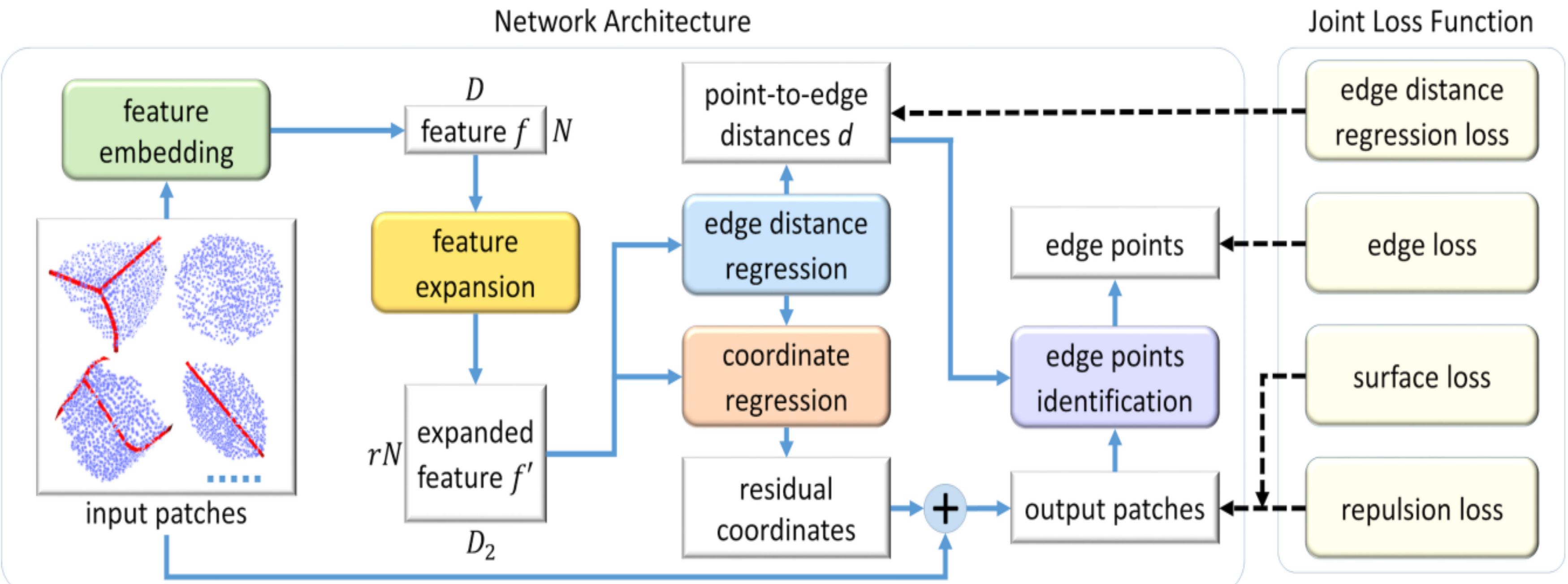
- **Deep learning-based up-sampling methods: PU-Net**
  - ➢Expand features using separated neural network branches.



L. Yu, *et al.*, PU-Net: Point Cloud Upsampling Network, *in Proc. CVPR*, 2018

Related Works

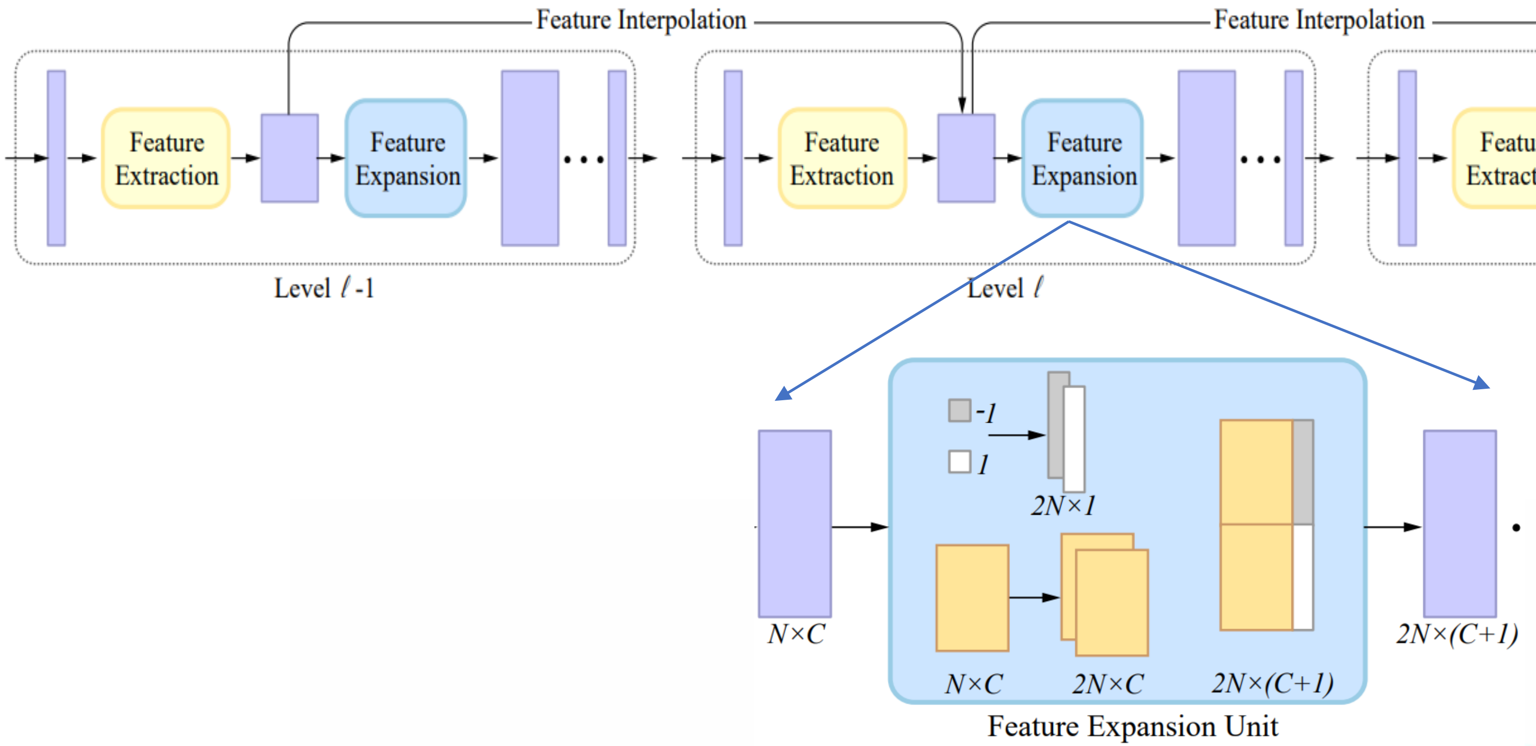
- **Deep learning-based up-sampling methods: EC-Net**
  - ➢Based on PU-Net, restoring sharp features with additional edge and surface annotations
  - ➢Require additional annotations for edges and surfaces, which are costly and infeasible for data with complex geometry



L. Yu, *et al.* EC-Net: an edge-aware point set consolidation network, *in Proc. ECCV.* 2018.

Related Works

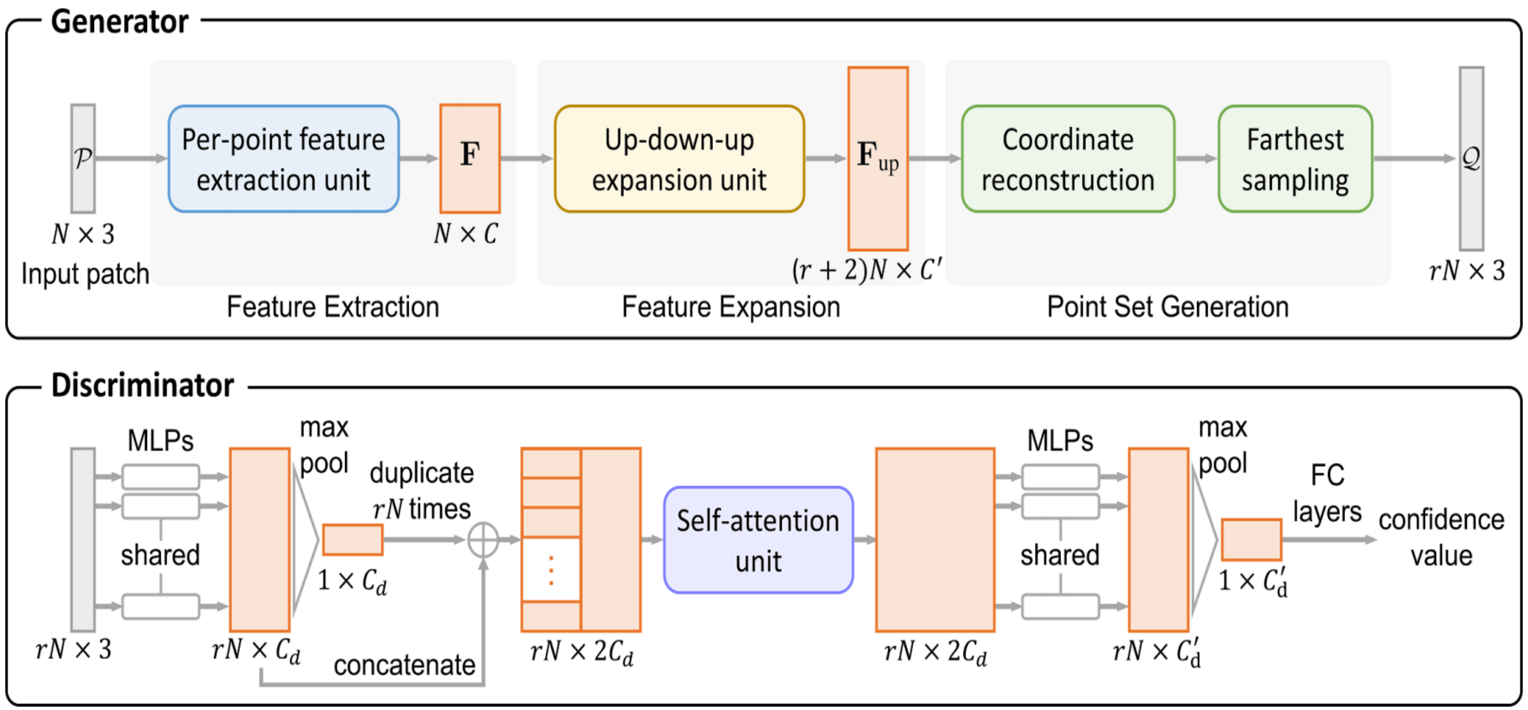
- **Deep learning-based up-sampling methods: MPU**
  - ➢A cascade structure that progressively up-samples the input 2x at each level.
  - ➢Append +1/-1 to feature to separate features



Y. Wang, et al. Patch-based progressive 3d point set upsampling, *in Proc. CVPR*, 2019.

Related Works

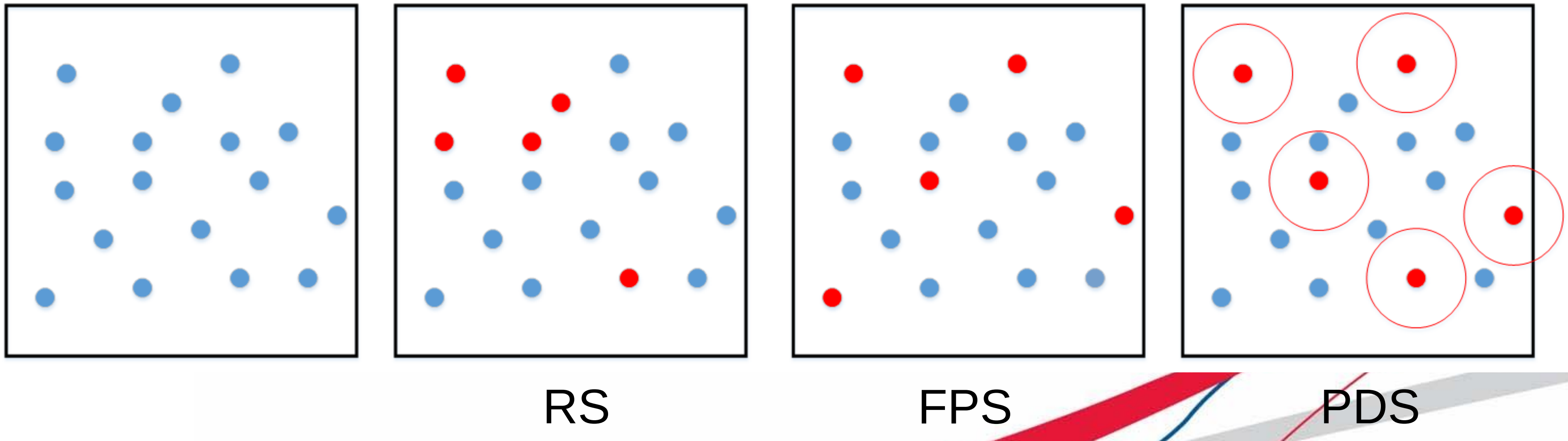
- **Deep learning-based up-sampling methods: PU-GAN**
  - Introduce an additional discriminator (GAN structure) to improve the generator's performance.
  - Extend the 1D code assign in MPU to the 2D code assign for feature expansion.



R. Li, et al., Pu-gan: a point cloud upsampling adversarial network *In Proc. ICCV*, 2019

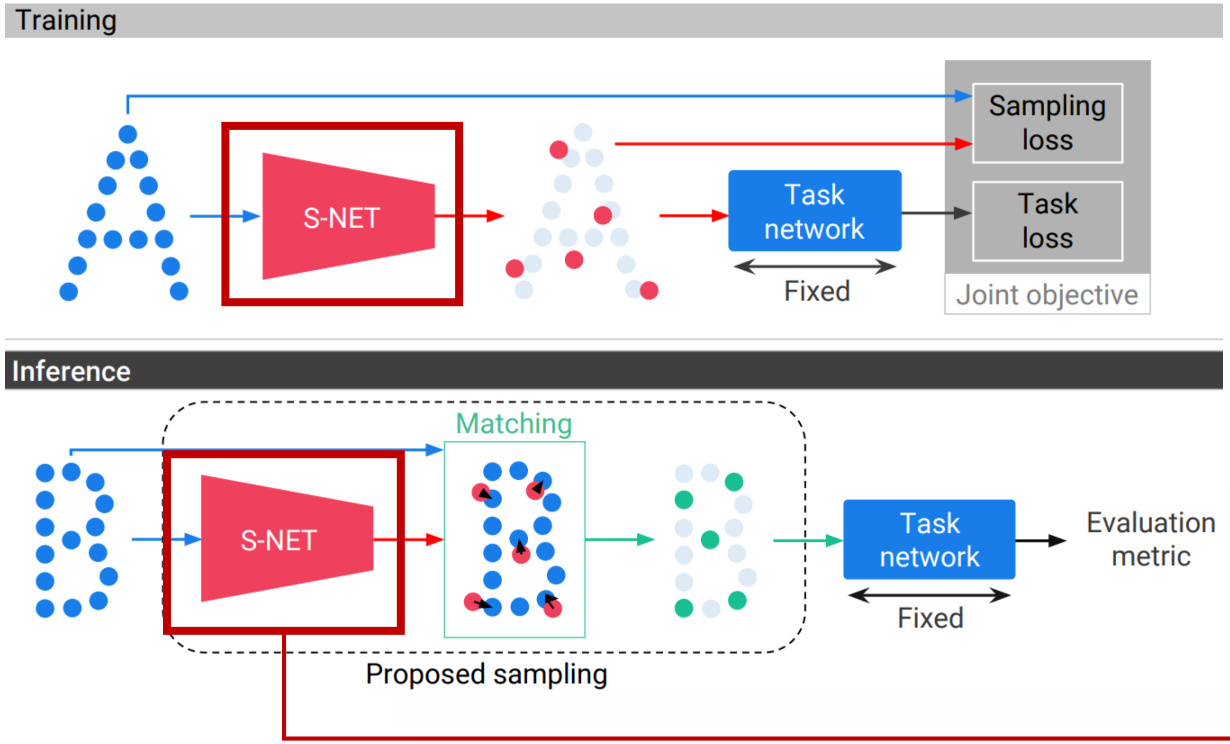
Related Works

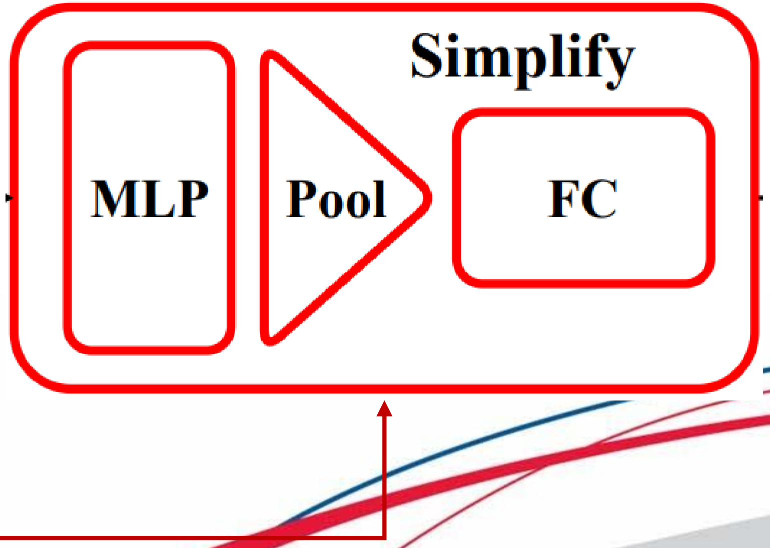
- **Classic down-sampling methods**
  - ➢ Random sampling (RS)
  - ➢ Farthest point sampling (FPS)
  - ➢ Poisson disk sampling (PDS)
  - ➢ The down-sampled point cloud is a subset of the dense one, which can preserve geometry well to some extent but are completely independent of downstream applications. Thus, the down-sampled point clouds may degrade the performance of the subsequent applications severely.



Related Works

- **Deep learning-based down-sampling methods: S-Net**
  - ➢ Task-oriented point cloud down-sampling supervised by a joint loss.
  - ➢ Trivially generate sparse points directly from the global feature without sufficient consideration of the local structure.

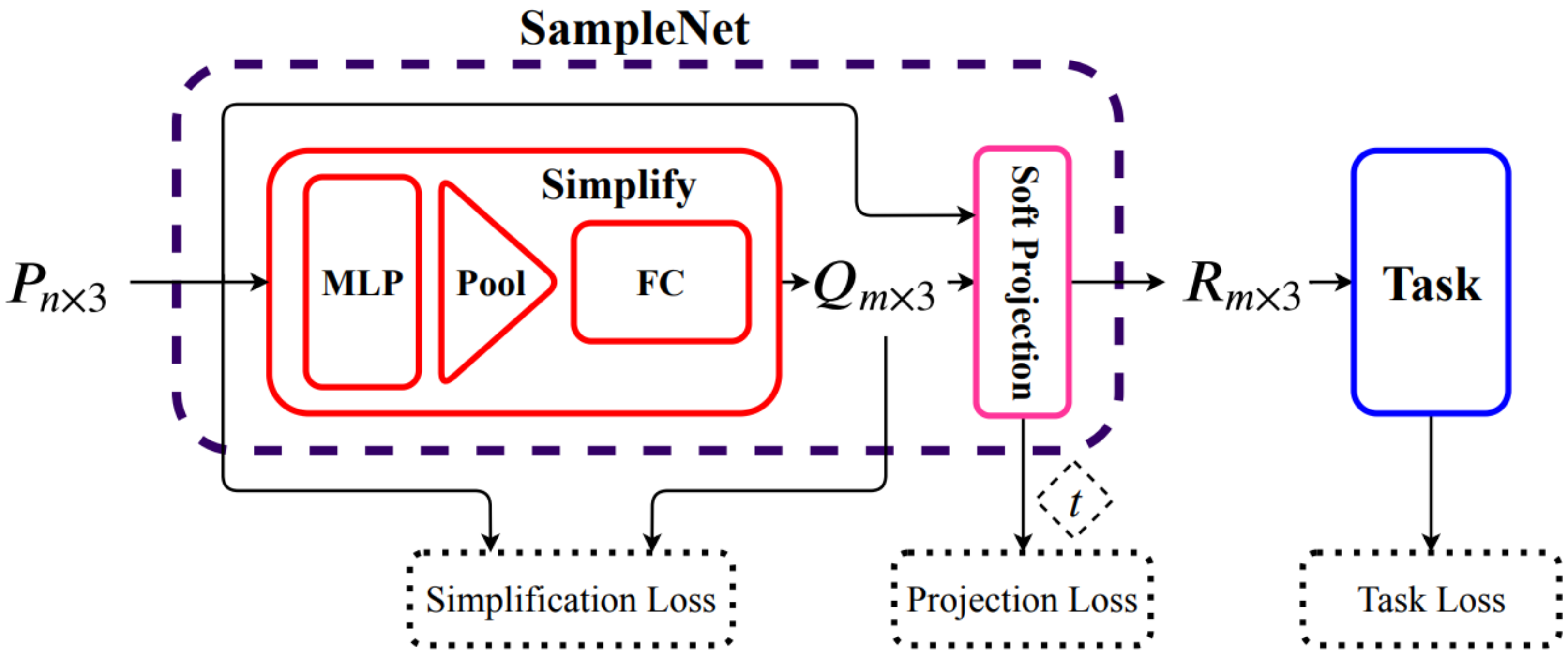




O. Dovrat, *et al*. “Learning to sample." *In Proc. CVPR*, 2019

Related Works

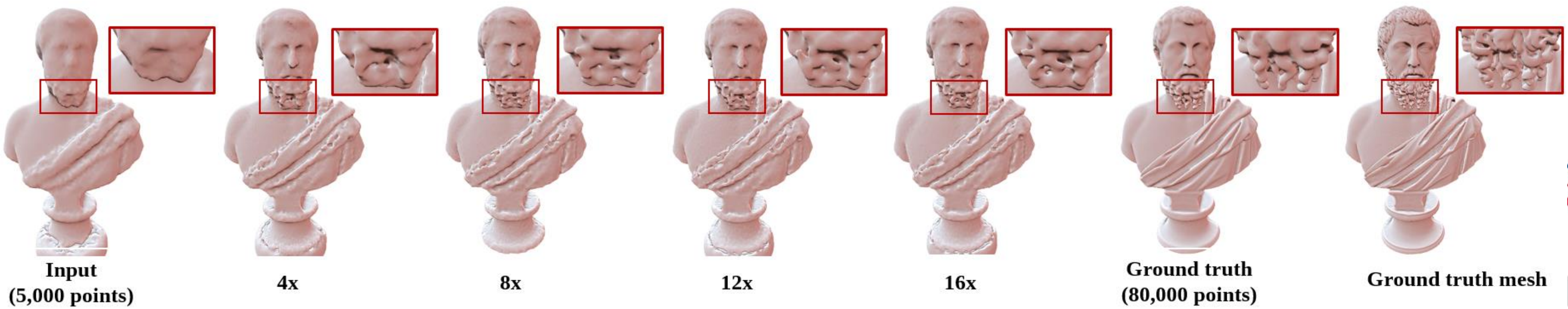
- **Deep learning-based down-sampling methods: Sample-Net**
  - Extension of S-Net, introduce an additional post-processing module (soft projection) to deal with non-differentiable sampling operation in S-Net.
  - Still suffer from the drawback of S-Net, i.e. the ignorance of the spatial correlation



O. Dovrat, *et al*. "SampleNet: Differentiable Point Cloud Sampling ." *In Proc. CVPR*, 2020

Proposed Up-sampling Method: PUGeo-Net

- **PUGeo-Net: A Geometry-centric Network for 3D Point Cloud Up-sampling**
  - **Geometry-centric**, link **differential geometry** and **deep learning** elegantly. Provide quantitative verification to confirm the interpretation.
  - Jointly generate **coordinates** and **normal**, which will be beneficial to downstream applications, e.g. surface reconstruction and shape analysis.
  - Outperform state-of-the-art methods for all metrics.
  - Robust to **noisy** and **non-uniform** input, e.g. real scanned LiDAR data.



Proposed Up-sampling Method: PUGeo-Net

- **Theoretical foundation of PUGeo-Net**
  - The Fundamental Theorem of the Local Theory of Surfaces states the local neighborhood of a point on a regular surface can be completely determined by the **first and second fundamental forms**, unique up to rigid motion

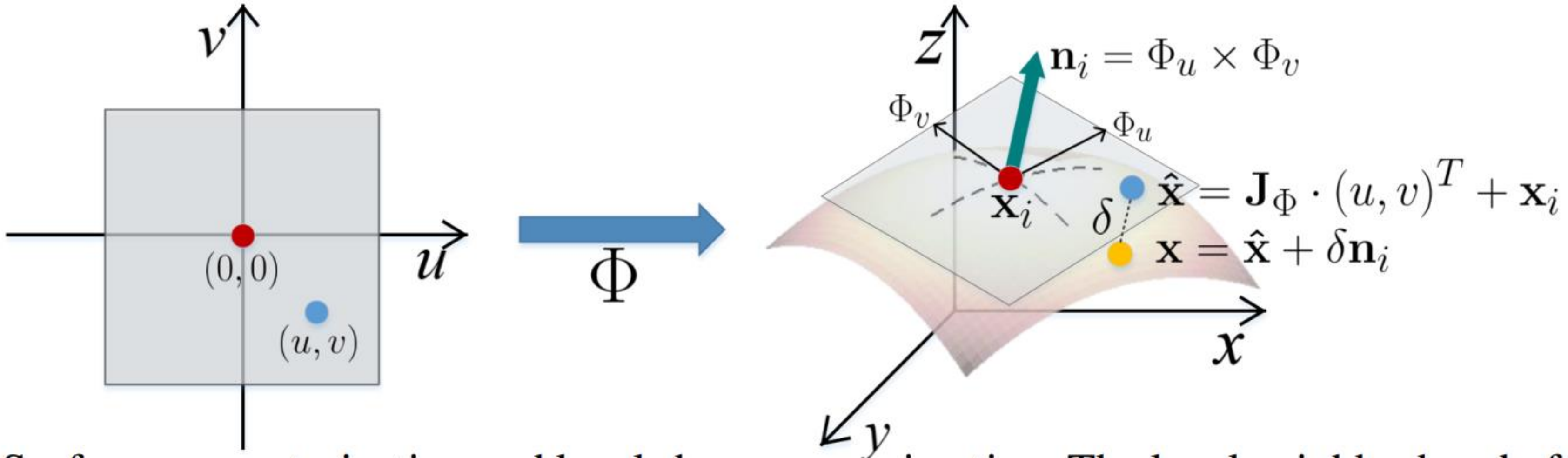
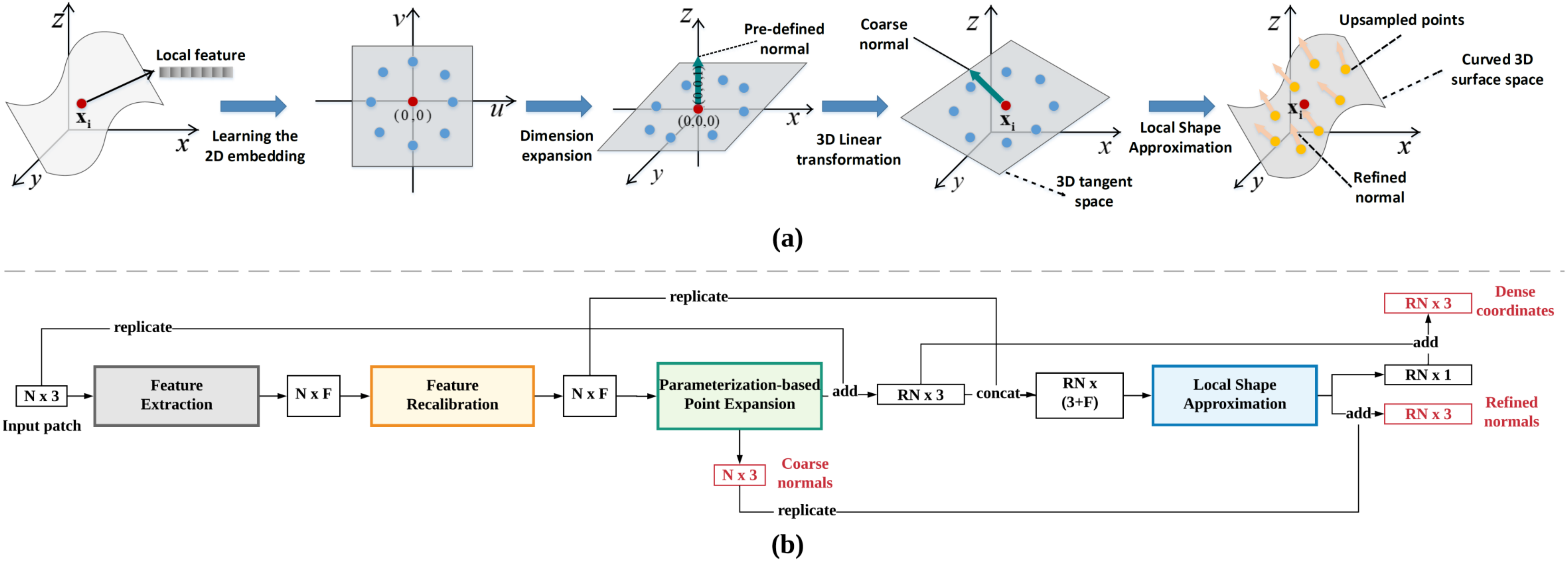


Fig. 2: Surface parameterization and local shape approximation. The local neighborhood of $\mathbf{x}_i$ is parameterized to a 2D rectangular domain via a differentiable map $\boldsymbol{\Phi} : \mathbb{R}^2 \rightarrow \mathbb{R}^3$. The Jacobian matrix $\mathbf{J}_{\Phi}(0,0)$ provides the best linear approximation of $\boldsymbol{\Phi}$ at $\mathbf{x}_i$, which maps (u,v) to a point $\hat{\mathbf{x}}$ on the tangent plane of $\mathbf{x}_i$. Furthermore, using the principal curvatures of $\mathbf{x}_i$, we can reconstruct the local geometry of $\mathbf{x}_i$ in the second-order accuracy.

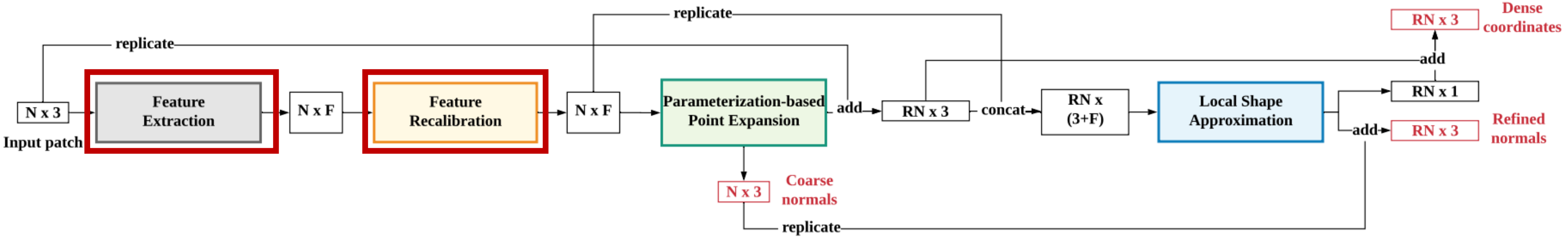
Proposed Up-sampling Method: PUGeo-Net

- **Flowchart of PUGeo-Net**

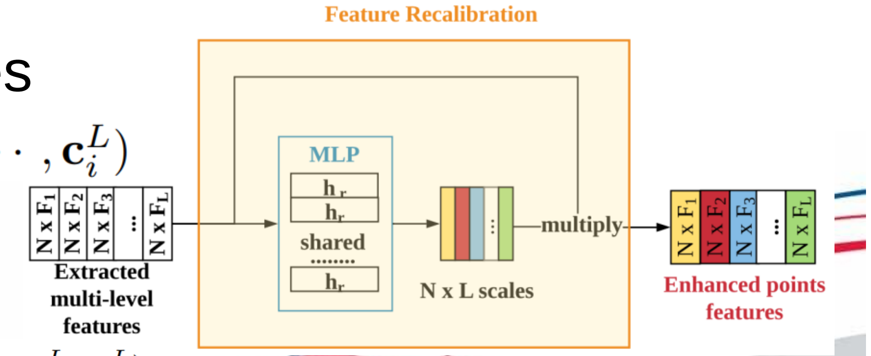


(a) Visual illustration of our method. (b) The neural network architecture.

Proposed Up-sampling Method: PUGeo-Net

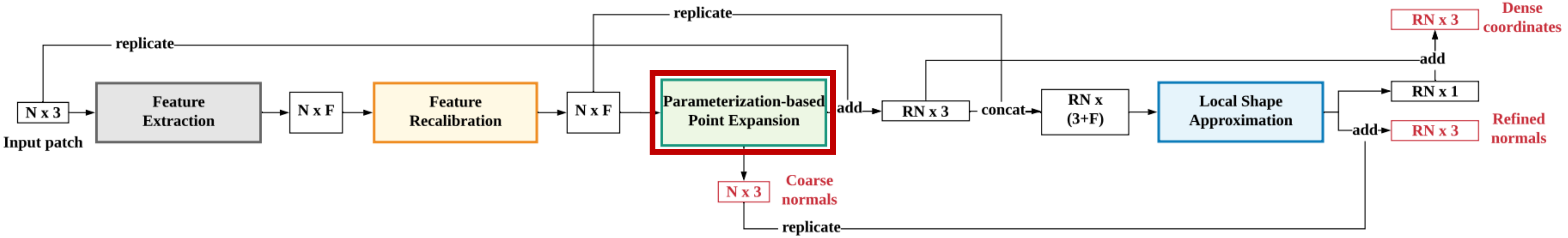


- **Hierarchical feature embedding module**
  - ➢Extract features from low- to high-levels. We adopt the standard DGCNN to realize this module.
- **Feature recalibration**
  - ➢Self-gating attention to enhance multi-scale features
    - ✓concatenate features of all L layers: $\hat{\mathbf{c}}_i = \text{Concat}(\mathbf{c}_i^1, \cdots, \mathbf{c}_i^L)$
    - ✓utilize an MLP to obtain logits: $\mathbf{a}_i = h_r(\hat{\mathbf{c}}_i)$
    - ✓obtain recalibration weights: $w_i^l = e^{a_i^l} / \sum_{k=1}^{L} e^{a_i^k}$
    - ✓recalibrate multi-scale features: $\mathbf{c}_i = \text{Concat}(w_i^1 \cdot \mathbf{c}_i^1, w_i^2 \cdot \mathbf{c}_i^2, \cdots, \hat{a}_i^L \cdot \mathbf{c}_i^L)$



Y. Wang, *et al*. "Dynamic graph cnn for learning on point clouds." *ACM TOG*, 38.5 (2019): 1-12.

Proposed Up-sampling Method: PUGeo-Net



- **Parameterization-based point expansion**
  - The input points are expended R times, leading to a coarse dense point cloud as well as coarse normal.
    - ✓adaptive sampling in the 2D parametric domain

$$\{(u_i^r, v_i^r)|r=1,2,\cdots,R\} = f_1(\mathbf{c}_i)$$

    - ✓ prediction of linear transformation

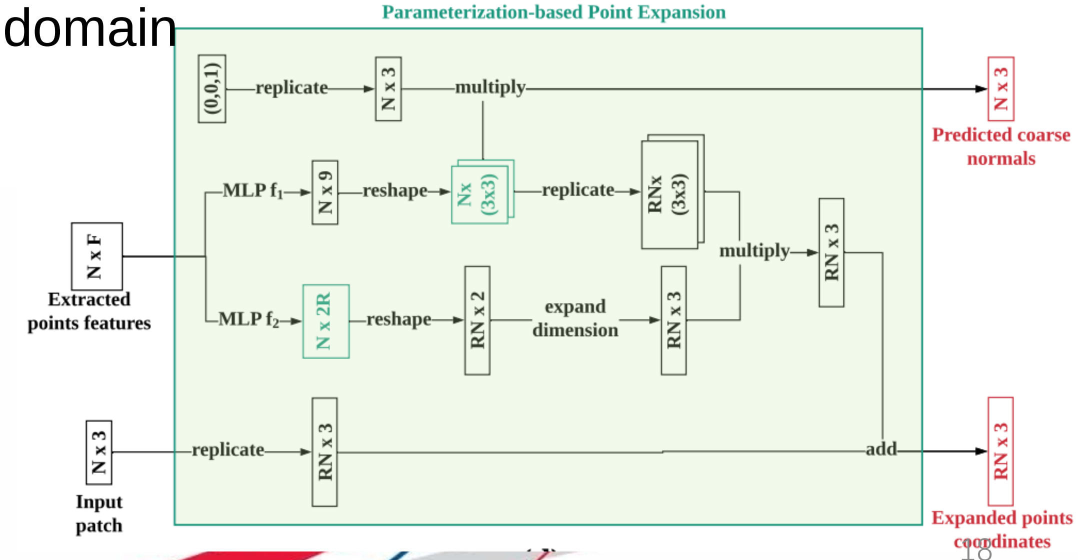
$$\mathbf{T}_i = f_2(\mathbf{c}_i)$$

    - ✓lift the points to the tangent plane of $\mathbf{x}_i$

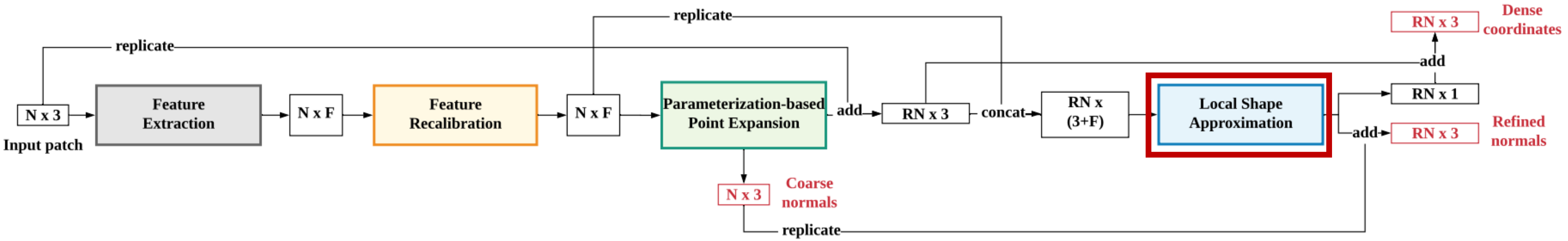
$$\widehat{\mathbf{x}}_i^r = (\widehat{x}_i^r, \widehat{y}_i^r, \widehat{z}_i^r)^\mathsf{T} = \mathbf{T}_i \cdot (u_i^r, v_i^r, 0)^\mathsf{T} + \mathbf{x}_i$$

    - ✓prediction of the coarse normal

$$\mathbf{n}_i = \mathbf{T}_i \cdot (0,0,1)^\mathsf{T}$$



Proposed Up-sampling Method: PUGeo-Net



- **Local shape approximation**
  - ➢The points located on the tangent plane are wrapped to the curved space. Based on the 2nd order approximation, the warping should be along the normal direction with a displacement.
    - ✓predict the displacement $\delta_i^r = f_3(\text{Concat}(\widehat{\mathbf{x}}_i^r, \mathbf{c}_i))$
    - ✓update dense points $\mathbf{x}_i^r = (x_i^r, y_i^r, z_i^r)^\mathsf{T} = \widehat{\mathbf{x}}_i^r + \mathbf{T}_i \cdot (0, 0, \delta_i^r)^\mathsf{T}$
    - ✓predict normal offset $\Delta\mathbf{n}_i^r = f_4\left(\text{Concat}(\widehat{\mathbf{x}}_i^r, \mathbf{c}_i)\right)$
    - ✓update dense normal $\mathbf{n}_i^r = \Delta\mathbf{n}_i^r + \mathbf{n}_i$

Proposed Up-sampling Method: PUGeo-Net

- **Joint training loss for PUGeo-Net**

$$L_{total} = \alpha L_{CD} + \beta L_{coarse} + \gamma L_{refined}$$

  - L_{CD} measures the distance between the up-sampled point cloud $\mathcal{X}_R$ and the corresponding ground-truth one $\mathcal{Y}_R = \{\mathbf{y}_k\}_{k=1}^{RM}$ via Chamfer Distance (CD):

$$L_{CD} = \frac{1}{RM}\left(\sum_{\mathbf{x}_i^r \in \mathcal{X}_R} ||\mathbf{x}_i^r - \phi(\mathbf{x}_i^r)||_2 + \sum_{\mathbf{y}_k \in \mathcal{Y}_R} ||\mathbf{y}_k - \psi(\mathbf{y}_k)||_2\right),$$

  - L_{coarse} measures the error between the predicted coarse normal $\widetilde{\mathcal{N}} = \{\tilde{\mathbf{n}}_i\}_{i=1}^{M}$ and the ground-truth one $\mathcal{N}$:

$$L_{coarse}(\mathcal{N}, \widetilde{\mathcal{N}}) = \sum_{i=1}^{M} L(\mathbf{n}_i, \tilde{\mathbf{n}}_i) \qquad L(\mathbf{n}_i, \tilde{\mathbf{n}}_i) = \max\left\{||\mathbf{n}_i - \tilde{\mathbf{n}}_i||_2^2, ||\mathbf{n}_i + \tilde{\mathbf{n}}_i||_2^2\right\}$$

  - $L_{refined}$ measures the error between the predicted dense normal $\overline{\mathcal{N}}_R = \{\overline{\mathbf{n}}_k\}_{k=1}^{RM}$ and the ground-truth one $\mathcal{N}_R$:

$$L_{refined}(\mathcal{N}_R, \overline{\mathcal{N}}_R) = \sum_{i=1}^{M}\sum_{r=1}^{R} L(\mathbf{n}_i^r, \overline{\mathbf{n}}_{\phi(\mathbf{x}_i^r)})$$

Proposed Up-sampling Method: PUGeo-Net

- **Experiments**
  - Quantitative comparisons with SOTA methods

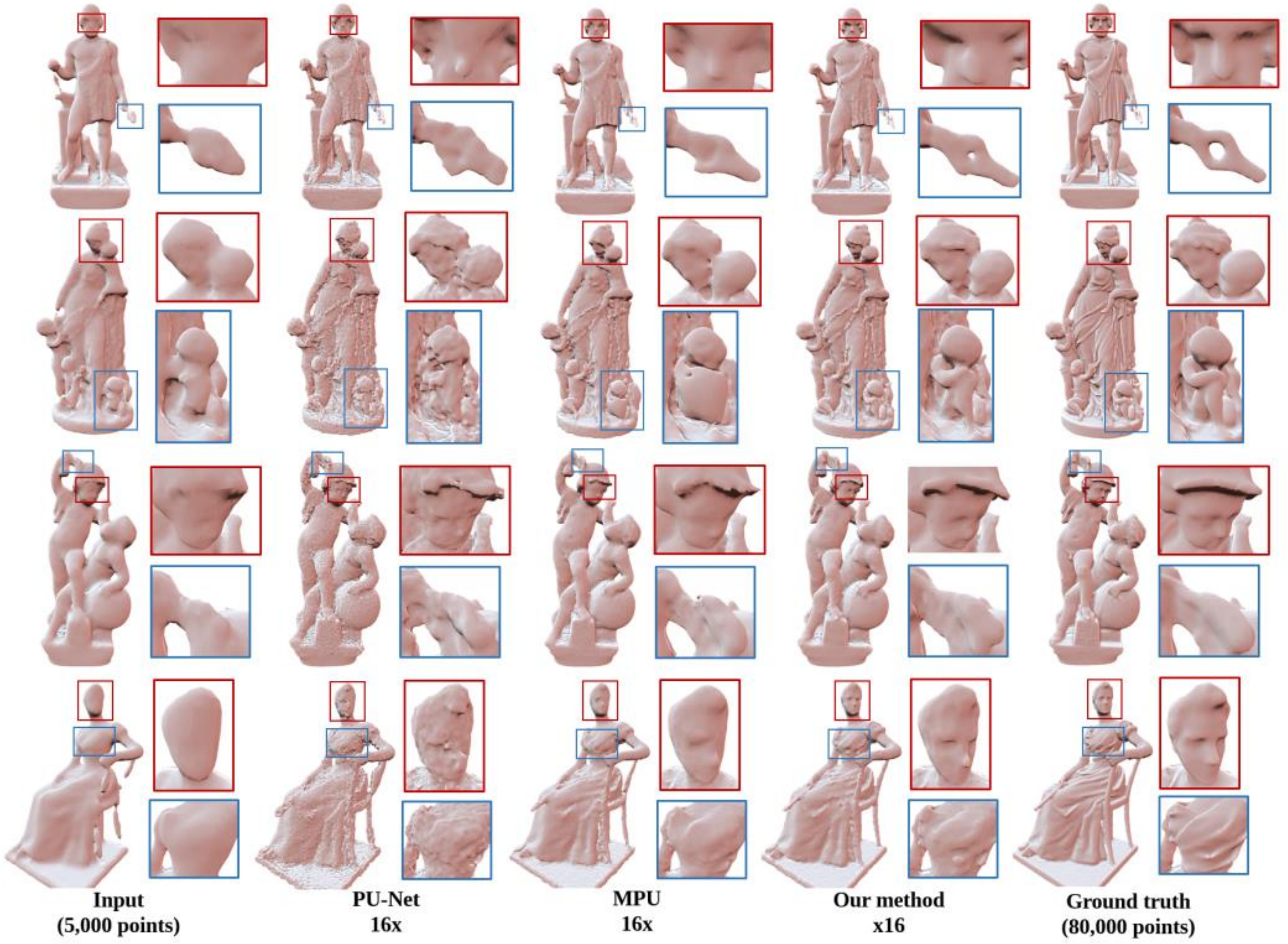
CD: Chamfer distance. **HD**: Hausdorff distance. **P2F:** Point-to-surface distance. **JSD**: Jensen-Shannon divergence

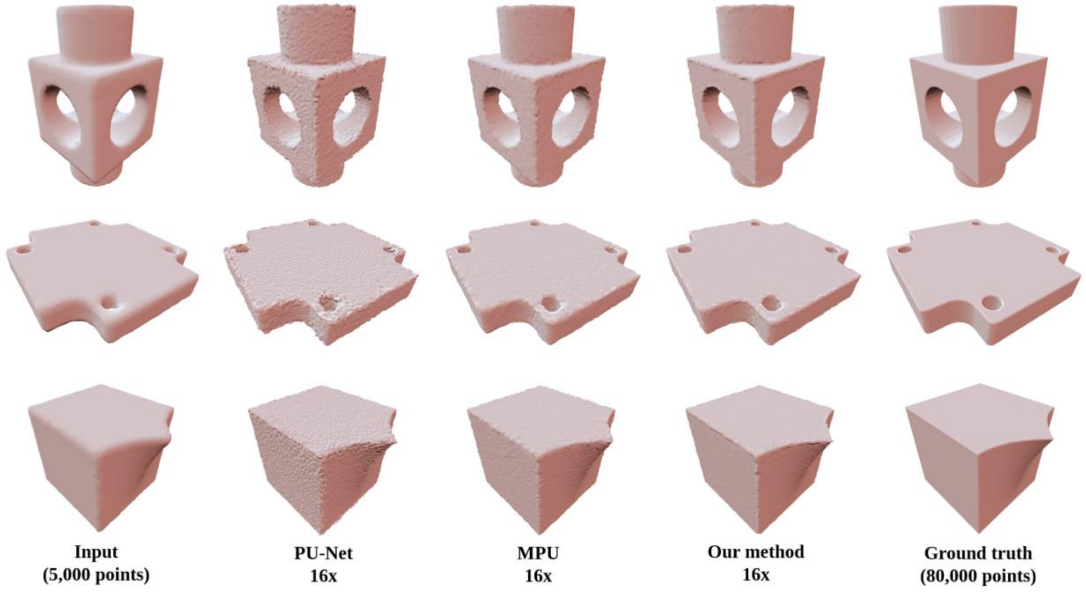
R	Method	Network size	CD (10^{-2})	HD (10^{-2})	JSD (10^{-2})	P2F mean (10^{-3})	P2F std (10^{-3})	CD# (10^{-2})	HD# (10^{-2})	JSD# (10^{-2})
4×	EAR [23]	-	0.919	5.414	4.047	3.672	5.592	1.022	6.753	7.445
	PU-Net [24]	10.1 MB	0.658	1.003	0.950	1.532	1.215	0.648	5.850	4.264
	MPU [26]	92.5 MB	0.573	1.073	0.614	0.808	0.809	0.647	5.493	4.259
	PUGeo-Net	26.6 MB	**0.558**	**0.934**	**0.444**	**0.617**	**0.714**	**0.639**	**5.471**	**3.928**
8×	EAR [23]	-	-	-	-	-	-	-	-	-
	PU-Net [24]	14.9 MB	0.549	1.314	1.087	1.822	1.427	0.594	5.770	3.847
	MPU [26]	92.5 MB	0.447	1.222	0.511	0.956	0.972	0.593	5.723	3.754
	PUGeo-Net	26.6 MB	**0.419**	**0.998**	**0.354**	**0.647**	**0.752**	**0.549**	**5.232**	**3.465**
12×	EAR [23]	-	-	-	-	-	-	-	-	-
	PU-Net [24]	19.7 MB	0.434	0.960	0.663	1.298	1.139	0.573	6.056	3.811
	MPU [26]	-	-	-	-	-	-	-	-	-
	PUGeo-Net	26.7 MB	**0.362**	**0.978**	**0.325**	**0.663**	**0.744**	**0.533**	**5.255**	**3.322**
16×	EAR [23]	-	-	-	-	-	-	-	-	-
	PU-Net [24]	24.5 MB	0.482	1.457	1.165	2.092	1.659	0.588	6.330	3.744
	MPU [26]	92.5 MB	0.344	1.355	0.478	0.926	1.029	0.573	5.923	3.630
	PUGeo-Net	26.7 MB	**0.323**	**1.011**	**0.357**	**0.694**	**0.808**	**0.524**	**5.267**	**3.279**

CD#, HD#, JSD#: these 3 metrics are used to measure the distance between dense point clouds sampled from reconstructed surfaces and ground truth meshes.

Proposed Up-sampling Method: PUGeo-Net

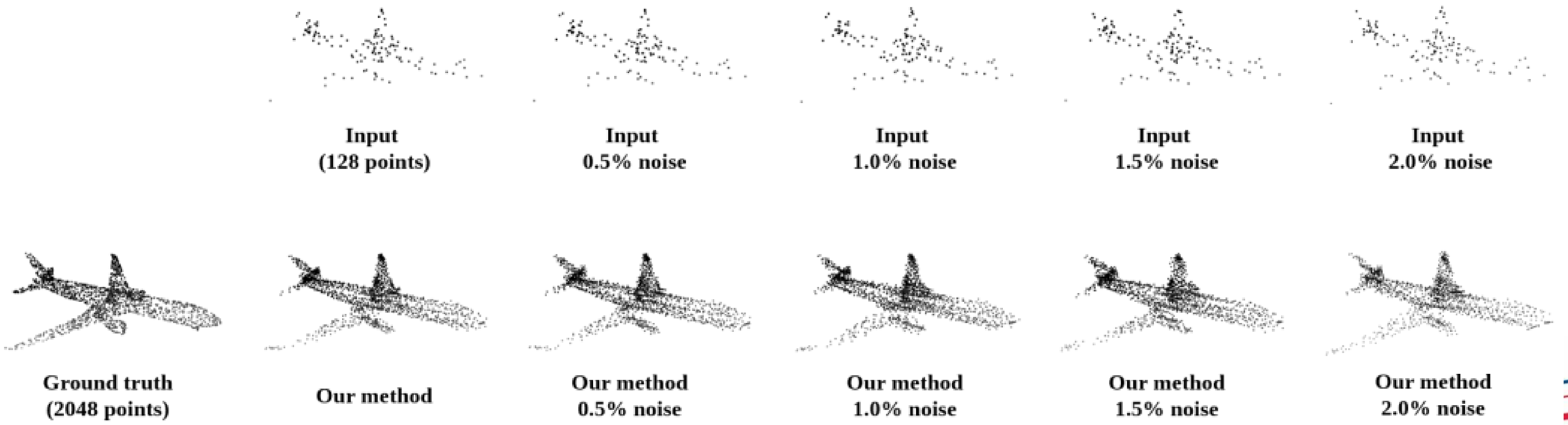
- **Experiments**
  - ➢Visual comparisons with SOTA methods





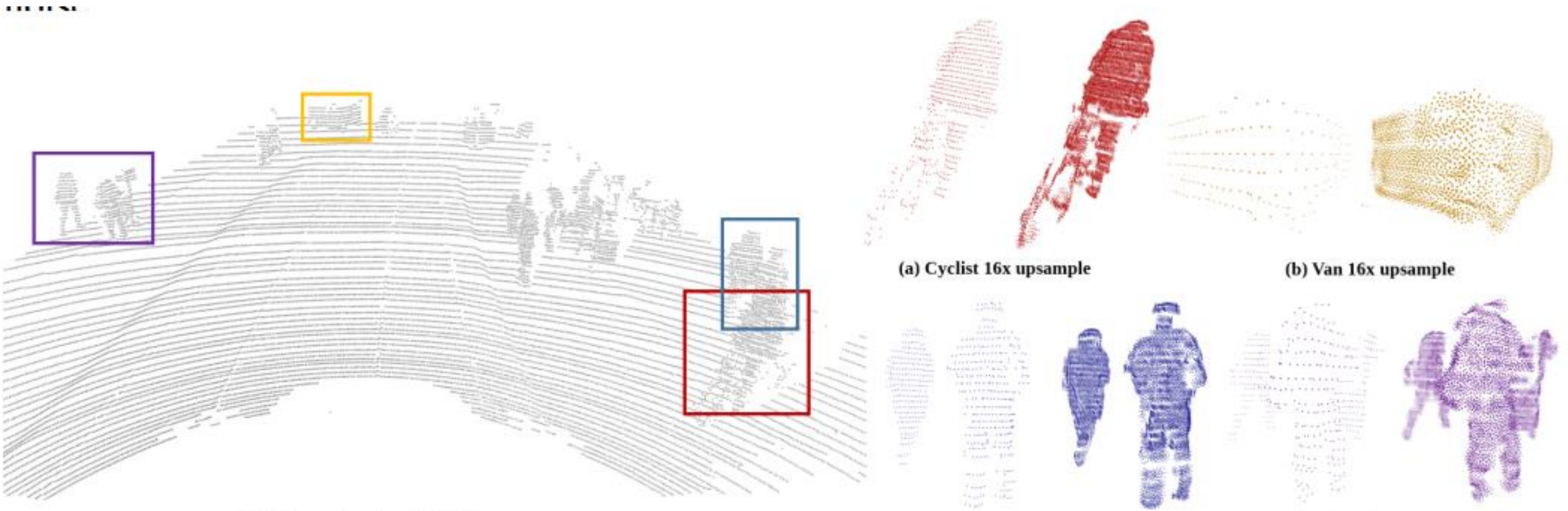
Proposed Up-sampling Method: PUGeo-Net

- **Experiments: Robustness validation**
  - ➢Noisy and non-uniform data



Proposed Up-sampling Method: PUGeo-Net

- **Experiments: Robustness validation**
  - ➢Scanned data by LiDAR



LiDAR scanning data (KITTI)

(a) Cyclist 16x upsample

(b) Van 16x upsample

(c) Pedestrians 16x upsample

(d) Pedestrians 16x upsample

Proposed Up-sampling Method: PUGeo-Net

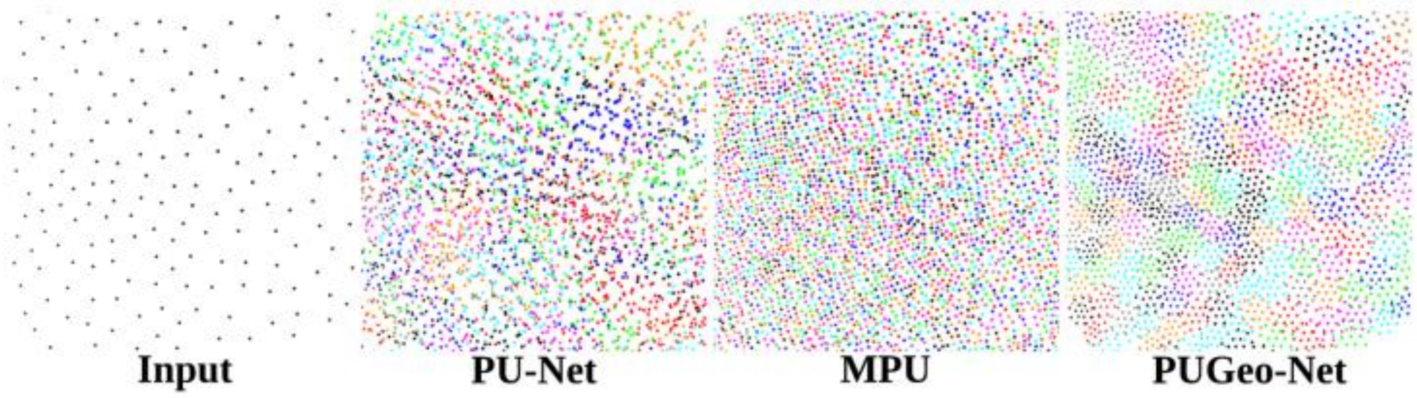
- **Experiments**
  - ➢Ablation studies

Table 3: Ablation study. **Feature recalibration**: concatenate multiscale feature directly without the recalibration module. **Normal prediction**: only regress coordinates of points without normal prediction and supervision. **Learned adaptive 2D sampling**: use a predefined 2D regular grid as the parametric domain instead of the learned adaptive 2D smapling. **Linear transformation**: regress coordinates and normals by non-linear MLPs directly without prediction of the linear transformation. **Coarse to fine**: directly regress coordinates and normals without the intermediate coarse prediction.

Networks	CD	HD	JSD	P2F mean	P2F std	CD#	HD#	JSD#
Feature recalibration	0.325	1.016	0.371	0.725	0.802	0.542	5.654	3.425
Normal prediction	0.331	2.232	0.427	0.785	0.973	0.563	5.884	3.565
Learned adaptive 2D sampling	0.326	1.374	0.407	0.701	0.811	0.552	5.758	3.456
Linear transformation	0.394	**1.005**	1.627	0.719	**0.720**	1.855	11.479	9.841
Coarse to fine	0.330	1.087	0.431	0.746	0.748	0.534	**5.241**	3.348
Full model	**0.323**	1.011	**0.357**	**0.694**	0.808	**0.524**	5.267	**3.279**

Proposed Up-sampling Method: PUGeo-Net

- **Experiments: validation of our method's properties**
  - ➢Comparison of the distribution of generated points by different methods



  - ➢Geometry-centric nature

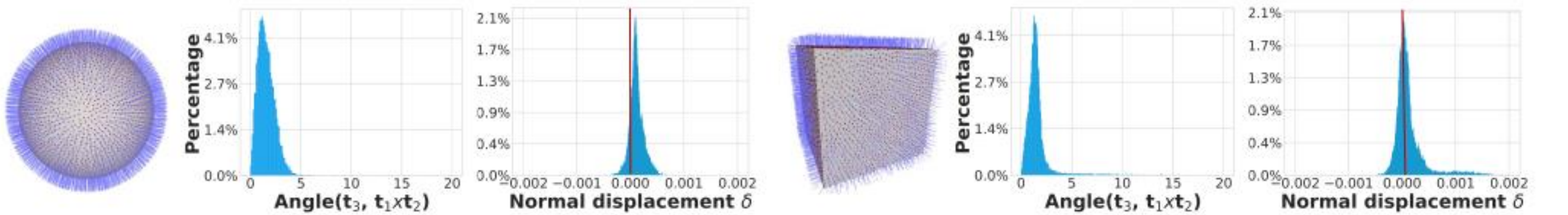


Fig. 10: Statistical analysis of the predicted transformation matrix $\mathbf{T} = [\mathbf{t}_1; \mathbf{t}_2; \mathbf{t}_3] \in \mathbb{R}^{3\times 3}$ and normal displacement δ, which can be used to fully reconstruct the local geometry.

Proposed Down-sampling Method

- **Problem formulation from the perspective of matrix optimization**
  - ➢Input point cloud $\mathcal{P} = \{\mathbf{p}_i \in \mathbb{R}^3\}_{i=1}^{n}$, down-sampled one $\mathcal{Q} = \{\mathbf{q}_i \in \mathbb{R}^3\}_{i=1}^{m}$

$$\min_{\mathcal{Q}} L_{task}(\mathcal{Q}) \quad s.t. \quad \mathcal{Q} \subset \mathcal{P}$$

$$\min_{\mathbf{S}} L_{task}(\mathbf{Q})$$
$$s.t.\ \mathbf{Q} = \mathbf{PS},\ \mathbf{1}_n^\top \mathbf{S} = \mathbf{1}_m^\top,\ \mathbf{S}^\top \mathbf{S} = \mathbf{I}_m,\ s_{i,j} \in \{0,1\}$$

$$\min_{\mathbf{S}} L_{task}(\mathbf{Q}) + \alpha L_{dist}(\mathbf{P},\mathbf{Q})$$
$$s.t.\ \mathbf{Q} = \mathbf{PS},\ \mathbf{S} \geq 0,\ \mathbf{1}_n^\top \mathbf{S} = \mathbf{1}_m^\top,\ \|\mathbf{S}^\top \mathbf{S} - \mathbf{I}_m\|_F < \epsilon,$$

$$\min_{\mathbf{S}} L_{task}(\mathbf{\Phi}^{-1}(\mathbf{\Phi}(\mathbf{Q}))) + \alpha L_{dist}(\mathbf{P}, \mathbf{\Phi}^{-1}(\mathbf{\Phi}(\mathbf{Q}))),$$
$$s.t.\ \mathbf{\Phi}(\mathbf{Q}) = \mathbf{\Phi}(\mathbf{P})\mathbf{S},\ \mathbf{S} \geq 0,\ \mathbf{1}_n^\top \mathbf{S} = \mathbf{1}_m^\top,\ \|\mathbf{S}^\top \mathbf{S} - \mathbf{I}_m\|_F < \epsilon,$$

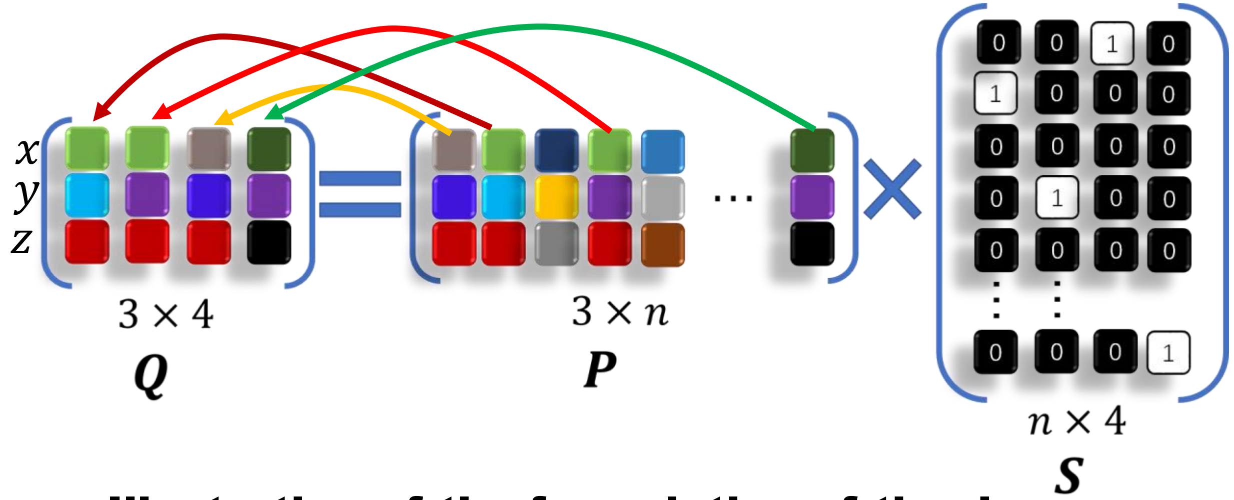
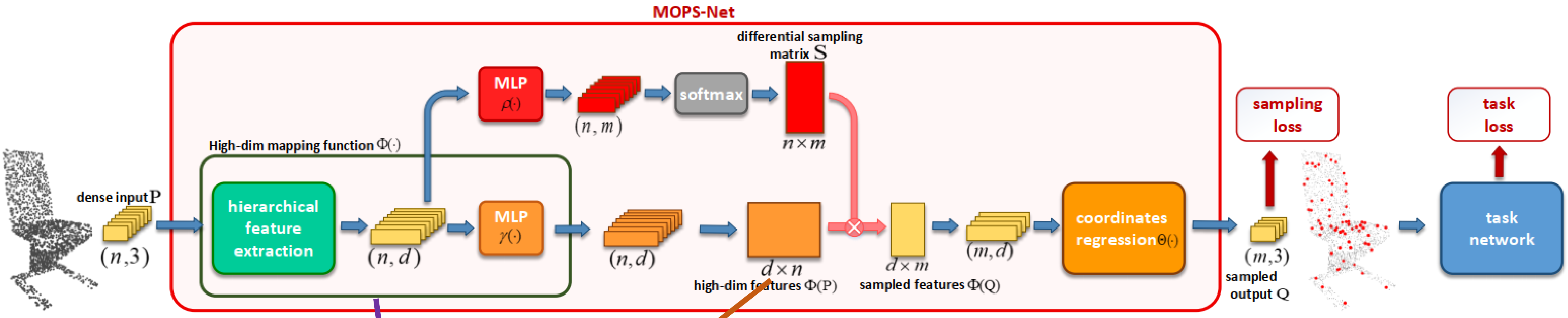


Illustration of the formulation of the down-sampling problem with matrix multiplication

$\mathbf{\Phi}(\cdot)$ is feature mapping function and $\mathbf{\Phi}^{-1}(\cdot)$ is its inverse.

Proposed Down-sampling Method: MOPS-Net

- **MOPS-Net: a matrix optimization-driven network**
  - ➢Flowchart

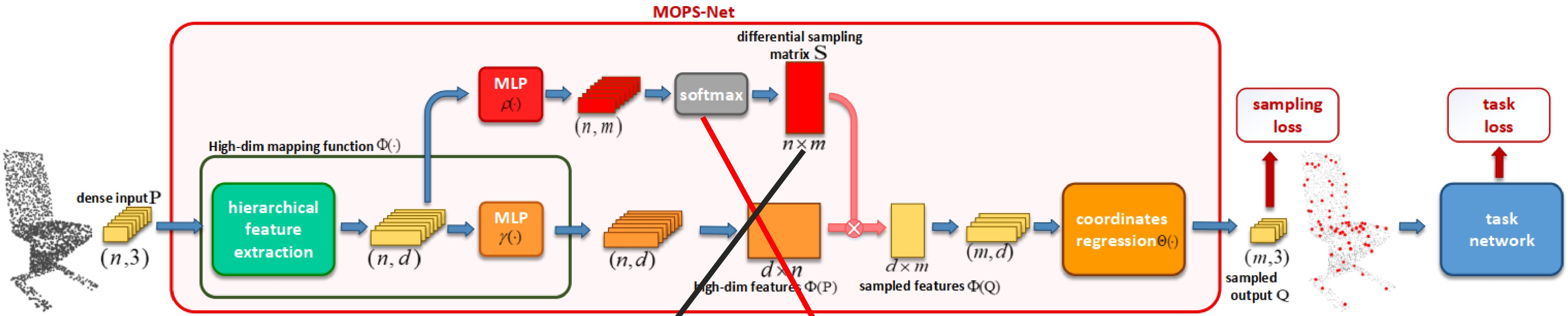


$$\min_{\mathbf{S}} L_{task}(\mathbf{\Phi}^{-1}(\mathbf{\Phi}(\mathbf{Q}))) + \alpha L_{dist}(\mathbf{P}, \mathbf{\Phi}^{-1}(\mathbf{\Phi}(\mathbf{Q}))),$$

$$s.t.\ \mathbf{\Phi}(\mathbf{Q}) = \boxed{\boxed{\mathbf{\Phi}}(\mathbf{P})}\mathbf{S},\ \mathbf{S} \geq 0,\ \mathbf{1}_n^{\mathsf{T}}\mathbf{S} = \mathbf{1}_m^{\mathsf{T}},\ \|\mathbf{S}^{\mathsf{T}}\mathbf{S} - \mathbf{I}_m\|_F < \epsilon,$$

Proposed Down-sampling Method: MOPS-Net

- **MOPS-Net: a matrix optimization-driven network**
  - ➢Flowchart

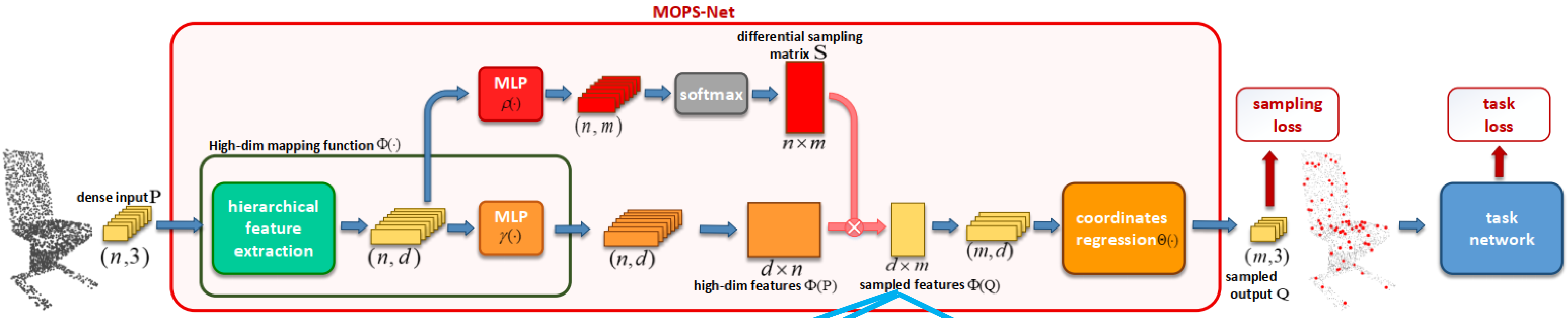


$$\min_{\mathbf{S}} L_{task}(\mathbf{\Phi}^{-1}(\mathbf{\Phi}(\mathbf{Q}))) + \alpha L_{dist}(\mathbf{P}, \mathbf{\Phi}^{-1}(\mathbf{\Phi}(\mathbf{Q}))),$$

$$s.t.\ \mathbf{\Phi}(\mathbf{Q}) = \mathbf{\Phi}(\mathbf{P})\boxed{\mathbf{S}}, \boxed{\mathbf{S} \geq 0,\ \mathbf{1}_n^{\mathsf{T}}\mathbf{S} = \mathbf{1}_m^{\mathsf{T}},\ \|\mathbf{S}^{\mathsf{T}}\mathbf{S} - \mathbf{I}_m\|_F < \epsilon,}$$

Proposed Down-sampling Method: MOPS-Net

- **MOPS-Net: a matrix optimization-driven network**
  - ➢Flowchart

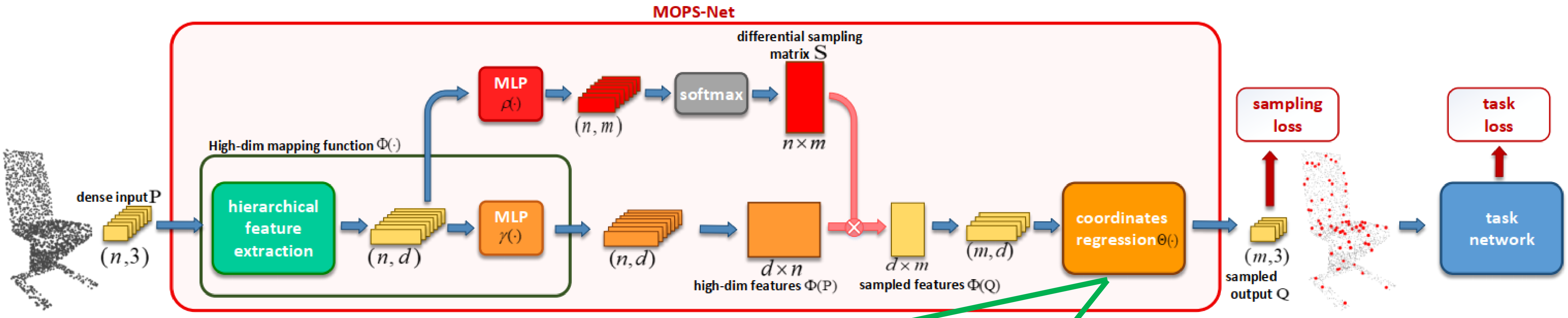


$$\min_{\mathbf{S}} L_{task}(\mathbf{\Phi}^{-1}(\boxed{\mathbf{\Phi}(\mathbf{Q})})) + \alpha L_{dist}(\mathbf{P}, \mathbf{\Phi}^{-1}(\boxed{\mathbf{\Phi}(\mathbf{Q})})),$$

$$s.t.\ \boxed{\mathbf{\Phi}(\mathbf{Q})} = \mathbf{\Phi}(\mathbf{P})\mathbf{S},\ \mathbf{S} \geq 0,\ \mathbf{1}_n^{\mathsf{T}}\mathbf{S} = \mathbf{1}_m^{\mathsf{T}},\ \|\mathbf{S}^{\mathsf{T}}\mathbf{S} - \mathbf{I}_m\|_F < \epsilon,$$

Proposed Down-sampling Method: MOPS-Net

- **MOPS-Net: a matrix optimization-driven network**
  - ➢Flowchart

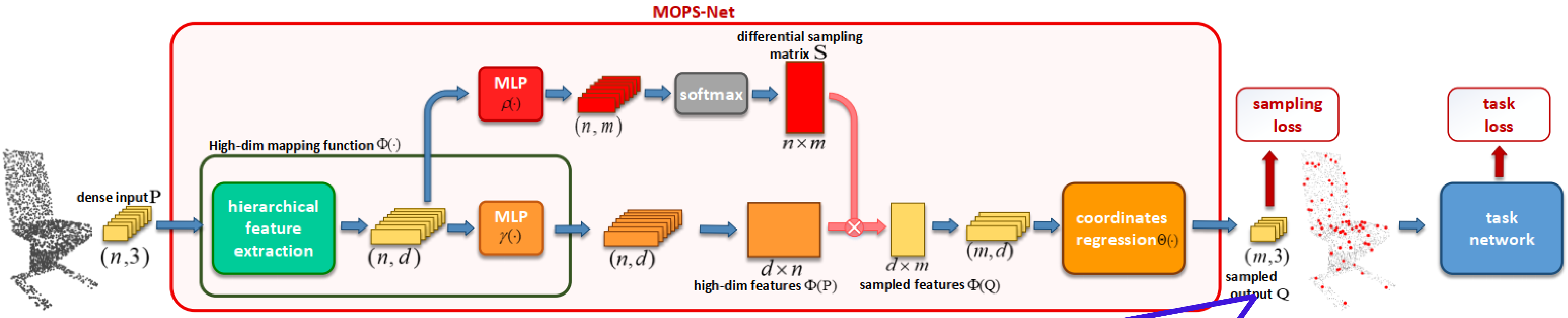


$$\min_{\mathbf{S}} L_{task}(\boxed{\mathbf{\Phi}^{-1}}(\mathbf{\Phi}(\mathbf{Q}))) + \alpha L_{dist}(\mathbf{P}, \boxed{\mathbf{\Phi}^{-1}}(\mathbf{\Phi}(\mathbf{Q}))),$$

$$s.t.\ \mathbf{\Phi}(\mathbf{Q}) = \mathbf{\Phi}(\mathbf{P})\mathbf{S},\ \mathbf{S} \geq 0,\ \mathbf{1}_n^{\mathsf{T}}\mathbf{S} = \mathbf{1}_m^{\mathsf{T}},\ \|\mathbf{S}^{\mathsf{T}}\mathbf{S} - \mathbf{I}_m\|_F < \epsilon,$$

Proposed Down-sampling Method: MOPS-Net

- **MOPS-Net: a matrix optimization-driven network**
  - ➢Flowchart

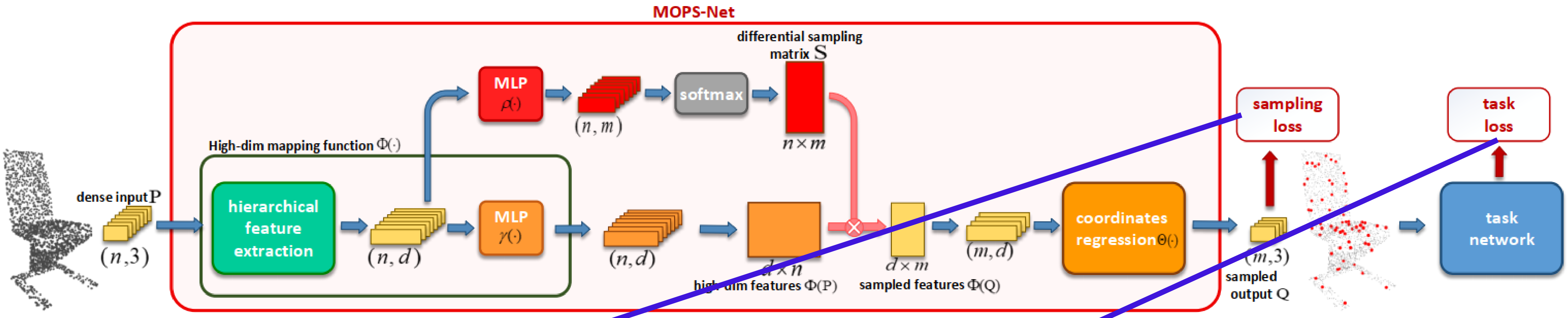


$$\min_{\mathbf{S}} L_{task}(\mathbf{\Phi}^{-1}(\mathbf{\Phi}(\boxed{\mathbf{Q}}))) + \alpha L_{dist}(\mathbf{P}, \mathbf{\Phi}^{-1}(\mathbf{\Phi}(\boxed{\mathbf{Q}}))),$$

$$s.t.\ \mathbf{\Phi}(\mathbf{Q}) = \mathbf{\Phi}(\mathbf{P})\mathbf{S},\ \mathbf{S} \geq 0,\ \mathbf{1}_n^{\mathsf{T}}\mathbf{S} = \mathbf{1}_m^{\mathsf{T}},\ \|\mathbf{S}^{\mathsf{T}}\mathbf{S} - \mathbf{I}_m\|_F < \epsilon,$$

Proposed Down-sampling Method: MOPS-Net

- **MOPS-Net: a matrix optimization-driven network**
  - ➢Flowchart



$$\min_{\mathbf{S}} \boxed{L_{task}}(\mathbf{\Phi}^{-1}(\mathbf{\Phi}(\mathbf{Q}))) + \alpha \boxed{L_{dist}}(\mathbf{P}, \mathbf{\Phi}^{-1}(\mathbf{\Phi}(\mathbf{Q}))),$$

$$s.t.\ \mathbf{\Phi}(\mathbf{Q}) = \mathbf{\Phi}(\mathbf{P})\mathbf{S},\ \mathbf{S} \geq 0,\ \mathbf{1}_n^{\mathsf{T}}\mathbf{S} = \mathbf{1}_m^{\mathsf{T}},\ \|\mathbf{S}^{\mathsf{T}}\mathbf{S} - \mathbf{I}_m\|_F < \epsilon,$$

Proposed Down-sampling Method: MOPS-Net

- **Joint training loss for MOPS-Net**

$$L_{total}(\mathcal{P}, \mathcal{Q}) = L_{task}(\mathcal{Q}) + \alpha L_{dist}(\mathcal{P}, \mathcal{Q})$$

- ➢ L_{task} measures the subsequent task error for down-sampled point clouds $\mathcal{Q}$.
- ➢ L_{dist} regularizes the network to learn down-sampled point clouds that are close to the input.

$$L_{dist}(\mathcal{P}, \mathcal{Q}) = L_{subset}(\mathcal{P}, \mathcal{Q}) + \beta L_{coverage}(\mathcal{P}, \mathcal{Q}),$$

✓ L_{subset} constrains $\mathcal{Q}$ close to subset of input.

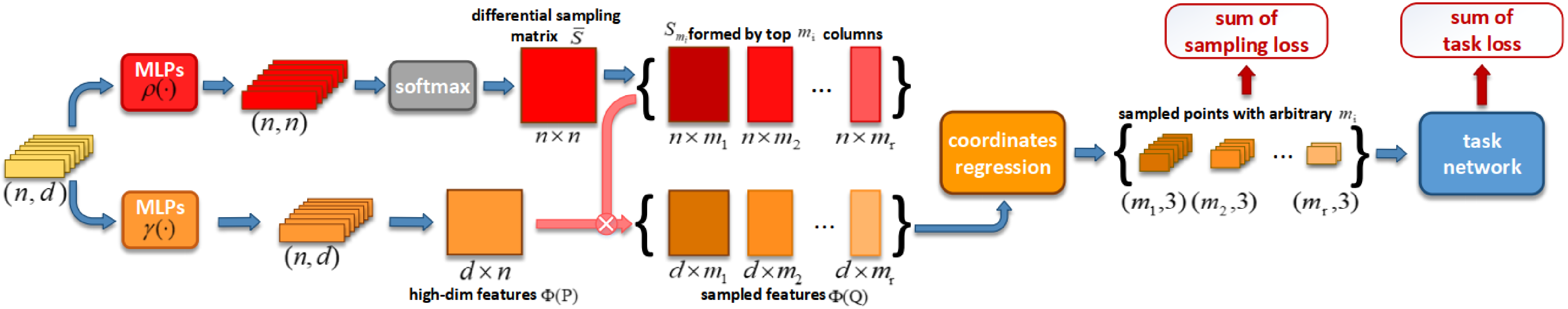
$$L_{subset}(\mathcal{P}, \mathcal{Q}) = \frac{\tau}{M} \sum_{i=1,\ldots,m} \min_{\mathbf{p}\in\mathcal{P}} ||\mathbf{q}_i - \mathbf{p}||_2^2 + \max_{i=1,\ldots,m} \min_{\mathbf{p}\in\mathcal{P}} ||\mathbf{q}_i - \mathbf{p}||_2^2$$

✓ $L_{coverage}$ encourages $\mathcal{Q}$ preserve the overall shapes of input.

$$L_{coverage}(\mathcal{P}, \mathcal{Q}) = \frac{1}{n} \sum_{i=1,\ldots,n} \min_{\mathbf{q}\in\mathcal{Q}} ||\mathbf{p}_i - \mathbf{q}||_2^2$$

Proposed Down-sampling Method: MOPS-Net

- **Extension: Flexible MOPS-Net for arbitrary ratios**



- A single network to down-sample with arbitrary sampling ratios after only one-time training.
- Instead of learning a rectangular sampling matrix, we learn a square matrix $\bar{\mathbf{S}} \in \mathbb{R}^{n \times n}$, the left-most columns are selected to form sampling matrix $\mathbf{S} \in \mathbb{R}^{n \times m}$ to produce m points.
- Trained by multi-level loss function

$$L_{sum}(\mathcal{P}, \{\mathcal{Q}_{m_i}\}) = \sum_{i=1,\ldots,r} \lambda_i L_{total}(\mathcal{P}, \mathcal{Q}_{m_i}; \mathbf{S}_{m_i})$$

Proposed Down-sampling Method: MOPS-Net

- **Experiments**
  - Classification-driven down-sampling

TABLE 1: Comparisons of the classification accuracy by different downsampling methods. The larger, the better. Note the classification accuracy of original point clouds with 1024 points each is 0.892 when using the same classification network.

m	RS	FPS	S-Net	S-Net-M	MOPS-Net	MOPS-Net-M	FMOPS-Net	FMOPS-Net-M
512	0.878	0.879	0.857	0.882	0.883	0.883	0.876	0.883
256	0.797	0.846	0.860	0.847	0.874	0.867	0.872	0.862
128	0.607	0.760	0.852	0.793	0.872	0.850	0.859	0.833
64	0.348	0.564	0.855	0.708	0.871	0.810	0.844	0.761
32	0.154	0.289	0.854	0.610	0.861	0.776	0.741	0.524
16	0.061	0.149	0.794	0.349	0.847	0.512	0.576	0.236

TABLE 2: Comparisons of classification accuracy of different downsampling methods when the classification network was trained with downsampled point clouds by each method.

m	RS	FPS	S-Net-M	MOPS-Net-M
512	0.869	0.886	0.885	0.886
256	0.873	0.883	0.881	0.883
128	0.863	0.875	0.874	0.885
64	0.827	0.863	0.865	0.879
32	0.764	0.841	0.864	0.879
16	0.668	0.812	0.844	0.873

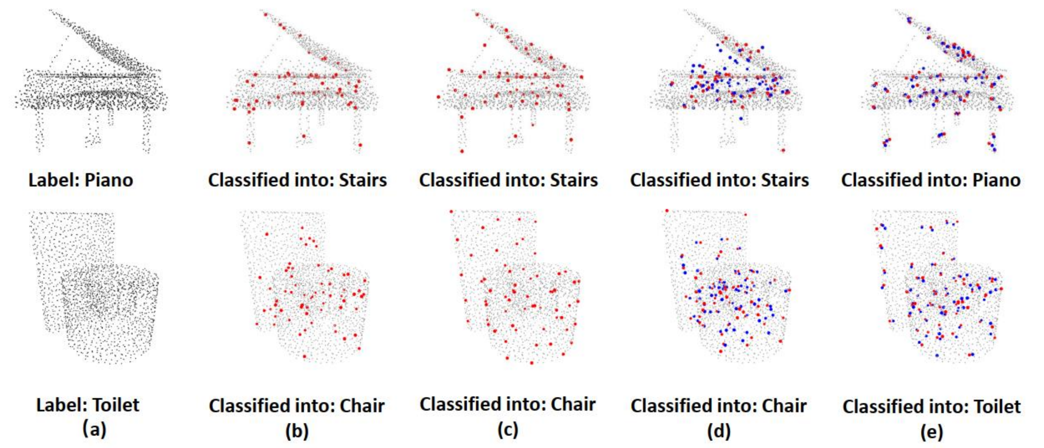


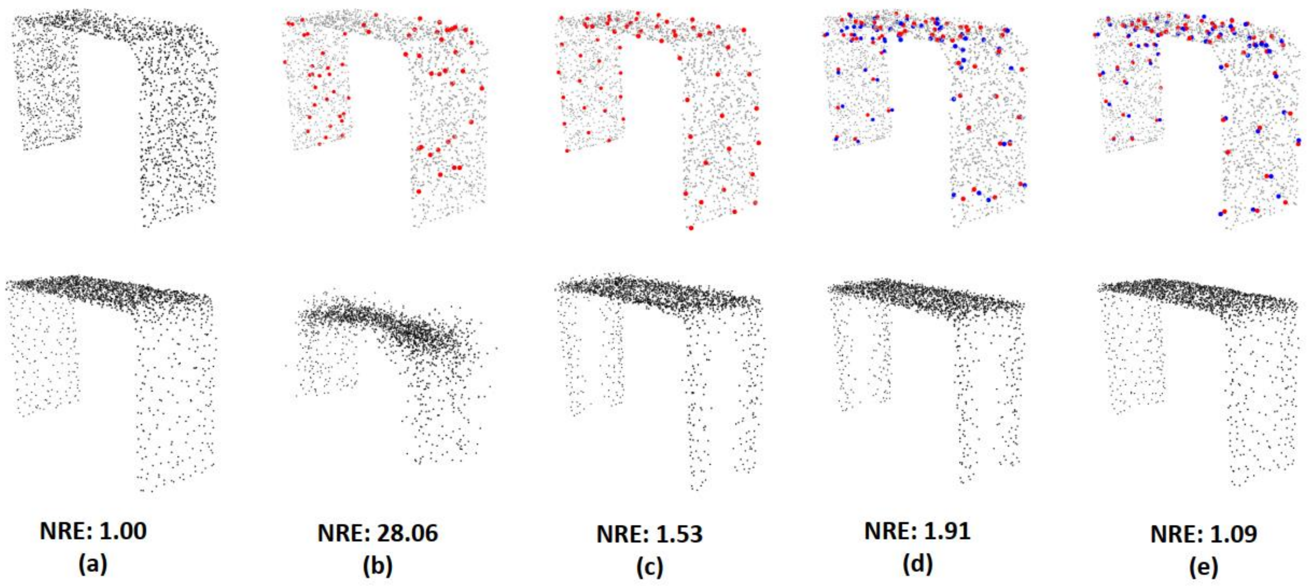
Fig. 4: Visual comparisons of sampled point clouds by different downsampling methods with $m = 64$ as well as the classification results. (a) The original dense point clouds (1024 points); (b) Random sampling results; (c) FPS results; (d) Results of S-Net (blue) and S-Net-M (red); and (e) Results of MOPS-Net (blue) and MOPS-Net-M (red). The classification results of (d) and (e) are those of S-Net-M and MOPS-Net-M.

Proposed Down-sampling Method: MOPS-Net

- **Experiments**
  - ➢Reconstruction-driven down-sampling

TABLE 4: Comparisons of the normalized reconstruction error (NRE) of different downsampling methods. The smaller, the better.

m	RS	FPS	S-Net	S-Net-M	MOPS-Net	MOPS-Net-M	FMOPS-Net-M
1024	1.013	1.000	1.090	1.000	1.030	1.000	1.005
512	1.096	1.014	1.084	1.018	1.096	1.019	1.068
256	1.340	1.084	1.124	1.086	1.055	1.061	1.128
128	2.226	1.330	1.172	1.207	1.059	1.101	1.276
64	4.089	2.030	1.419	1.535	1.140	1.270	1.541
32	7.702	3.767	2.677	2.867	2.182	2.457	2.136



Visual comparisons of the reconstructed point clouds by different downsampling methods with m = 64. The top row shows the sampled points (colored points) by different methods. The bottom row shows the reconstructed point clouds by different methods. (a) The original point cloud; (b) RS; (c) FPS; (d) S-Net (blue points) and S-Net-M (red points); and (e) MOPS-Net (blue points) and MOPS-Net-M (red points). Note the bottom row of (c) and (d) are the reconstructions by S-Net-M and MOPS-Net-M.

Proposed Down-sampling Method: MOPS-Net

- **Experiments: ablation studies**
  - The joint loss

TABLE 6: Classification accuracy of MOPS-Net and MOPS-Net-M when MOPS-Net was trained with different settings of $L_{dist}(\cdot,\cdot)$. Here, $m = 64$.

Setting of L_{dist}	MOPS-Net	MOPS-Net-M
$L_{dist} = L_{subset}$	0.883	0.457
$L_{dist} = L_{coverage}$	0.881	0.652
$L_{dist} = L_{subset} + \beta L_{coverage}$	0.871	0.810

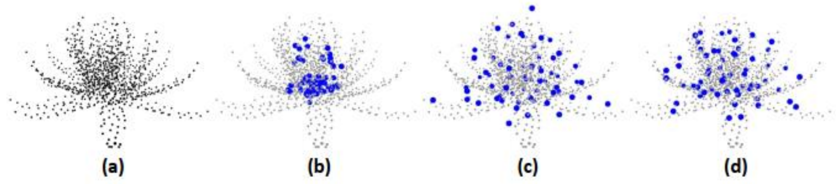


Fig. 7: Visual illustrations of the sampled points by MOPS-Net when trained with different settings of L_{dist}. $m = 64$. (a) The origin point cloud with 1024 points; (b) The sampled points by MOPS-Net when $L_{dist} = L_{subset}$; (c) The sampled points by MOPS-Net when $L_{dist} = L_{coverage}$; and (d) The sampled points by MOPS-Net when $L_{dist} = L_{subset} + \beta L_{coverage}$.

  - Appearance of he learned matrix $\boldsymbol{S}$

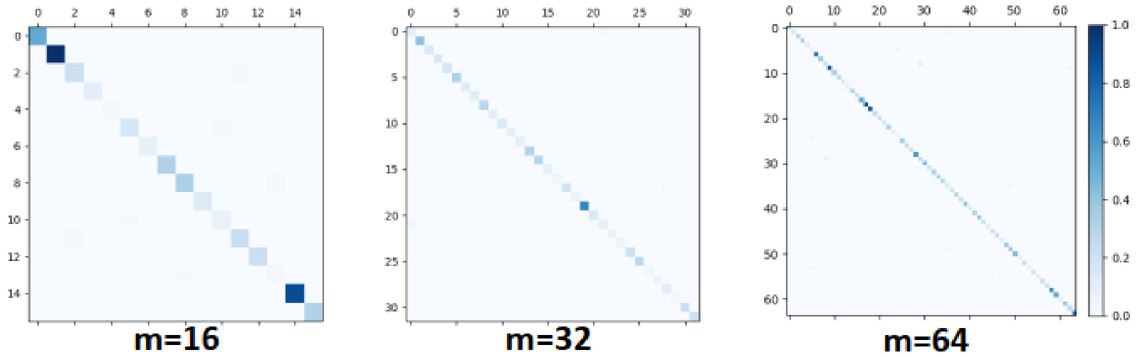


Fig. 8: Visual illustrations of the learned $\mathbf{S}^{\mathsf{T}}\mathbf{S}$ by MOPS-Net under various sample sizes.

  - Validation of the inverse mapping function

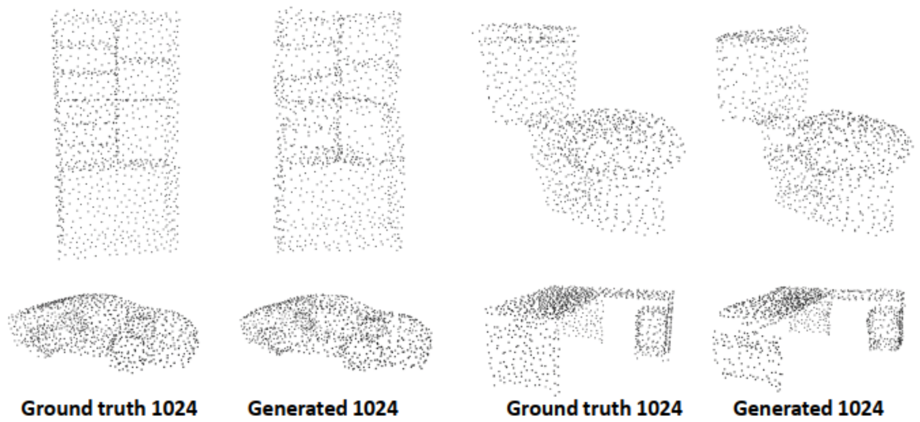


Fig. 10: Experimental verification that the learned function $\theta(\cdot)$ is an approximation of $\phi^{-1}(\cdot)$ in MOPS-Net.

Conclusions

香港城市大學
City University of Hong Kong

- We proposed the **first geometry-centric** deep neural network for 3D point cloud up-sampling, which is **essentially** different from the existing methods which are largely motivated by image super-resolution techniques.
- We presented MOPS-Net, a novel end-to-end deep learning framework for task-oriented point cloud down-sampling. In contrast to the existing methods, we designed MOPS-Net from the perspective of **matrix optimization**.
- Extensive experiments demonstrate the significant superiority of our methods over state-of-the-art approaches.
- Our methods not only brings new perspectives to the well-studied problem, but also links discrete differential geometry, matrix optimization, and deep learning in a more elegant way. we believe they has the potential for a wide range 3D processing tasks.