ChartPointFlow for Topology-Aware 3D Point Cloud Generation

Takumi Kimura[1], Takashi Matsubara[2], Kuniaki Uehara[3]

[1]Kobe University, Japan

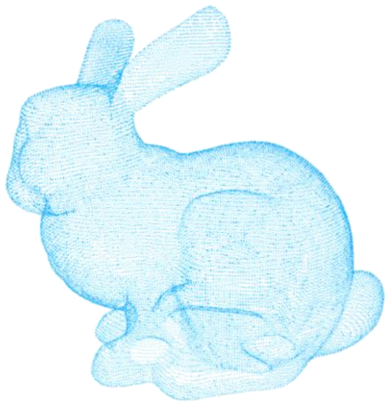
[2]Osaka University, Japan

[3]Osaka Gakuin University, Japan

Point Cloud

Point Cloud

- 3D data representation in Cartesian coordinates
- higher resolution than voxels
- easier to manipulate and obtain than mesh



It's generative model is useful for

- shape completion
- shape synthesis
- super-resolution

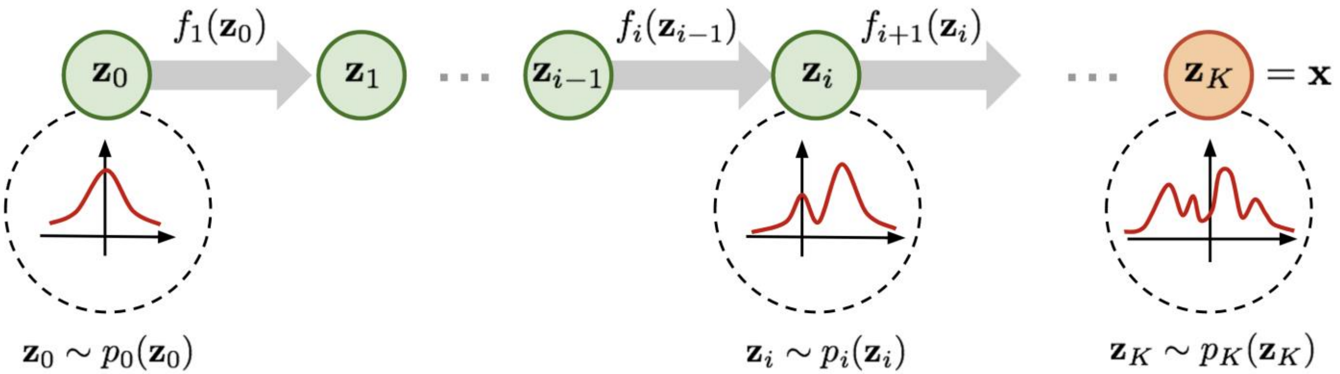
➢ Improve point cloud generation performance

Related work: Flow-based Generative Model

Flow-based Generative Model (Flow)

- is one of the deep generative models
- represents complex distributions by iteratively transforming random variables
- directly maximizes log-likelihood
- can do inference and generation using a single model

$$x = f(z), \qquad z = f^{-1}(x), \qquad f = f_n \circ f_{n-1} \cdots f_1$$

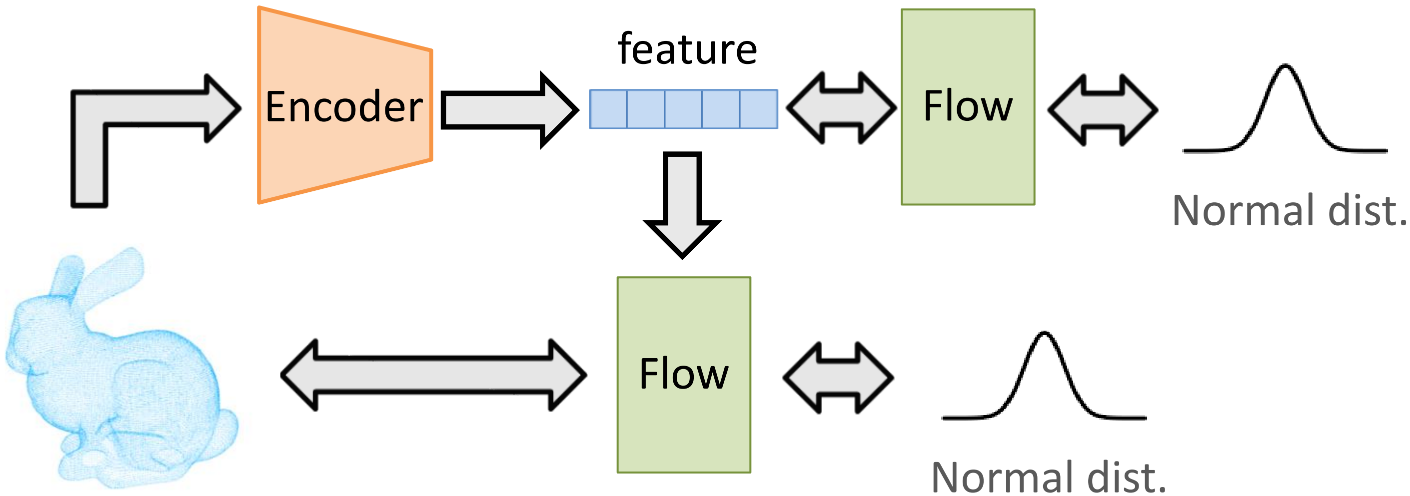


L.Weng. "Flow-based Deep Generative Models" https://lilianweng.github.io/lil-log/2018/10/13/flow-based-deep-generative-models.html

Related work: PointFlow

PointFlow [Yang+, ICCV2019]

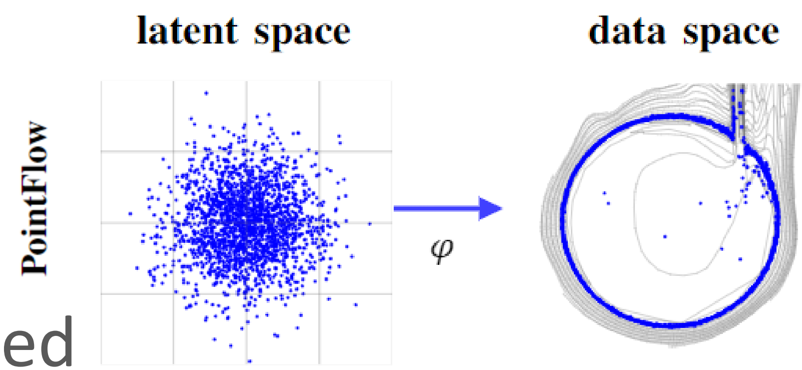
- uses Flows for point cloud generation
- has two Flows
  - ➢ One for the distribution of shape
  - ➢ The other for the distribution of points given the shape
- can generate arbitrary number of points



Proposed Method: ChartPointFlow

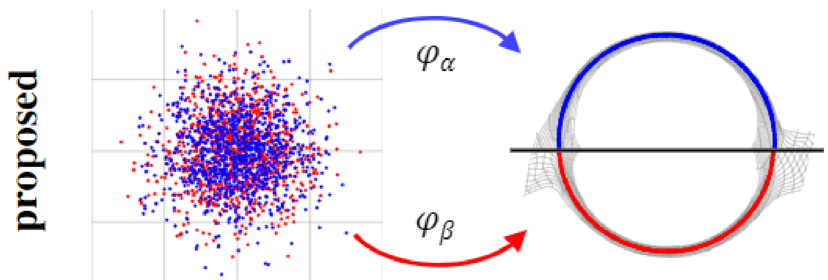
PointFlow

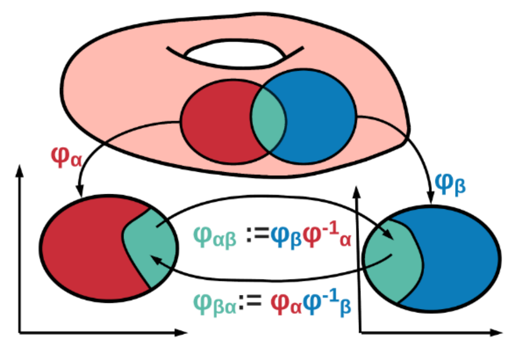
- fails in generating a point clouds of a different topology



Proposed

- generates a point cloud as a manifold, covered by multiple **charts**



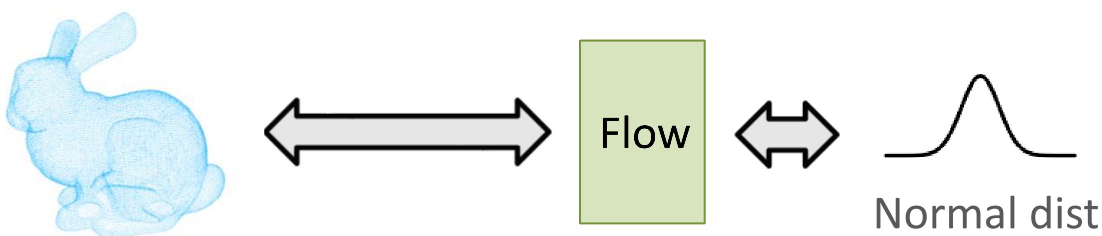


[Schonscheck+, Chart Auto-Encoders, arXiv2019]

Proposed Method: ChartPointFlow

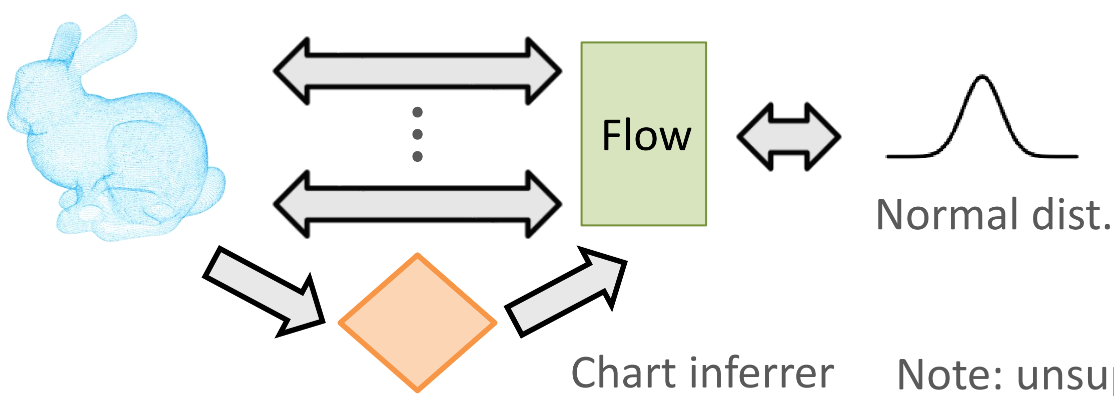
PointFlow

- fails in generating a point clouds of a different topology



Proposed

- generates a point cloud as a manifold, covered by multiple **charts**



Note: unsupervised

Proposed Method: ChartPointFlow

Training Objective

- Log-likelihood for PointFlow: $\log p_F(x)$
- Marginal log-likelihood for ChartPointFlow:

$$\log p_F(x) \geq \mathbb{E}_{q_C(y|x)}[\log p_F(x|y)] - H[q_C(y|x)|p(y)] + H[q_C(y|x)]$$

Averaged over all possible labels

➢ many charts -> high computational and memory costs

First trick

- Use **Gumbel-Softmax** [Jang+, ICLR2017] to infer a label y in a differentiable manner

$$\tilde{y} = \mathrm{softmax}((\log \pi_C(x) + g) \,/\, \tau)$$

$g \sim \mathrm{Gumbel}(0,1)$

τ : temperature

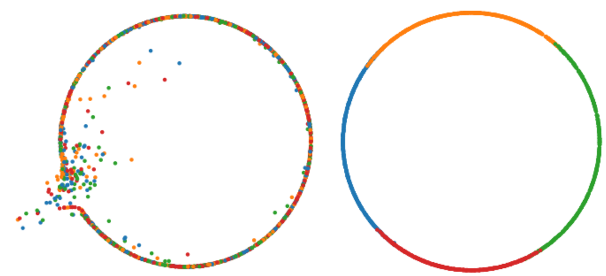
Proposed Method: ChartPointFlow

Training Objective

- Log-likelihood for PointFlow: $\log p_F(x)$
- Marginal log-likelihood for ChartPointFlow:

$$\log p_F(x) \geq \mathbb{E}_{q_C(y|x)}[\log p_F(x|y)] - H[q_C(y|x)|p(y)] + \underline{H[q_C(y|x)]}$$

  - Maximizing $H[q_C(y|x)]$ makes each point assigned to all labels uniformly
  - The charts overlap with each other



Color represents the chart

Second trick

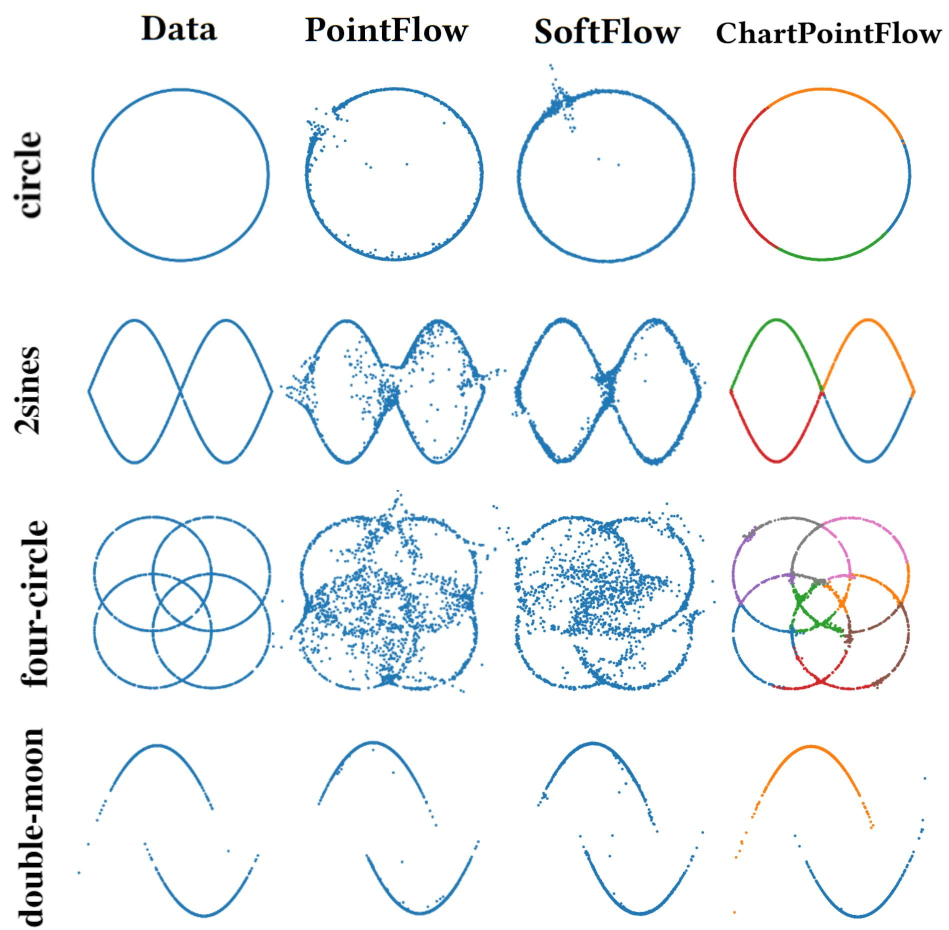
- Use the **regularization term** to avoid the charts overlapping

$$I(y;x) = H[q_C(y)] - H[q_C(y|x)]$$

2D Experiment: Generation results

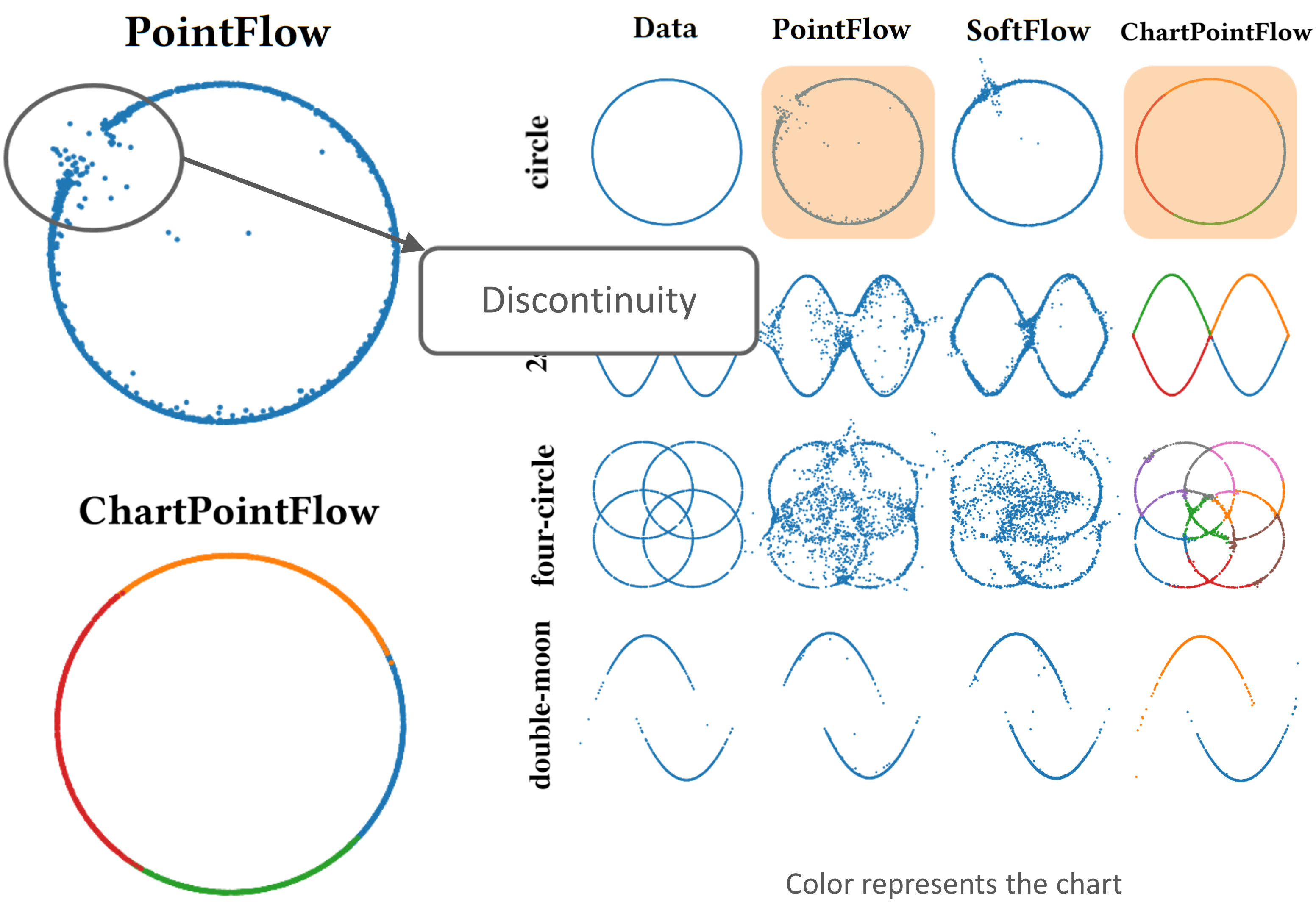
Dataset

- Synthetic data



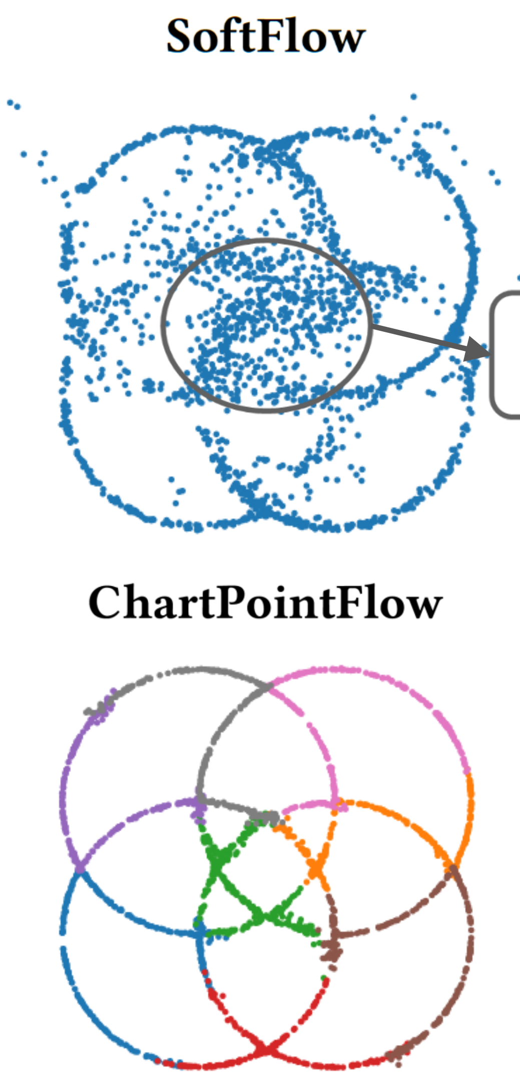
Color represents the chart

2D Experiment: Generation results

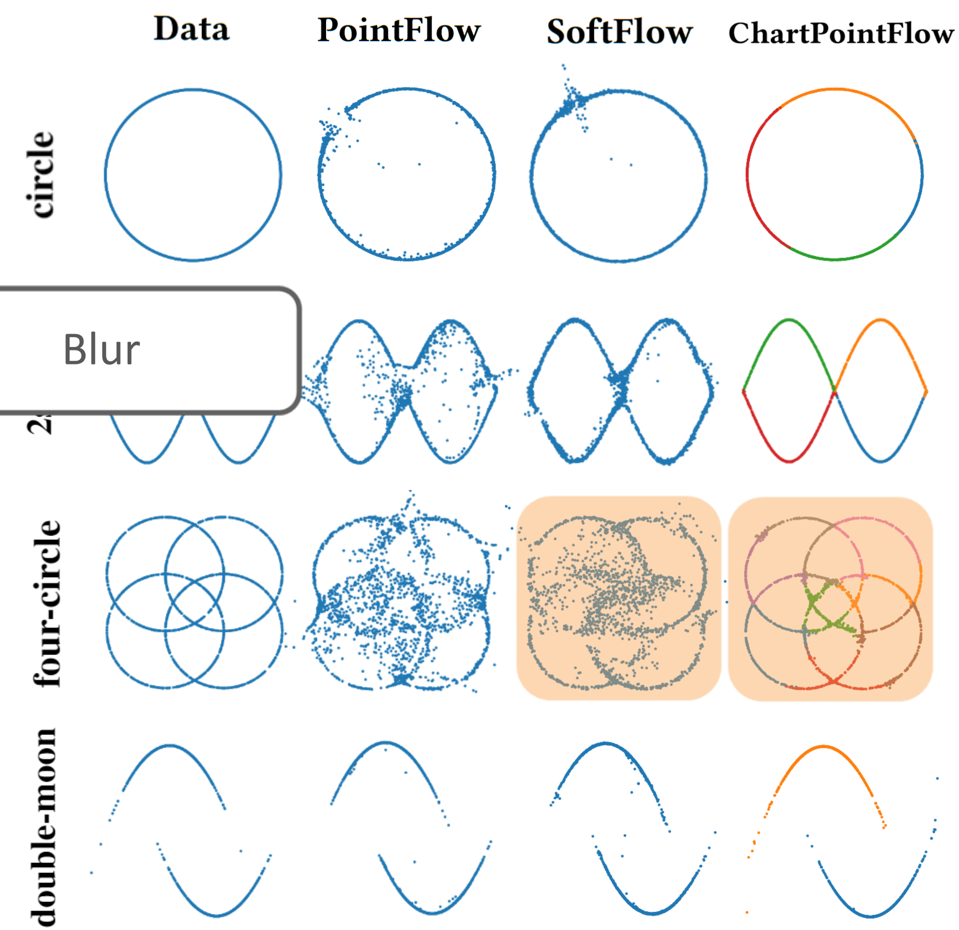


Color represents the chart

2D Experiment: Generation results

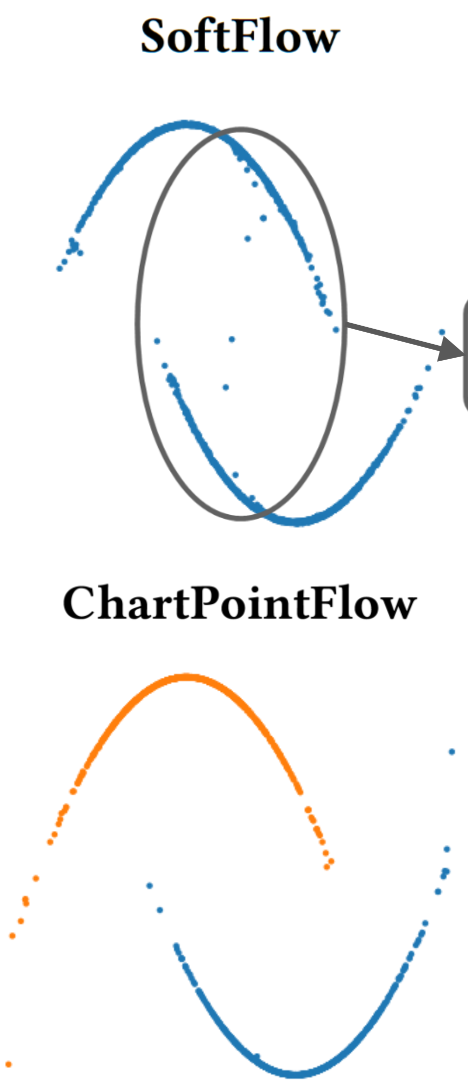


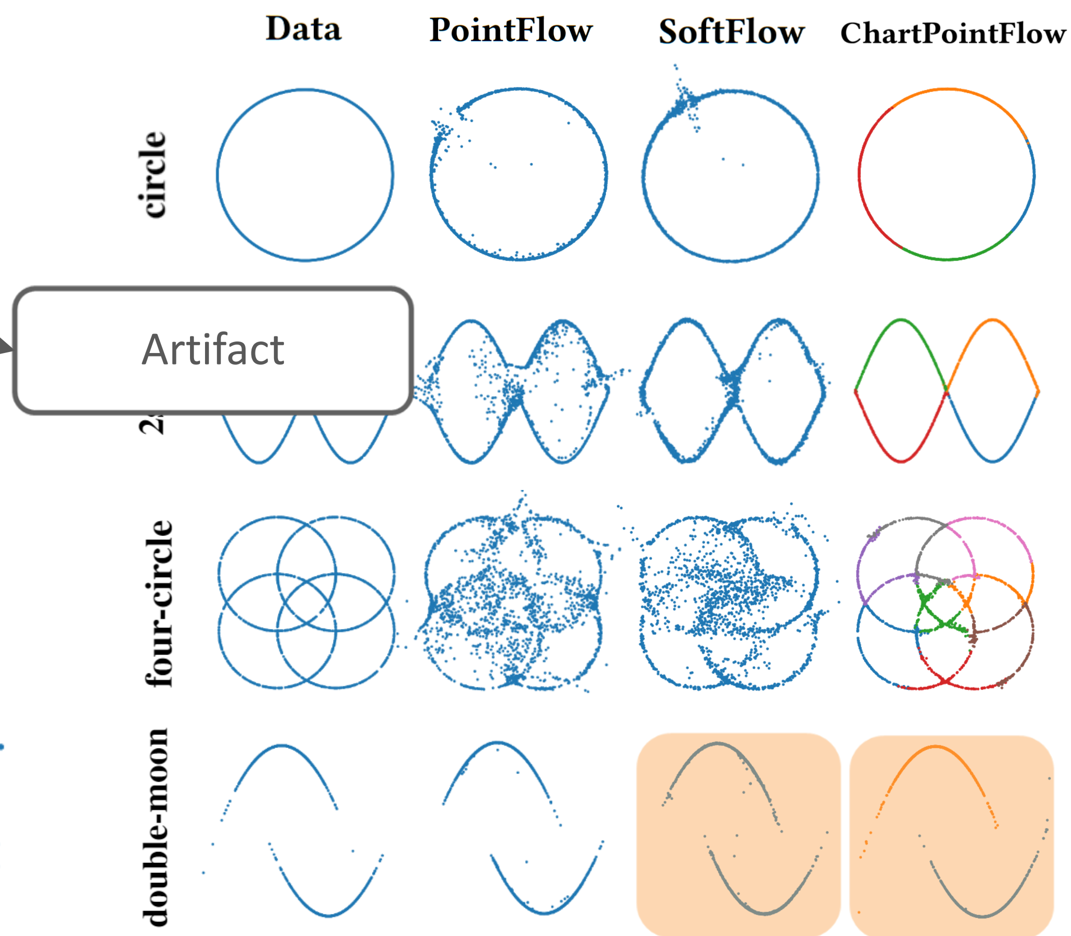
Blur



Color represents the chart

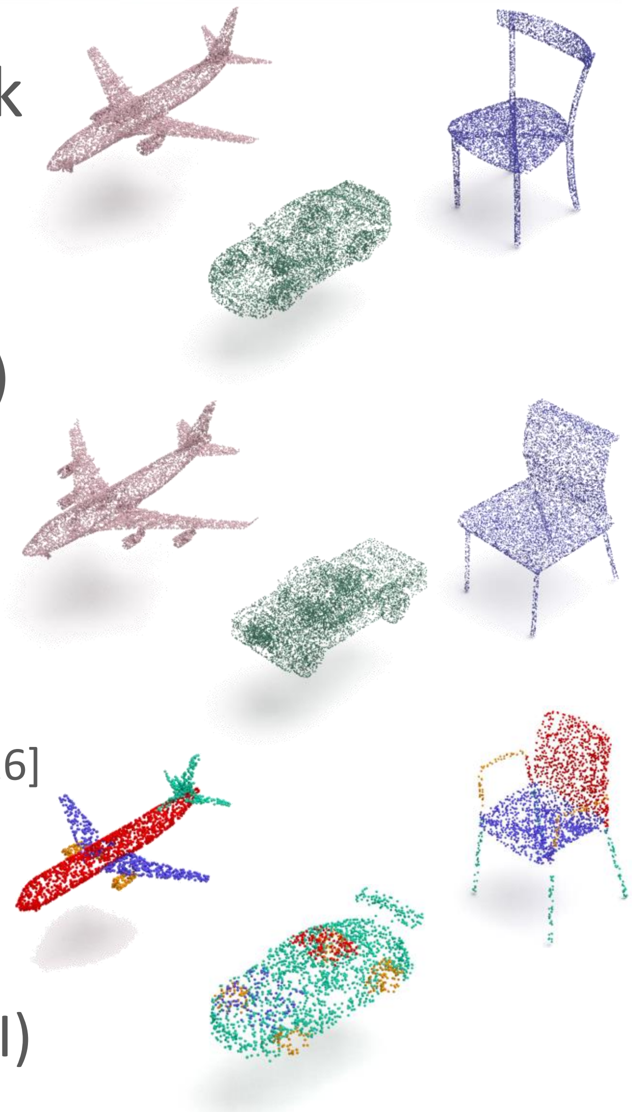
2D Experiment: Generation results





Color represents the chart

3D Experiment: Dataset, Evaluation metrics



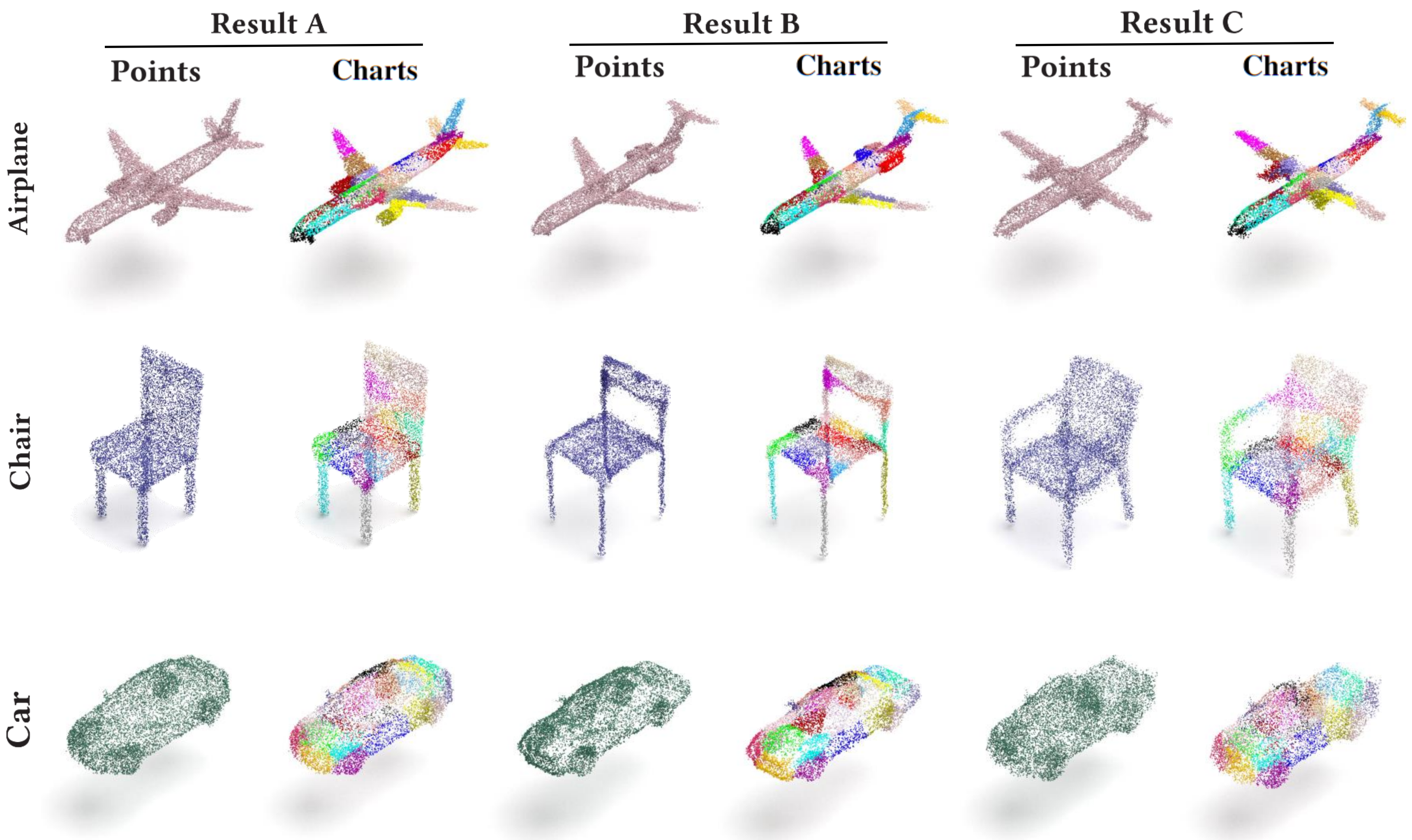
Generation and Reconstruction task

- Dataset: ShapeNet [Chang+, arXiv2015]
- Evaluation metrics
  - 1-nearest neighbor accuracy (1-NNA)
  - earth mover’s distance (EMD)

Unsupervised Segmentation

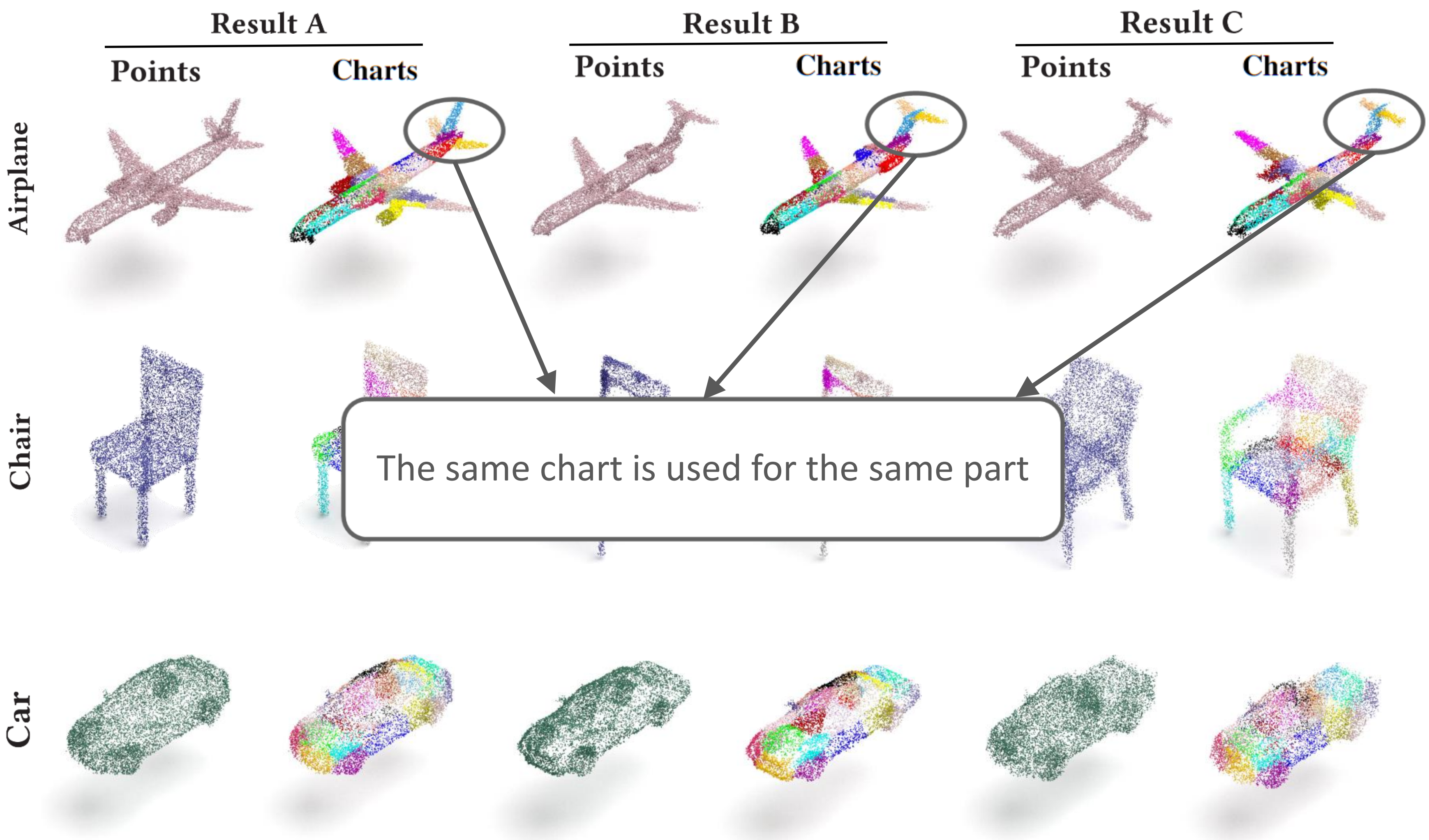
- Dataset: PartDataset [Yi+, SIGGRAPH Asia2016]
- Evaluation metrics
  - purity (PUR)
  - Normalized mutual information (NMI)

3D Experiment: Results (Generation)



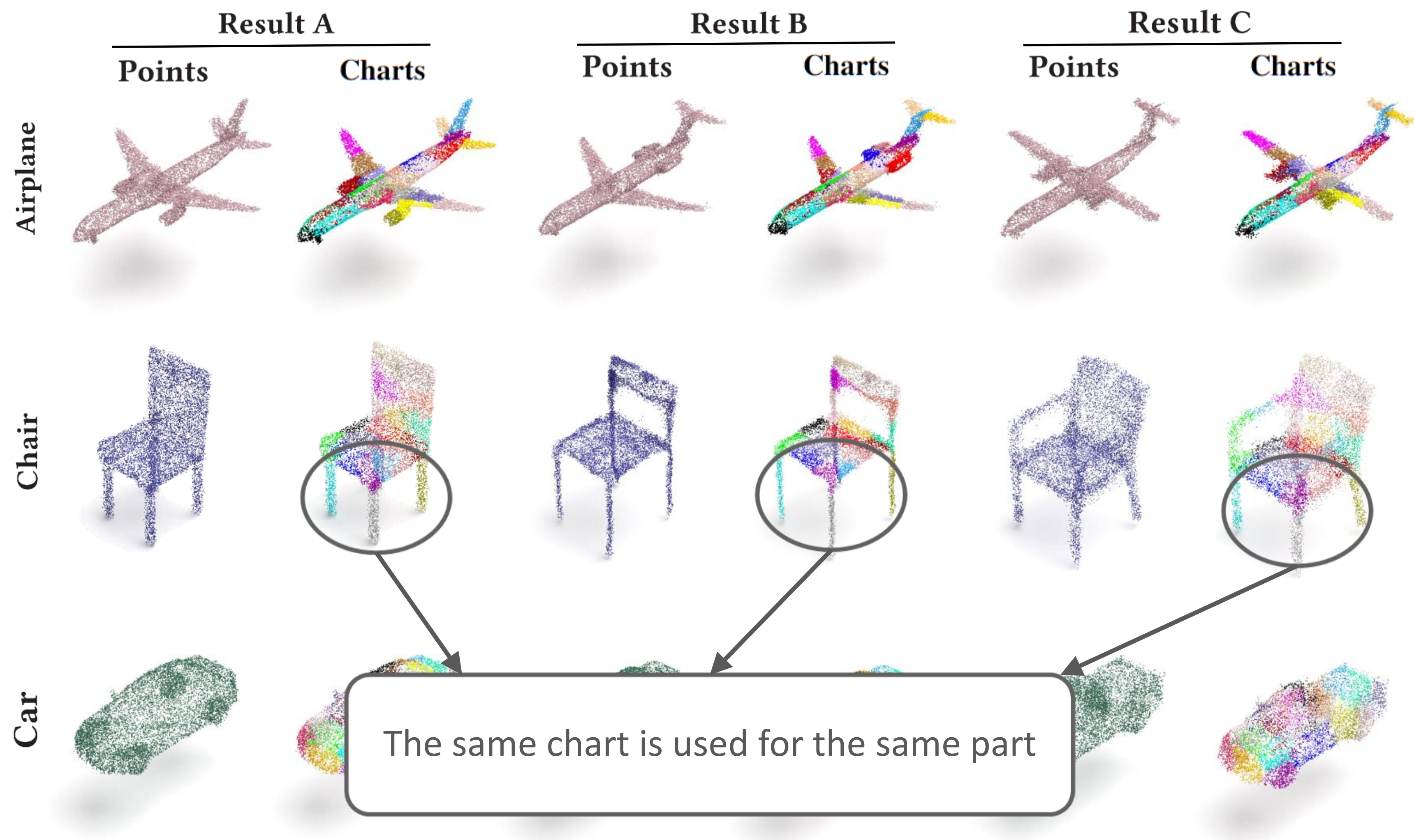
The samples generated by ChartPointFlow

3D Experiment: Results (Generation)



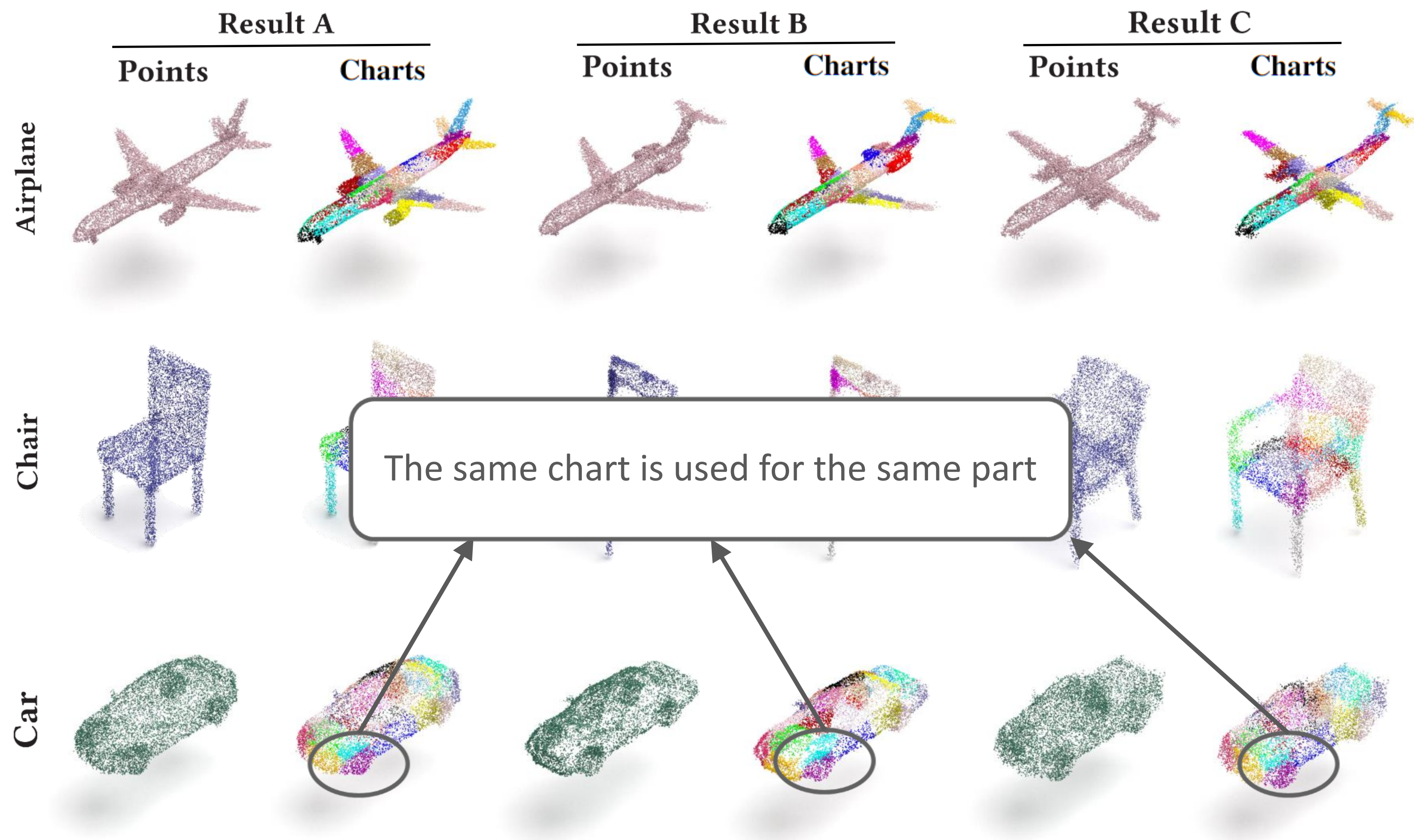
The samples generated by ChartPointFlow

3D Experiment: Results (Generation)



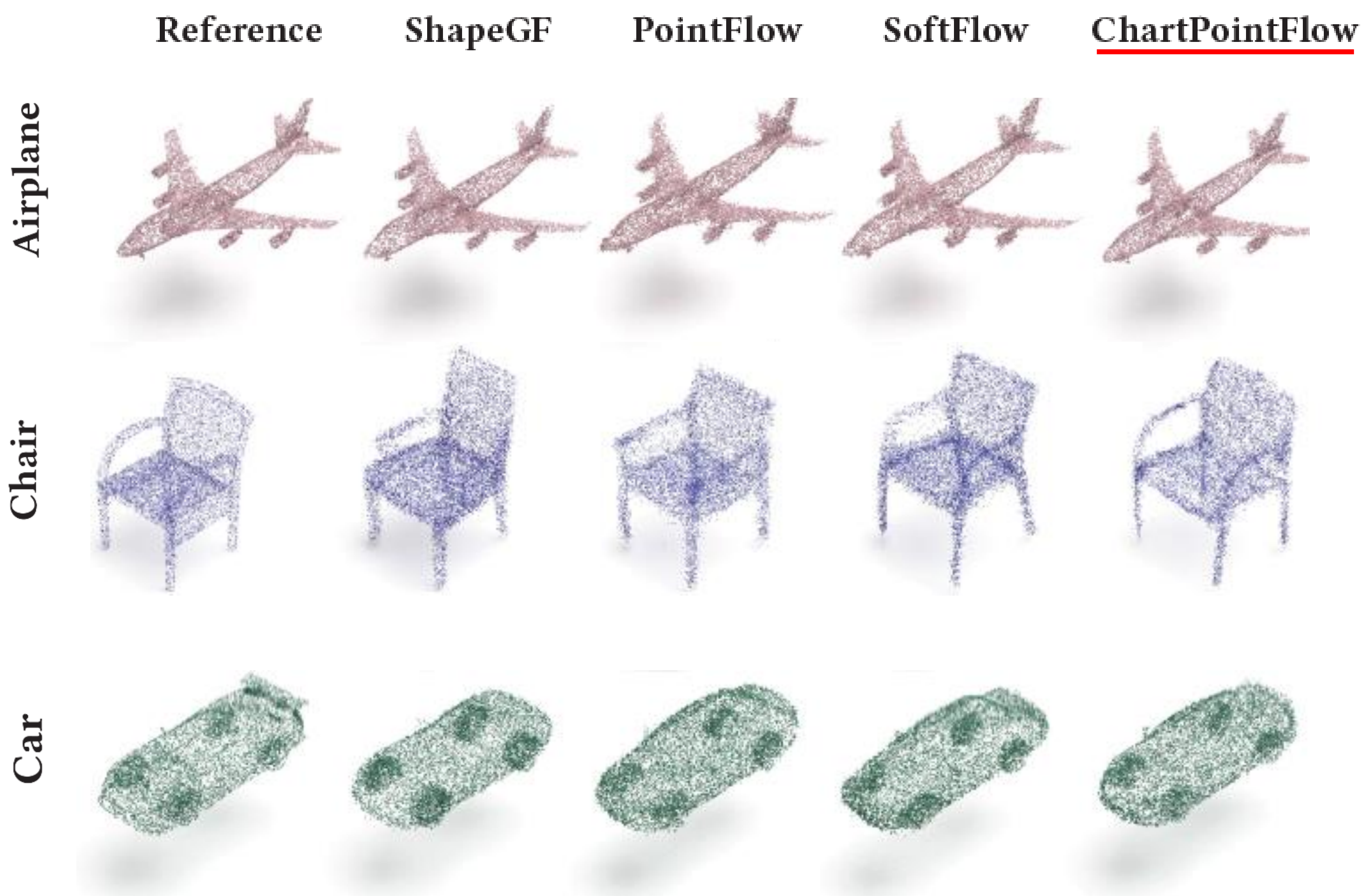
The samples generated by ChartPointFlow

3D Experiment: Results (Generation)



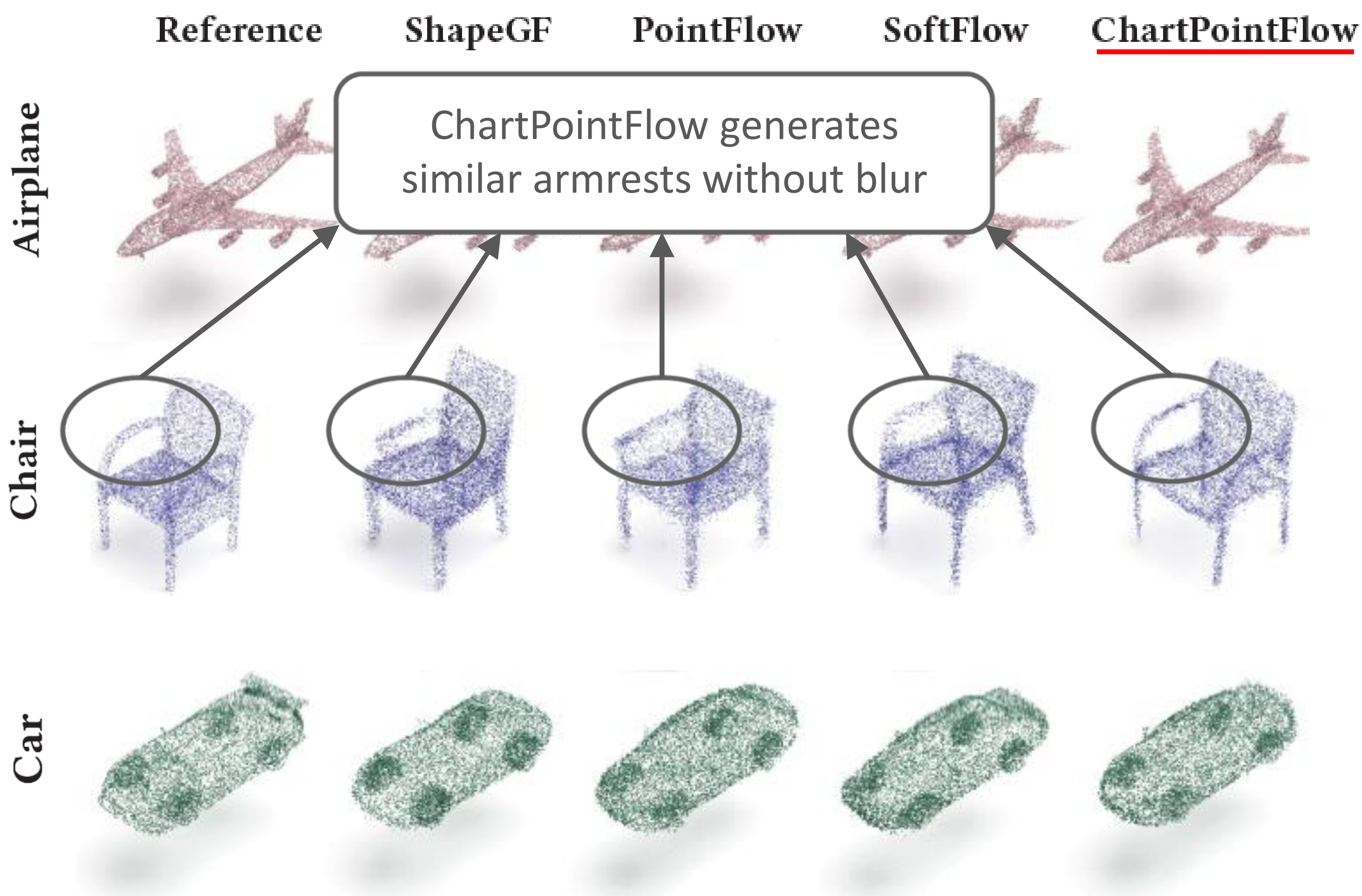
The samples generated by ChartPointFlow

3D Experiment: Comparisons (Generation)



The samples nearest to the reference generated by each model

3D Experiment: Comparisons (Generation)



The samples nearest to the reference generated by each model

3D Experiment: Comparisons (Generation)

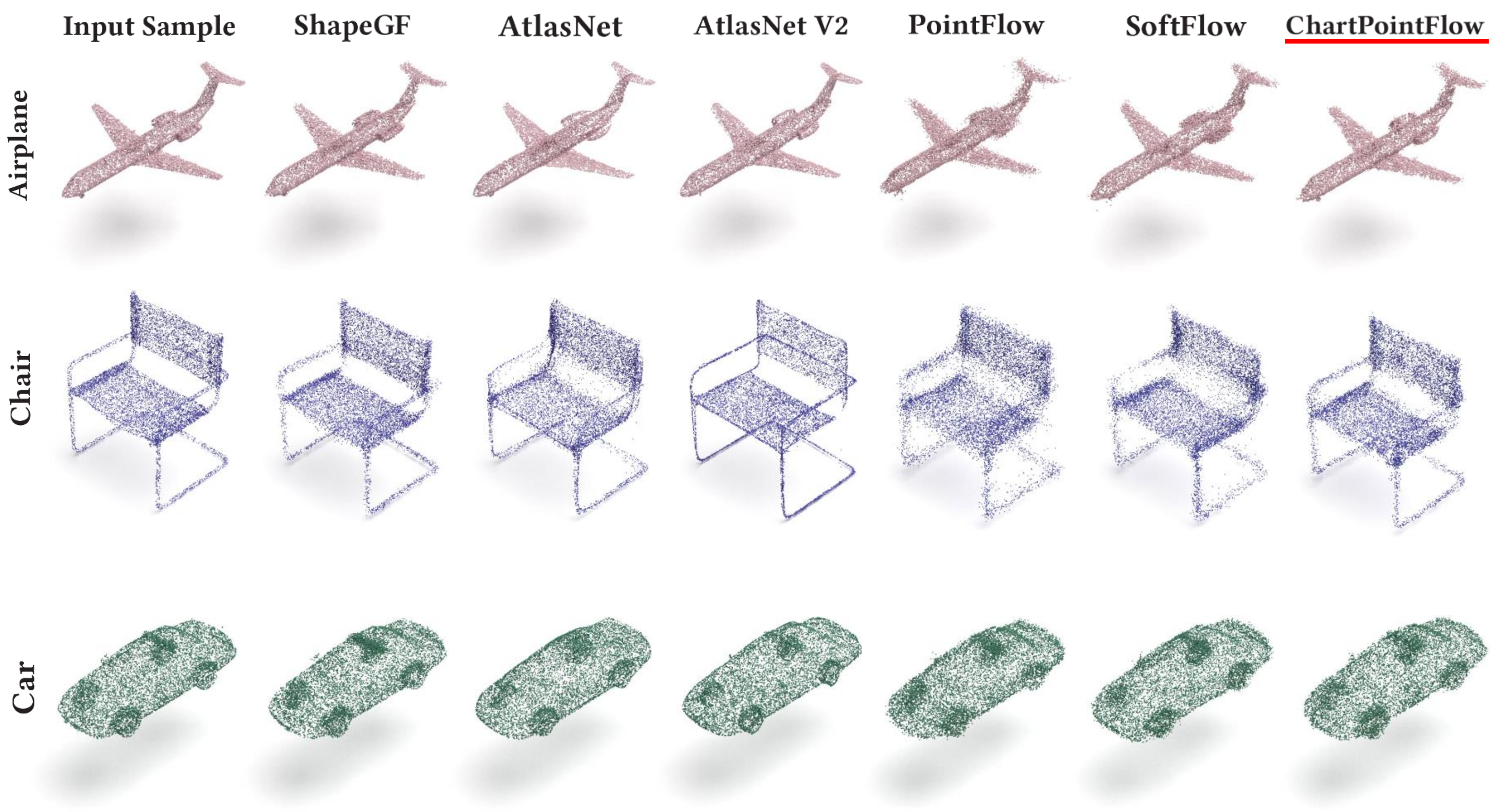
ChartPointFlow outperforms other state-of-the-art point cloud generators

Model	1-NNA(%, ↓) Airplane	Chair	Car
r-GAN	99.51	99.47	99.86
l-GAN (CD)	97.28	85.27	88.07
l-GAN (EMD)	85.68	65.56	68.32
PC-GAN	92.32	78.37	90.87
ShapeGF	81.44	59.60	60.31
PointFlow	75.06	59.89	62.36
SoftFlow	69.44	63.51	64.71
ChartPointFlow	**65.08**	**58.31**	**58.68**

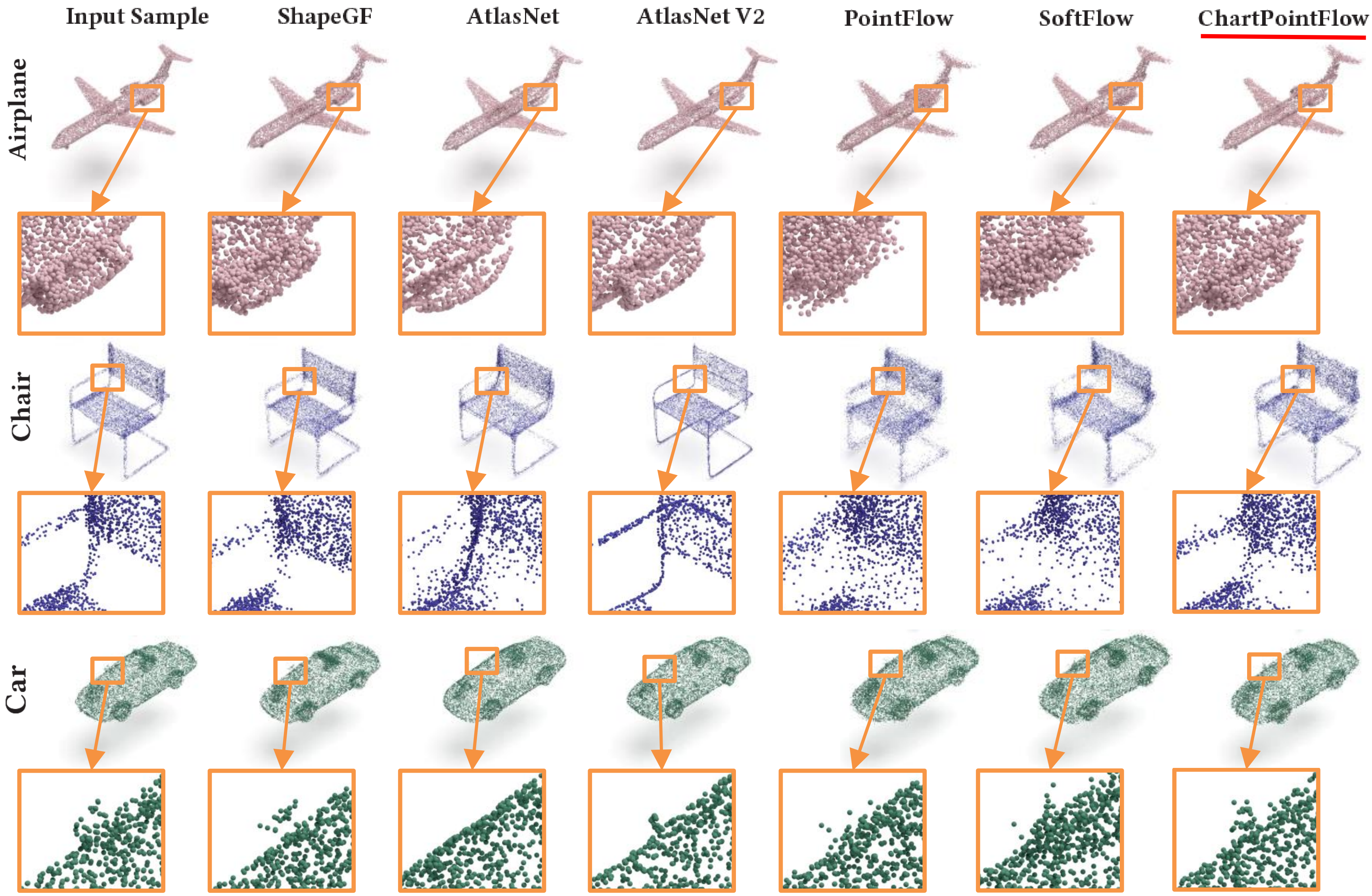
+ chart

3D Experiment: Comparisons (Reconstruction)

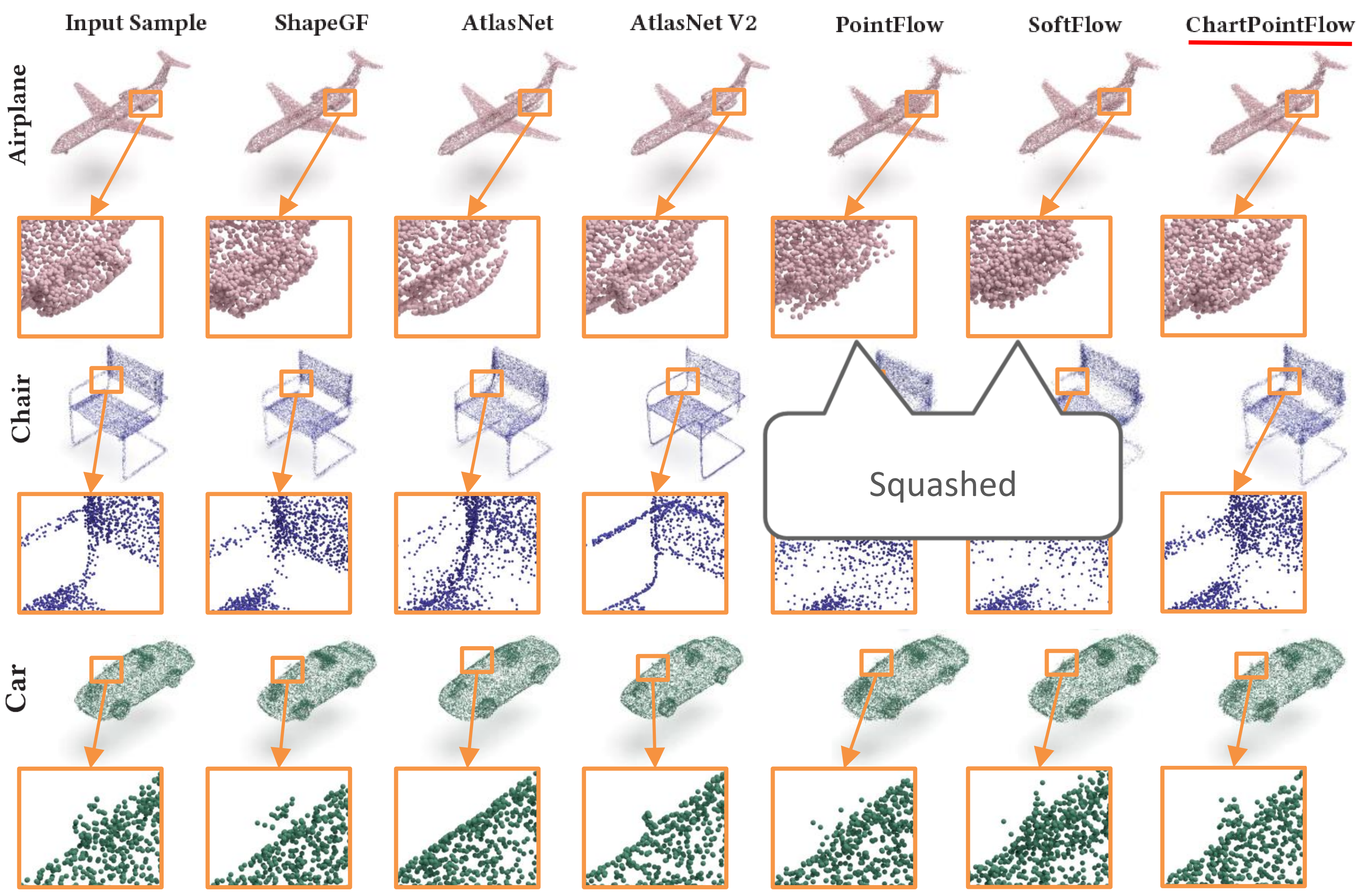
The examples reconstructed by each model



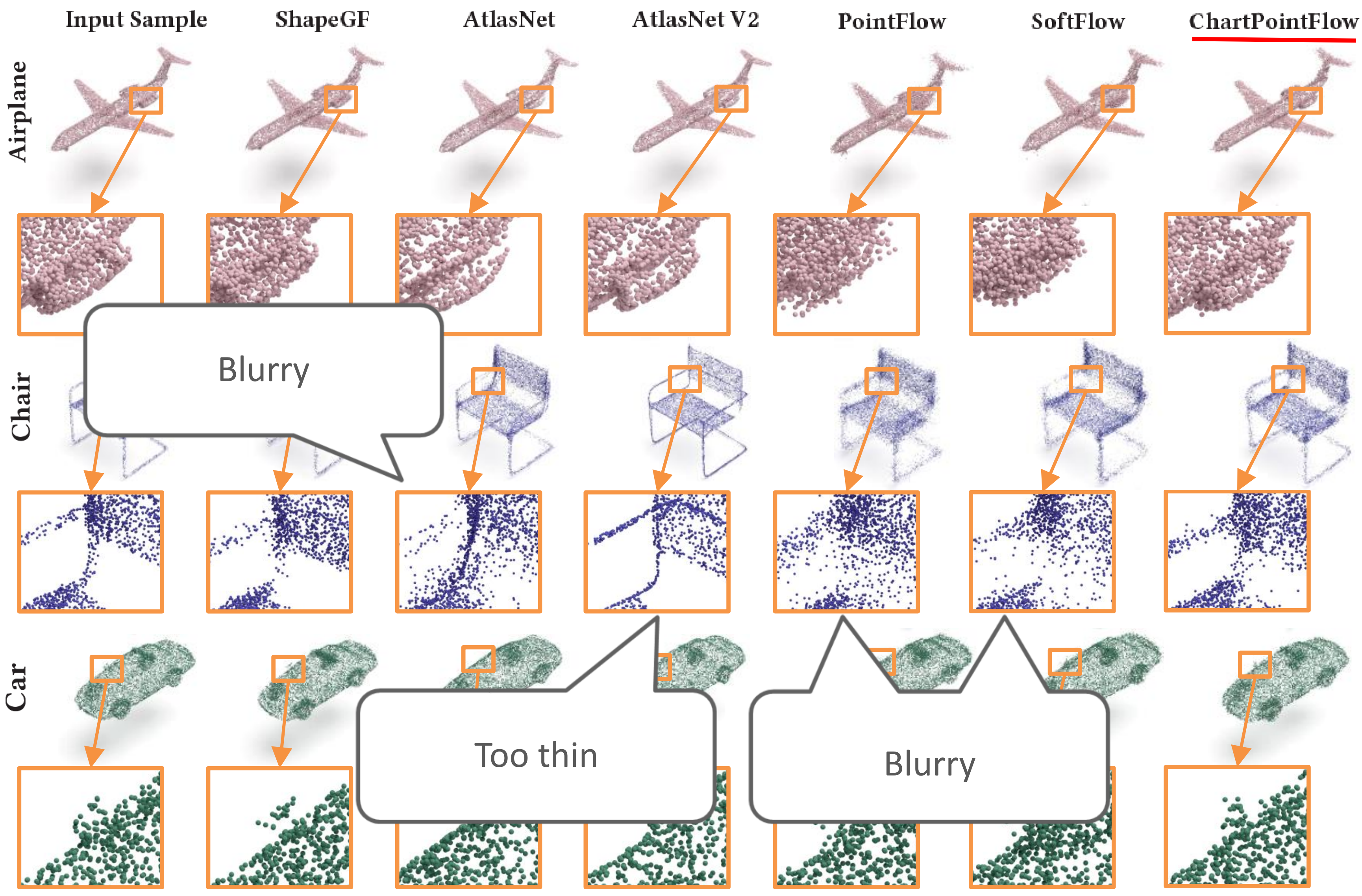
3D Experiment: Comparisons (Reconstruction)



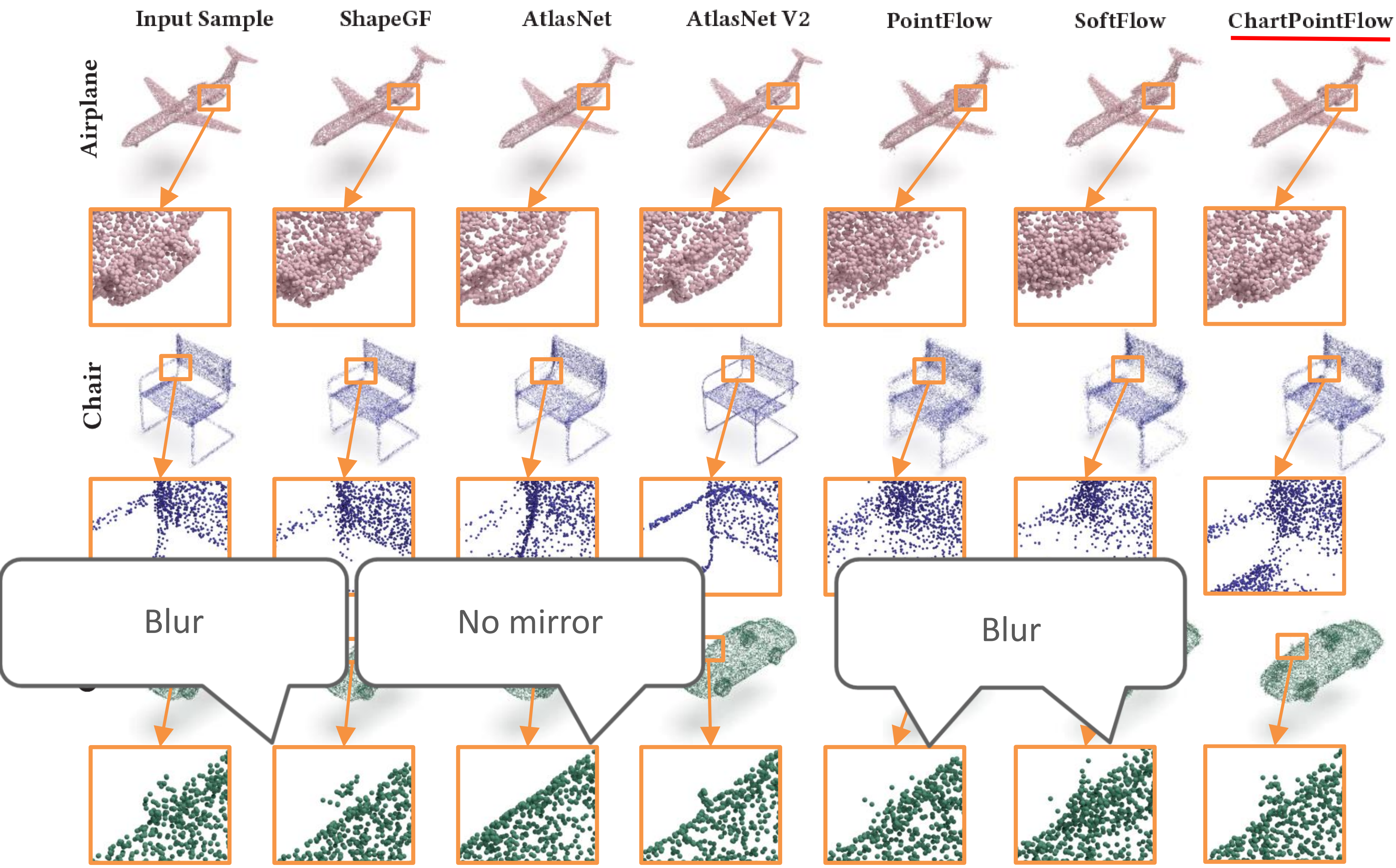
3D Experiment: Comparisons (Reconstruction)



3D Experiment: Comparisons (Reconstruction)



3D Experiment: Comparisons (Reconstruction)



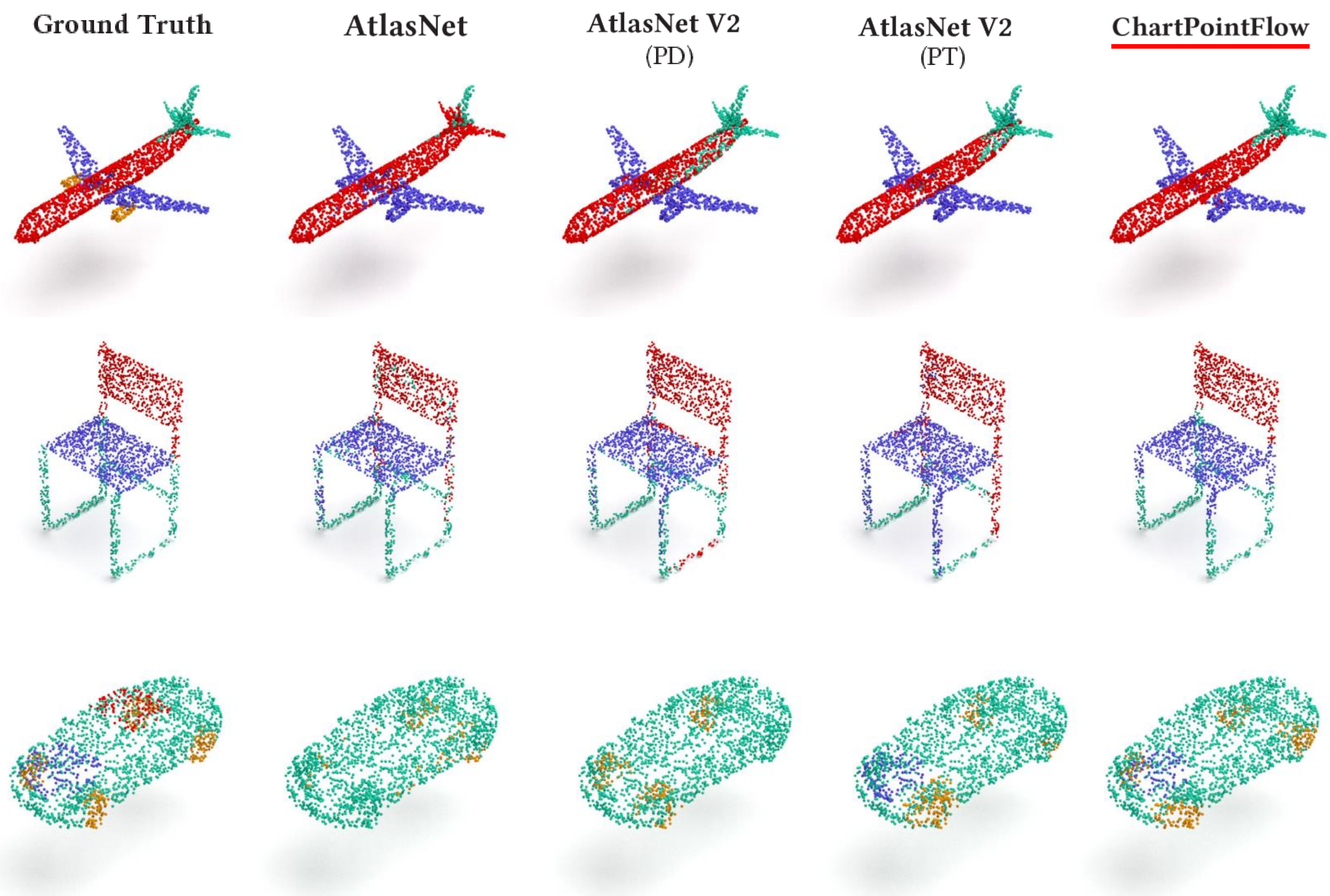
3D Experiment: Comparisons (Reconstruction)

The EMD between a reference point cloud and a reconstructed one

Model	Airplane	Chair	Car
ShapeGF	2.55	5.22	4.63
AtlasNet	2.95	6.68	4.75
AtlasNet V2 (PD)	3.28	5.67	4.51
AtlasNet V2 (PT)	3.57	5.97	5.13
PointFlow	2.77	6.42	5.16
SoftFlow	2.60	6.60	5.08
ChartPointFlow	**2.23**	**4.62**	**3.96**

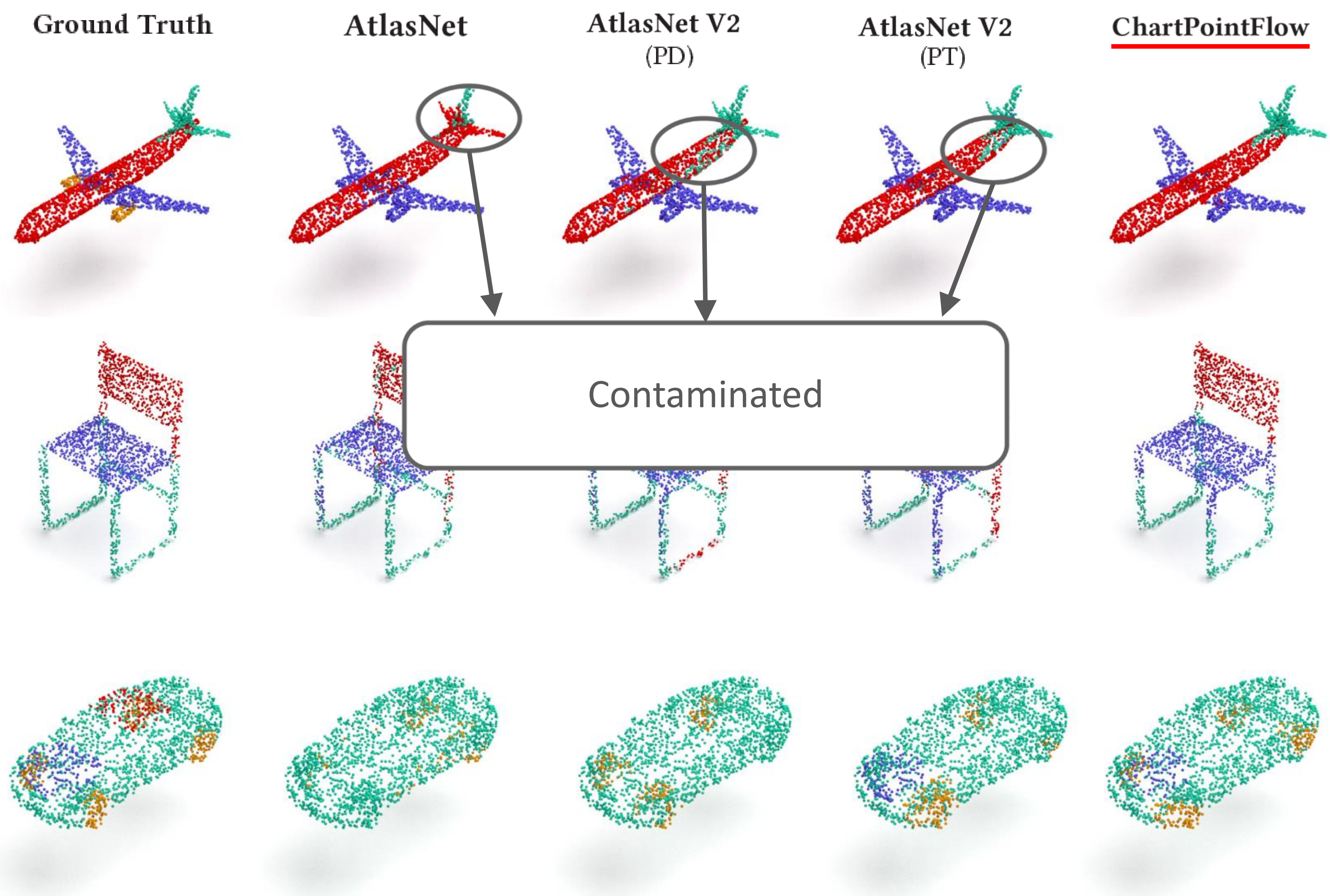
Significantly improved, especially for chair category

3D Experiment: Comparisons (Segmentation)



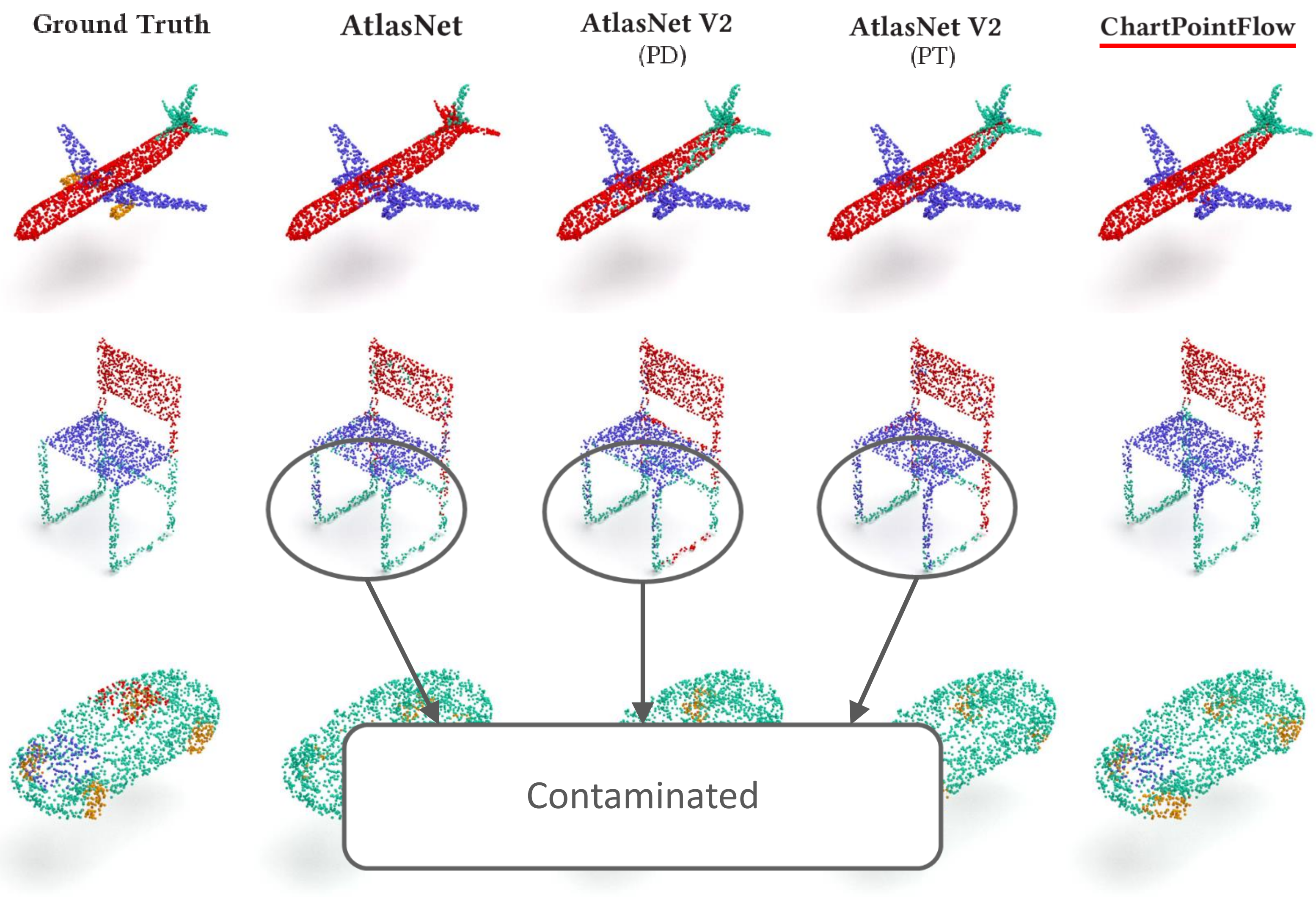
The results of unsupervised segmentation

3D Experiment: Comparisons (Segmentation)



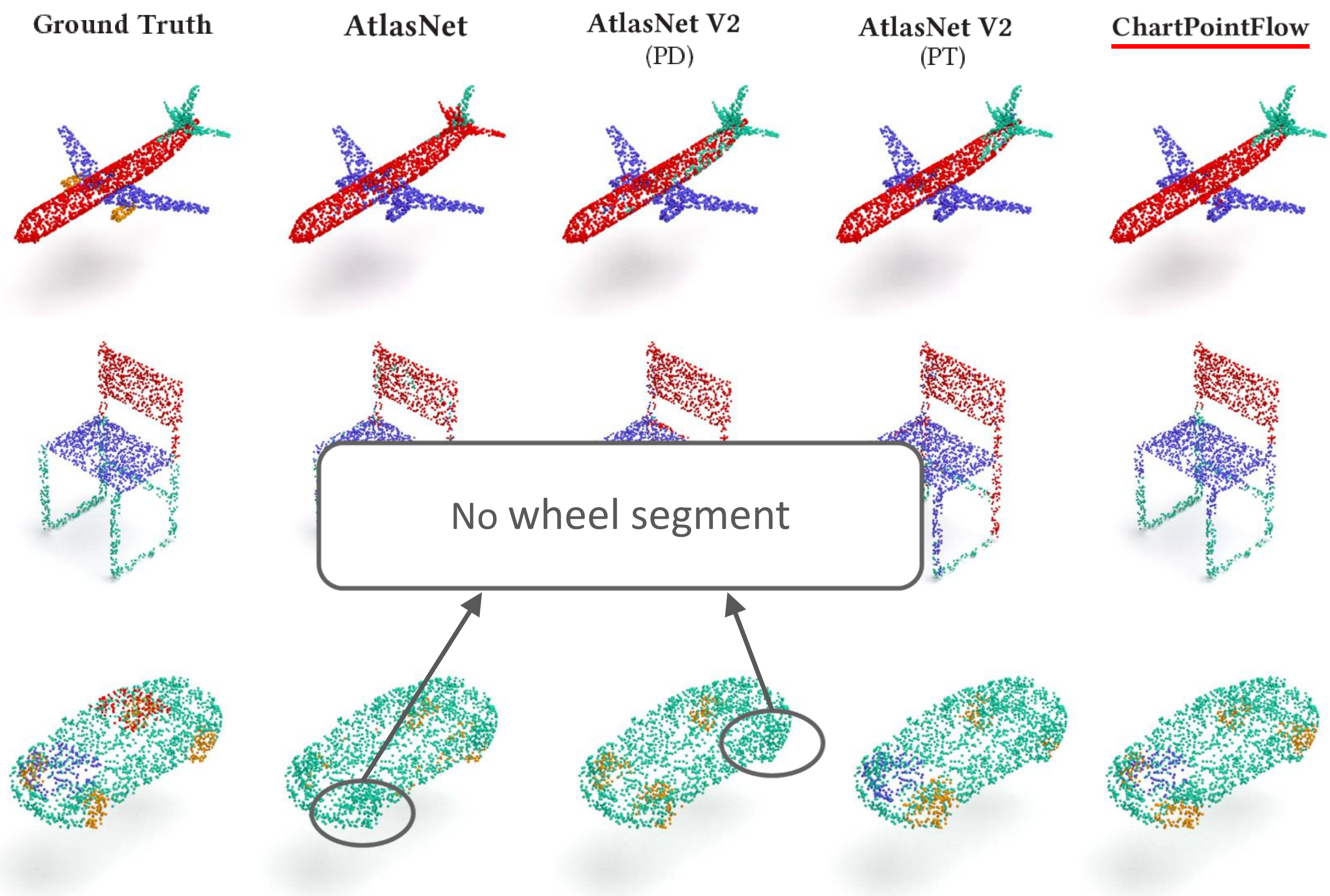
The results of unsupervised segmentation

3D Experiment: Comparisons (Segmentation)



The results of unsupervised segmentation

3D Experiment: Comparisons (Segmentation)



The results of unsupervised segmentation

3D Experiment: Comparisons (Segmentation)

ChartPointFlow outperformed AtlasNets for both criteria in all categories

Model	Airplane	Chair	Car
AtlasNet	0.22 / 0.76	0.23 / 0.74	0.11 / 0.71
AtlasNet V2 (PD)	0.25 / 0.79	0.24 / 0.75	0.13 / 0.72
AtlasNet V2 (PT)	0.27 / **0.80**	0.24 / 0.74	0.17 / 0.73
ChartPointFlow	**0.30** / **0.80**	**0.35** / **0.86**	**0.19** / **0.79**

Segmentation performances (NMI/purity)

Conclusion

- ChartPointFlow
  - a flow-based generative model
  - with **multiple latent labels**
  - Each label is assigned to points in an **unsupervised** manner
  - Owing to **Gumbel-Softmax** and **regularization**
    - ✓ reduce the computational cost
    - ✓ each chart is assigned a specific connected region

- Experiments:
  - ChartPointFlow preserves the topological structure
  - ChartPointFlow outperforms other point cloud generators

Appendix

Metrics

$$CD(X_1, X_2) = \sum_{x \in X_1} \min_{\xi \in X_2} \|x - \xi\|_2^2 + \sum_{x \in X_2} \min_{\xi \in X_1} \|x - \xi\|_2^2,$$

$$EMD(X_1, X_2) = \min_{\phi: X_1 \to X_2} \sum_{x \in X_1} \|x - \phi(x)\|_2,$$

X_1, X_2: point cloud

ϕ: bijective map

$$\text{1-NNA}(\mathcal{X}_1, \mathcal{X}_2) = \frac{\sum_{X_1 \in \mathcal{X}_1} \mathbb{1}[N_{X_1} \in \mathcal{X}_1] + \sum_{X_2 \in \mathcal{X}_2} \mathbb{1}[N_{X_2} \in \mathcal{X}_2]}{|\mathcal{X}_1| + |\mathcal{X}_2|},$$

$\mathcal{X}_1, \mathcal{X}_2$: point sets

$\mathbb{1}[\cdot]$: indicator function

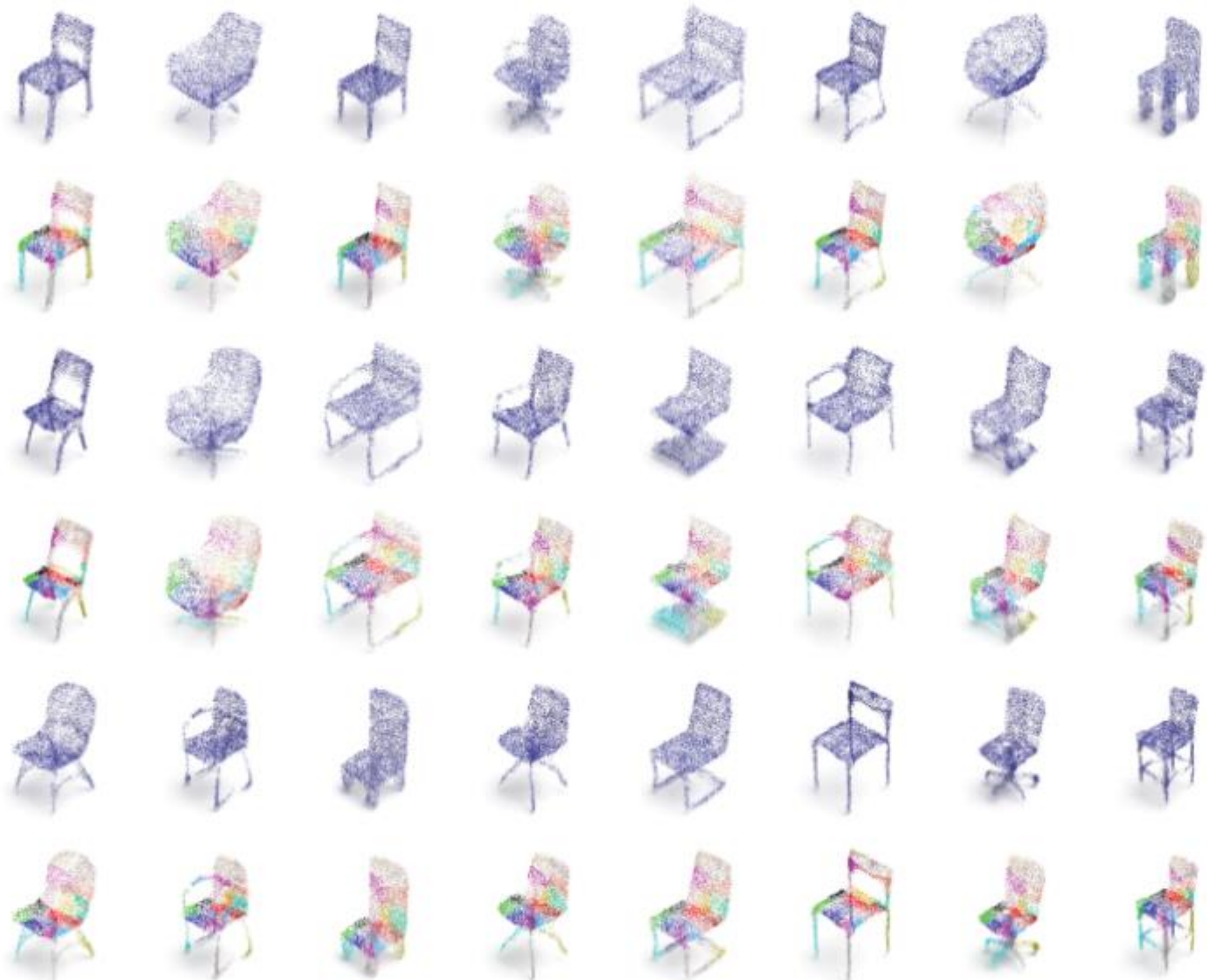
$N_{X_\blacksquare}$: the nearest neighbor of $X_\blacksquare$ in $\mathcal{X}_1 \cup \mathcal{X}_2 - \{X_\bullet\}$

Additional Results (Generation)

Category	Number of Charts	JSD(↓)	MMD(↓)		COV(%, ↑)		1-NNA(%)	
			CD	EMD	CD	EMD	CD	EMD
Airplane	1	3.54	0.221	3.15	49.63	**53.21**	72.67	68.90
	4	3.62	0.220	3.11	48.89	51.79	71.77	67.30
	8	3.39	0.217	3.08	49.66	51.70	70.90	66.54
	12	3.60	0.213	3.06	48.40	51.73	70.20	65.99
	16	3.93	0.215	3.07	49.52	51.08	70.72	66.48
	20	3.82	0.218	3.09	48.10	51.02	71.20	66.53
	24	**3.01**	0.214	3.06	50.20	51.79	**69.39**	65.62
	28	3.49	0.213	3.05	**50.57**	52.35	69.48	**65.08**
	32	3.46	**0.211**	**3.04**	49.69	51.82	70.59	66.08
Chair	1	1.96	2.50	8.06	43.04	45.38	59.75	63.16
	4	1.96	2.48	7.90	43.45	44.30	59.64	61.79
	8	1.82	2.50	7.86	43.85	44.76	58.76	60.44
	12	1.89	2.54	7.87	44.82	45.50	58.37	59.96
	16	**1.57**	2.48	7.78	45.37	**46.03**	58.04	59.51
	20	1.83	2.52	7.84	**45.61**	45.85	**57.89**	**58.31**
	24	1.97	2.53	7.87	45.05	45.69	58.20	59.29
	28	1.97	2.45	**7.75**	43.76	45.78	58.40	58.94
	32	1.63	**2.44**	7.79	44.23	45.42	59.52	60.76
Car	1	0.96	0.95	5.25	44.98	47.78	61.86	61.56
	4	0.93	0.92	5.17	46.20	46.86	60.94	60.48
	8	0.91	0.90	5.15	45.42	46.45	60.04	60.84
	12	0.93	0.92	5.14	44.76	46.31	59.50	59.76
	16	0.86	0.91	5.13	**46.41**	**48.81**	**58.13**	58.80
	20	**0.83**	0.92	5.14	45.38	46.89	59.10	59.65
	24	0.87	0.94	5.14	44.83	47.66	59.42	**58.68**
	28	0.90	0.94	5.12	44.50	46.06	60.49	59.67
	32	**0.83**	**0.89**	**5.07**	45.81	48.08	58.96	58.75

Additional Results (Generation)

Additional Results (Generation)



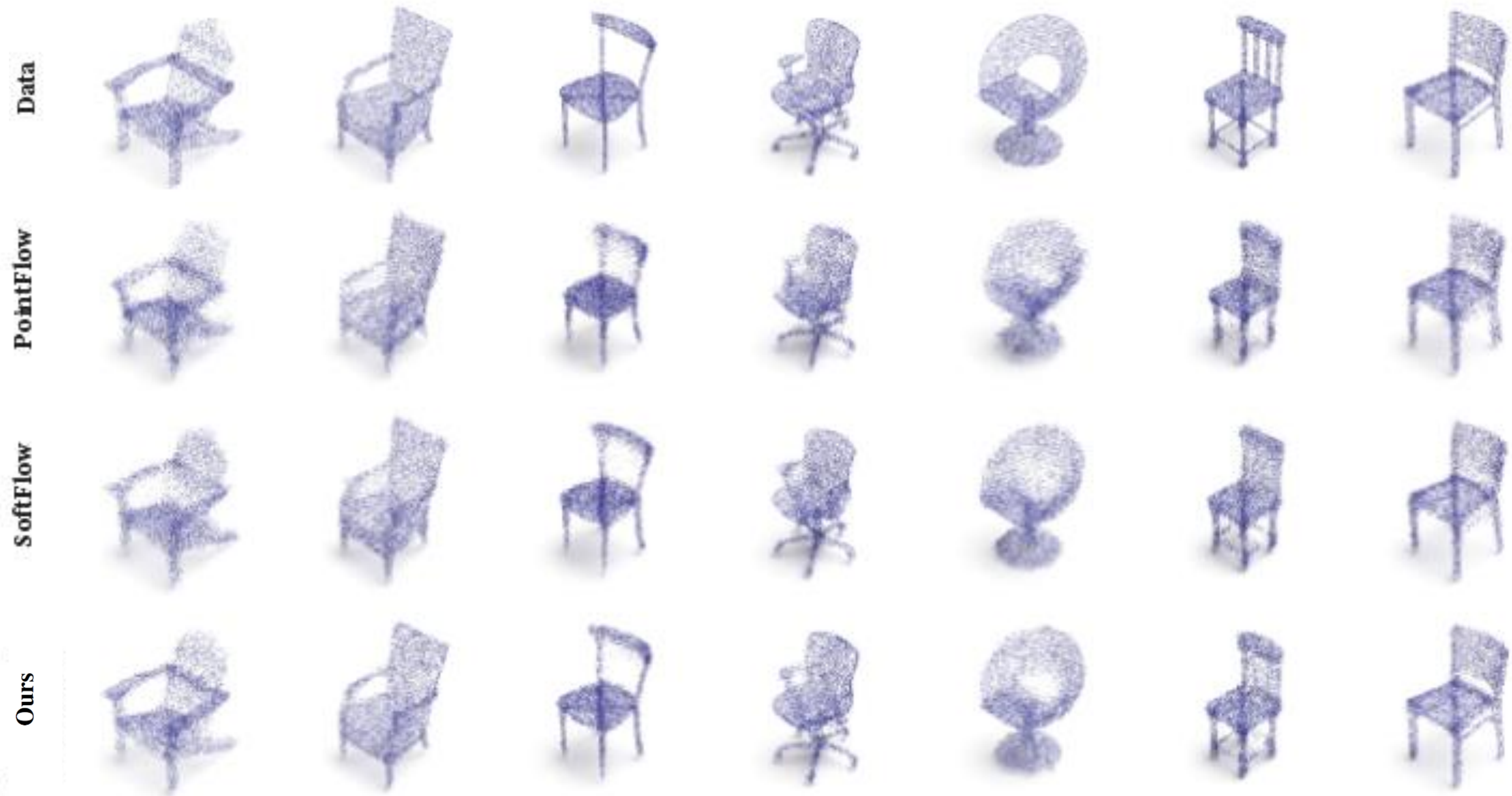
Additional Results (Generation)

Additional Results (Reconstruction)

Category	Number of Charts	CD	EMD
Airplane	1	1.18	2.64
	4	1.13	2.40
	8	1.13	2.32
	12	1.14	2.30
	16	1.09	2.26
	20	1.08	2.25
	24	**1.07**	**2.23**
	28	1.12	2.27
	32	1.14	2.25
Chair	1	11.76	6.92
	4	10.89	5.82
	8	10.43	5.47
	12	9.40	4.90
	16	9.04	4.71
	20	**8.76**	4.64
	24	8.78	**4.62**
	28	9.47	**4.62**
	32	10.31	4.79
Car	1	6.95	5.47
	4	6.78	4.58
	8	6.66	4.39
	12	6.56	4.19
	16	6.34	4.12
	20	6.31	4.08
	24	**6.20**	**3.96**
	28	6.35	3.98
	32	6.27	**3.96**

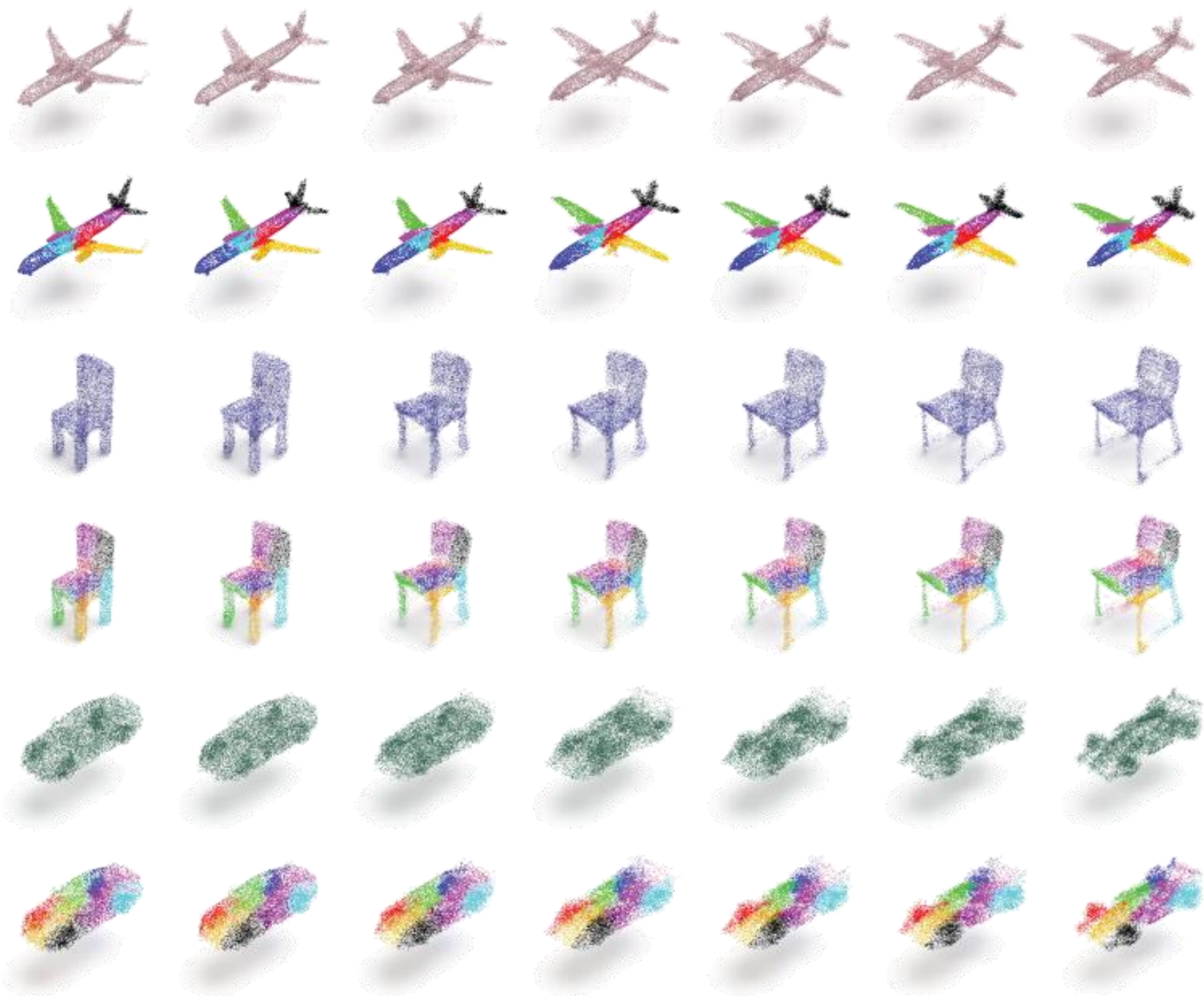
Additional Results (Reconstruction)

Additional Results (Reconstruction)

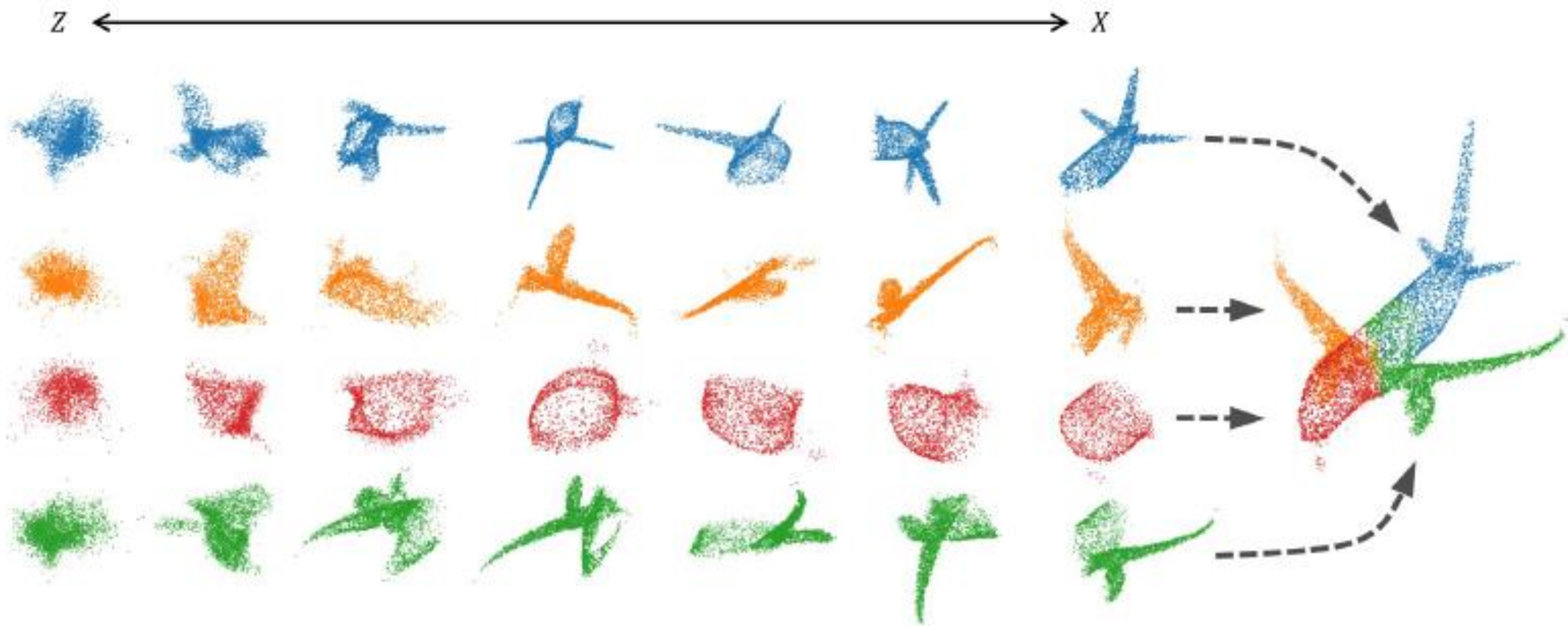


Additional Results (Reconstruction)

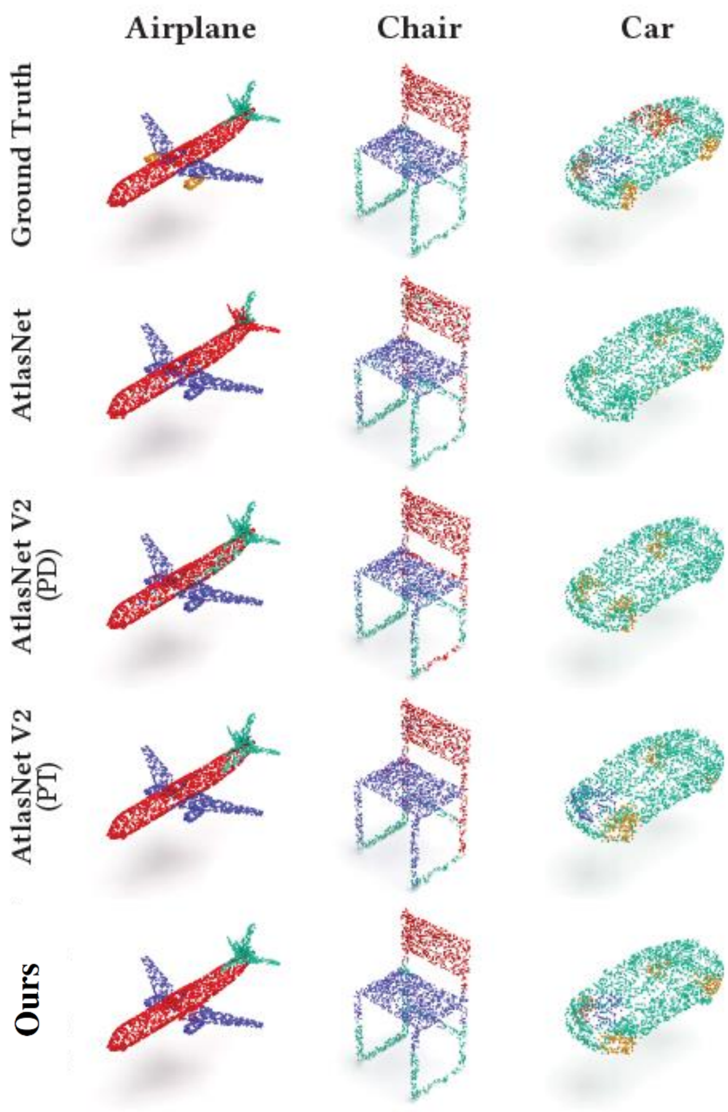
Linear Interpolation



Transformation From A Simple Dist.



Unsupervised part segmentation



Model	Airplane	Chair	Car
AtlasNet [11]	0.22 / 0.76	0.23 / 0.74	0.11 / 0.71
AtlasNet V2 (PD) [5]	0.25 / 0.79	0.24 / 0.75	0.13 / 0.72
AtlasNet V2 (PT) [5]	0.27 / **0.80**	0.24 / 0.74	0.17 / 0.73
Ours	**0.30 / 0.80**	**0.35 / 0.86**	**0.19 / 0.79**

Segmentation performances (NMI / purity)

Objective Function

提案手法のObjective Function

$$\mathcal{L} = \mathcal{L}(F, C, E, G, K; \chi, \mu, \lambda) + \mathcal{L}_{CP}(K; X)$$

生成モデルとしてのObjective Function

$$\mathcal{L}(F, C, E, G, K; \chi, \mu, \lambda) = \sum_{X \in \chi} [\mathcal{L}_{ELBO} + \mathcal{L}_{MI}]$$

negative ELBO $\mathcal{L}_{ELBO}$

$$\begin{aligned} \log p(X) &\geq \mathbb{E}_{q_E(s_X|X) q_C(Y|X, s_X)} \left[\log \frac{p_F(X|Y, s_X) p(Y) p_G(s_X)}{q_C(Y|X, s_X) q_E(s_X|X)} \right] \\ &= \mathbb{E}_{q_E(s_X|X)} \left[\sum_j \left\{ \mathbb{E}_{q_C(y_j|x_j, s_X)} \left[\log p_F(x_j|y_j, s_X) \right] \right. \right. \\ &\quad \left. \left. + H[q_C(y_j|x_j, s_X)] \right\} \right] - D_{KL}(q_E(s_X|X) \| p_G(s_X)) \\ &\triangleq -\mathcal{L}_{ELBO}(F, C, E, G; X). \end{aligned}$$

正則化項

$$\begin{aligned} &\sum_j \left\{ -\mu H \left[\frac{1}{|X|} \sum_{\tilde{x} \in X} q_C(y_j|\tilde{x}) \right] \right. \\ &\left. + \lambda H[q_C(y_j|x_j)] \right\} \\ &\triangleq \mathcal{L}_{MI}(C; X, \mu, \lambda). \end{aligned}$$

Objective Function

提案手法のObjective Function

$$\mathcal{L} = \mathcal{L}(F, C, E, G, K; \chi, \mu, \lambda) + \mathcal{L}_{CP}(K; X)$$

生成のためにChartの割合を学習する

$$\mathcal{L}_{CP}(K; X) = D_{KL}(p_K(y|s_X) \| q_C(y|X))$$

Gumbel-Softmax

ノイズgを次のように生成する

$$u \sim \mathrm{Uniform}(0,1)$$
$$g = -\log(-\log(u))$$

温度パラメータをτとしlabel y の各要素の値を

$$y_i = \frac{\exp((\log(\pi_i) + g_i)/\tau)}{\sum_{j=1}^{k} \exp((\log(\pi_i) + g_i)/\tau)}$$

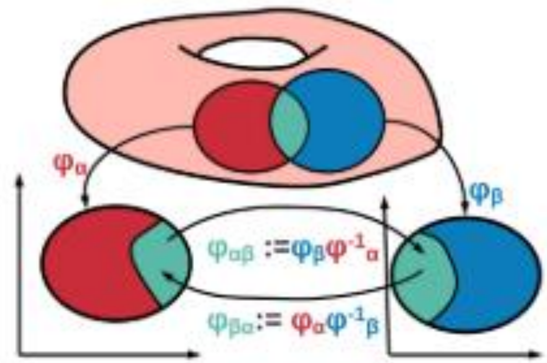
つまりNN f_ϕを用いて

$$\boldsymbol{y} = \mathrm{softmax}((\log(f_\phi(\boldsymbol{x})) + \boldsymbol{g})/\tau)$$

Background | 多様体,Chart

多様体：局所的にはユークリッド空間と見なせるような図形や空間

$\mathcal{M}$：位相空間，$\mathcal{M}_\alpha$：$\mathcal{M}$ の開集合
$\mathcal{M}'_\alpha$:ユークリッド空間での開集合



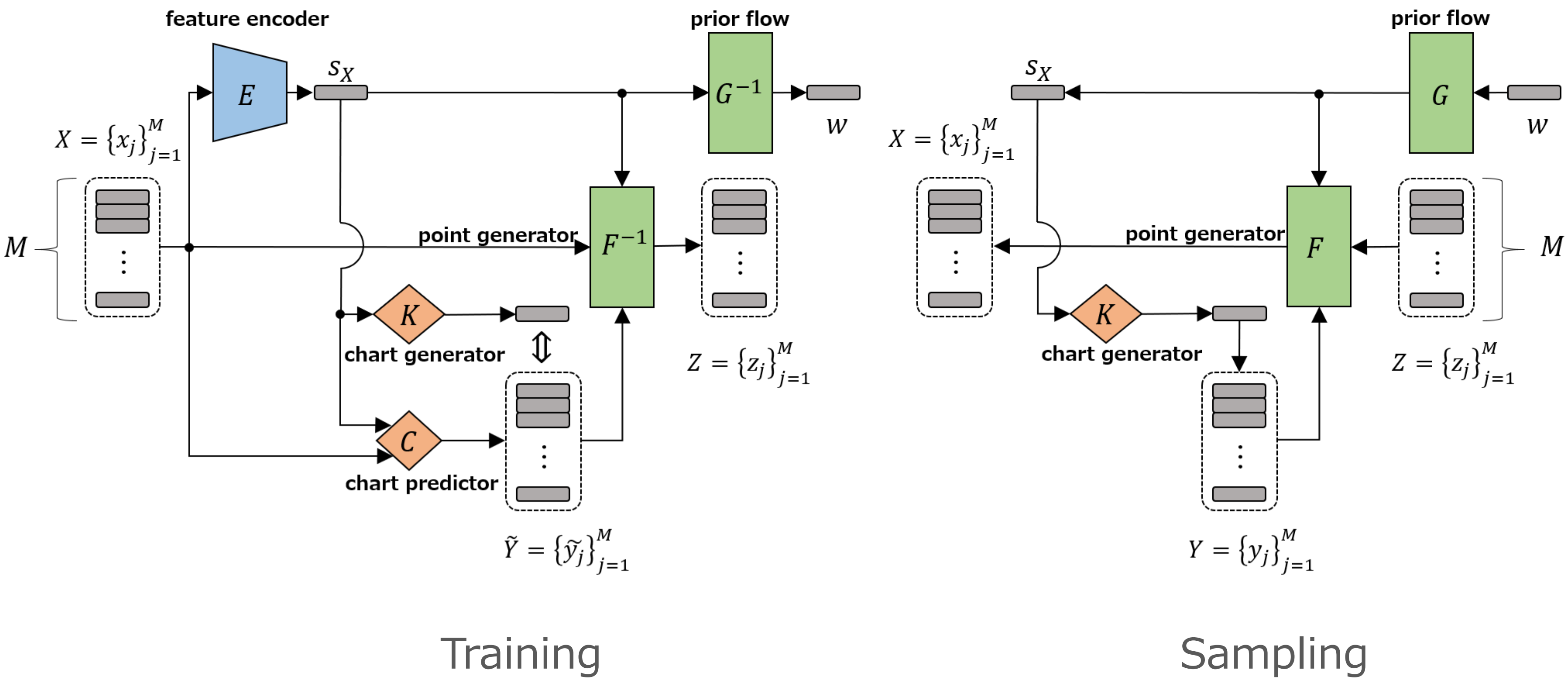
$$\varphi_\alpha : \mathcal{M}_\alpha \to \mathcal{M}'_\alpha$$

この同相写像をChart(($\mathcal{M}_\alpha, \varphi_\alpha$)とも表す)という

$S = \{(\mathcal{U}_\alpha, \varphi_\alpha) | \alpha \in \mathrm{A}\}$が$\mathcal{M}$全体を覆っている
- ➢Atlas

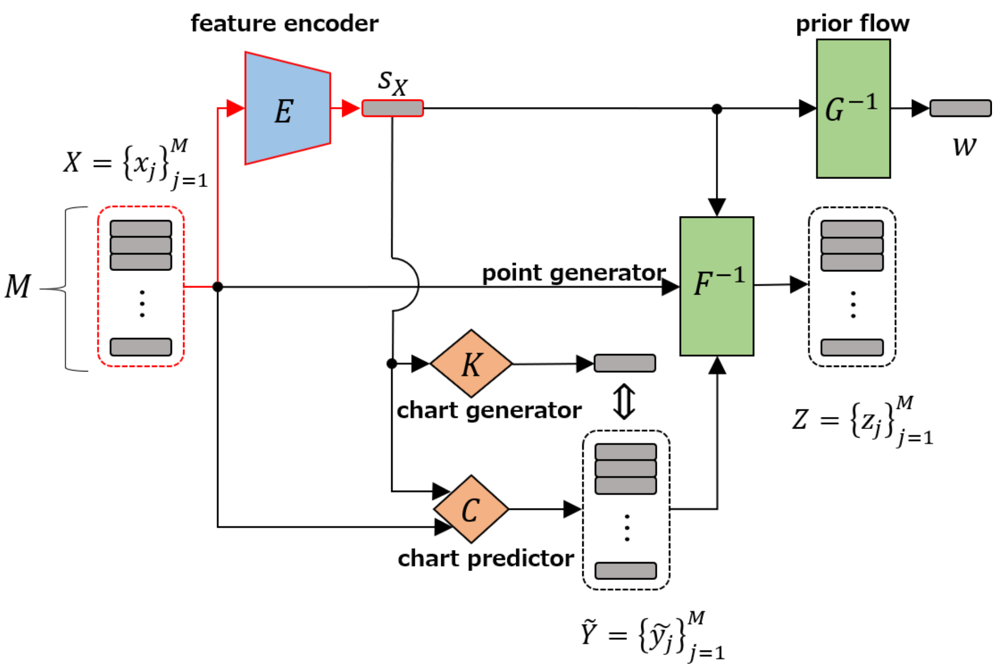
提案手法 | 3D Point Cloudsへの拡張

ネットワークアーキテクチャ



提案手法 | 3D Point Cloudsへの拡張 (学習時)

ネットワークアーキテクチャ



Training

feature encoder E

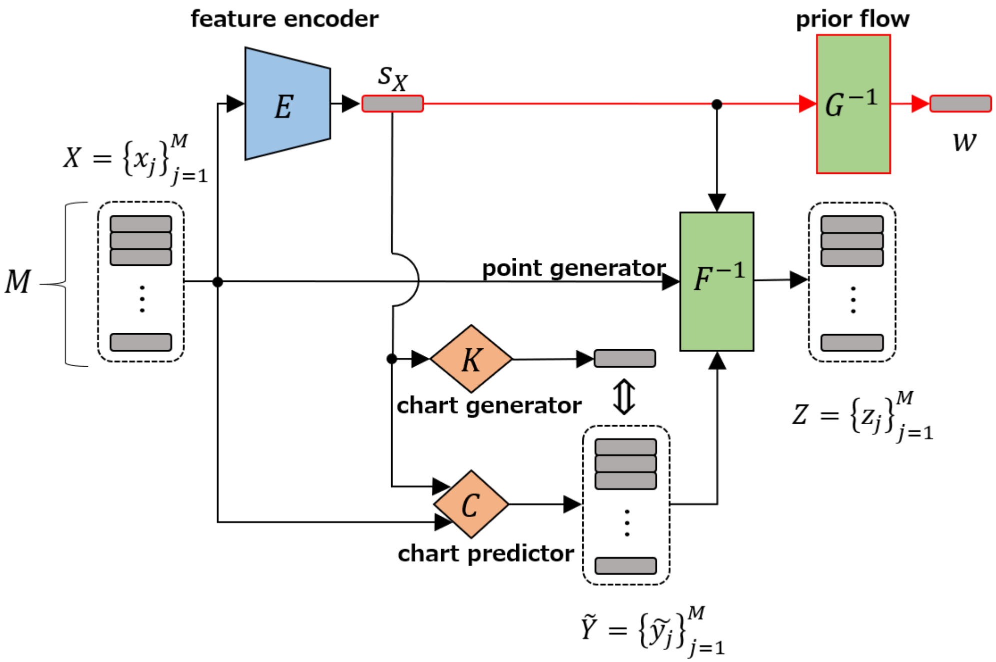
- 特徴ベクトルをs_X抽出
- Point Cloudの順不変な性質に対応する処理

prior flow G

- 特徴ベクトルs_Xを潜在変数wに写像
- 点群の分布を学習

提案手法 | 3D Point Cloudsへの拡張 (学習時)

ネットワークアーキテクチャ



Training

feature encoder E

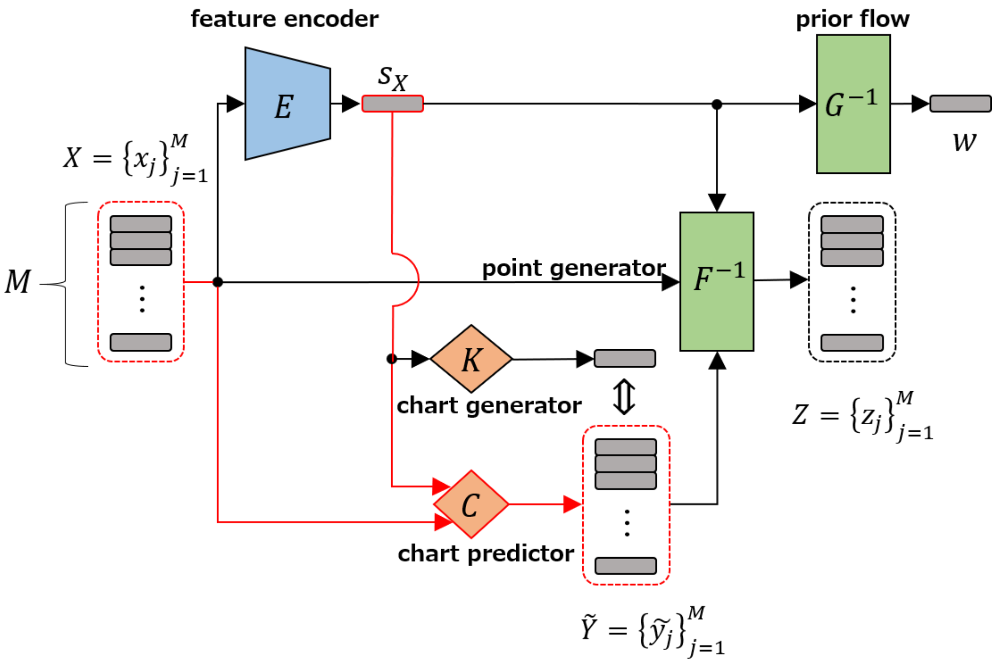
- 特徴ベクトルをs_X抽出
- Point Cloudの順不変な性質に対応する処理

prior flow G

- 特徴ベクトルs_Xを潜在変数wに写像
- 点群の分布を学習

提案手法 | 3D Point Cloudsへの拡張 (学習時)

ネットワークアーキテクチャ



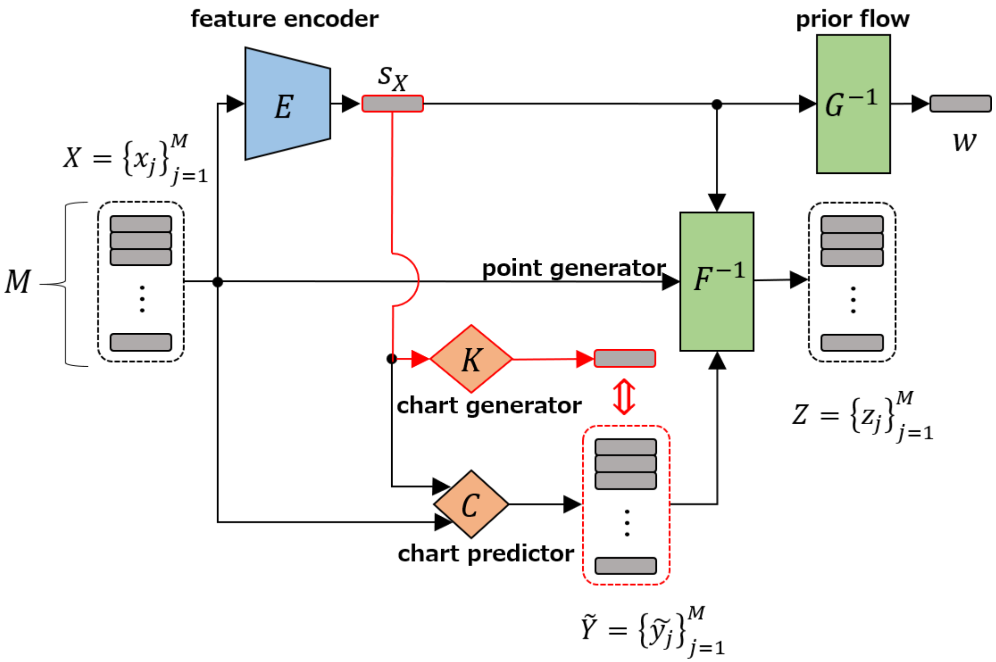
Training

chart predictor C
- 各点を入力とし，属するチャートに対応するラベルy_jを推論
- 特徴ベクトルs_Xにより条件付け

chart generator K
- ラベルyの事後分布$q_C(y|X)$を推定

提案手法 | 3D Point Cloudsへの拡張 (学習時)

ネットワークアーキテクチャ



Training

chart predictor C

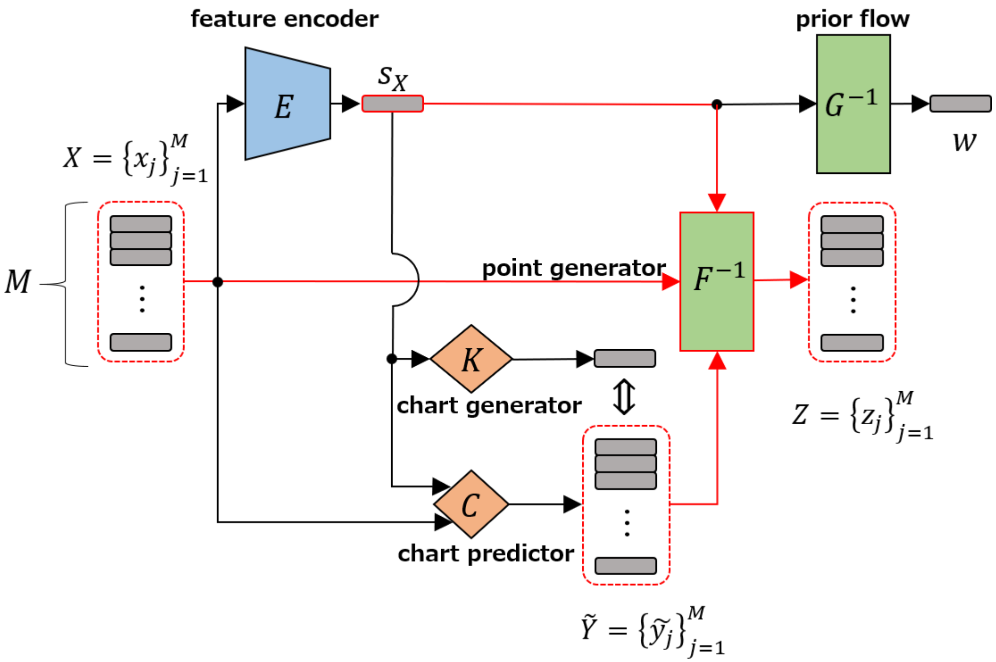
- 各点を入力とし，属するチャートに対応するラベルy_jを推論
- 特徴ベクトルs_Xにより条件付け

chart generator K

- ラベルyの事後分布$q_C(y|X)$を推定

提案手法 | 3D Point Cloudsへの拡張 (学習時)

ネットワークアーキテクチャ



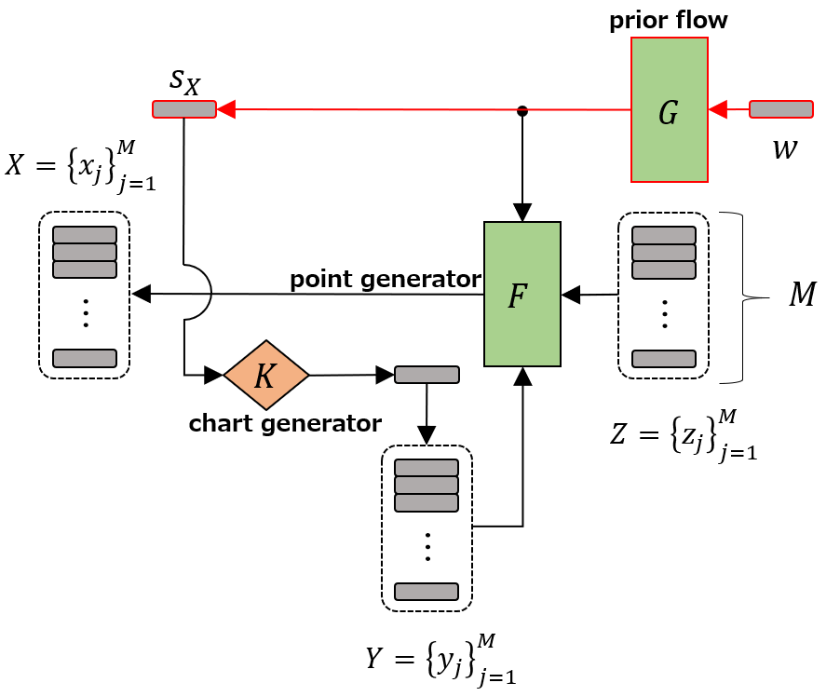
Training

point generator F

- 各点x_jを入力とし，潜在変数z_jに写像
- ラベルy_jと特徴ベクトルs_Xにより条件付け

提案手法 | 3D Point Cloudsへの拡張 (生成時)

ネットワークアーキテクチャ



Sampling

prior flow G

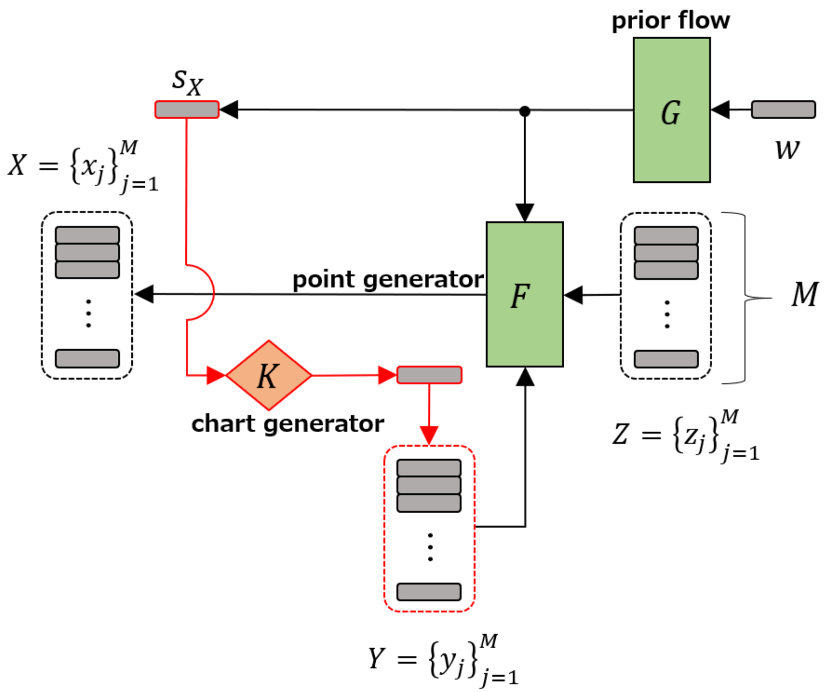
- 潜在変数wを事前分布$p(w)$からサンプリング
- 潜在変数wから特徴ベクトルS_Xを取得

chart generator K

- 特徴ベクトルS_Xを入力としてラベルyの事後分布$p_K(y|S_X)$を取得
- one-hotベクトルとしてラベルy_jを取得

提案手法 | 3D Point Cloudsへの拡張 (生成時)

ネットワークアーキテクチャ



Sampling

prior flow G

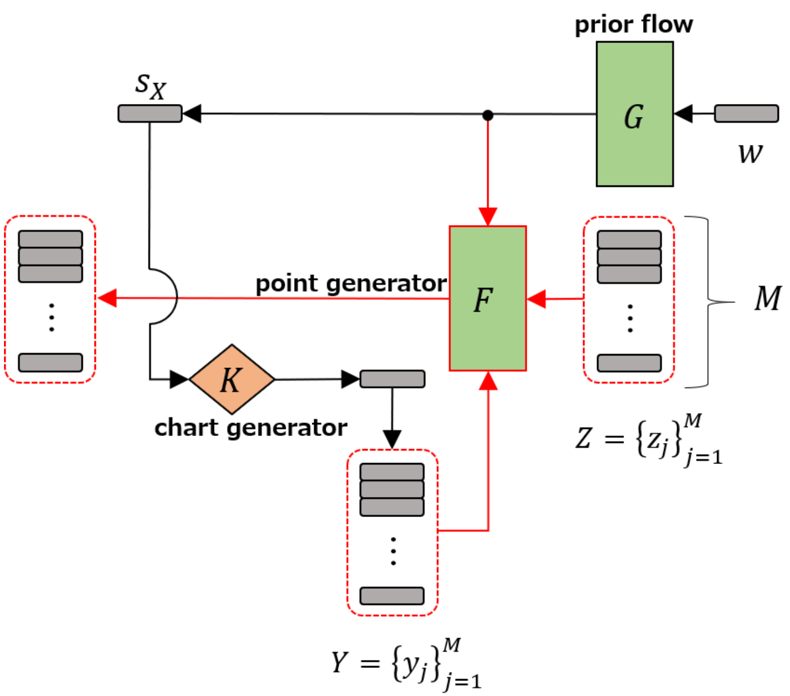
- 潜在変数wを事前分布$p(w)$から サンプリング
- 潜在変数wから特徴ベクトルS_Xを 取得

chart generator K

- 特徴ベクトルS_Xを入力として ラベルyの事後分布$p_K(y|S_X)$を取得
- one-hotベクトルとしてラベルy_j を取得

提案手法 | 3D Point Cloudsへの拡張 (生成時)

ネットワークアーキテクチャ



Sampling

point generator F

- 潜在変数z_jを事前分布$p(z)$からサンプリング
- 潜在変数z_j，特徴ベクトルs_Xラベルy_jより点x_jを生成