

トポロジーを考慮した3次元点群深層生成モデル

木村 匠^{†1,a)} 松原 崇^{†2,b)} 上原 邦昭^{†3,c)}

概要: 3次元データの表現としての点群は、deep learningにより分類やセグメンテーション、そして最近では生成に適応されてきた。従来の点群のための deep generative model は、球状の潜在変数から単一連続写像による変動をモデル化するように学習されており、点群のトポロジカルな構造を考慮することができていない。そのために、連続写像の前後によって変化する穴の数や、交点を表現することができなかった。そこで本論文では、複数の潜在ラベルを持つ flow-based deep generative model を提案する。潜在ラベルにより各連続写像は、多様体におけるチャートのように機能し、アトラスとして点群を生成する。これにより、既存手法ではぼやけたり穴の生成に失敗する傾向がある一方、提案手法においては明確な境界を持つトポロジカルな構造を維持することが可能になった。実験結果により、提案手法は sampling-based な点群生成手法の中で、生成と再構成において最高の性能を達成することが示された。

1. はじめに

3次元点群表現はロボット工学や自動運転などに利用され、その認識や分析をすることは重要であり、さまざまな研究が行われている [14], [19], [20], [23], [25], [26], [27], [29]. 物体の表面の表現として機能する点群は、LiDAR や Kinect などを用いることにより容易に取得することが可能である。さらに、ボクセルよりも高い解像度で物体をキャプチャーすることが可能であり、メッシュよりも幾何学的処理が簡単である。深層生成モデルを学習することにより、点群の生成、再構成、超解像などさまざまなタスクに活用することが可能である [1], [6], [8], [15], [22], [24], [28]. 点群のための深層生成モデルの多くは flow-based generative model [3], [5], [10] や、generative adversarial networks (GANs) [4] を用いており、これらは潜在変数空間からデータ空間の物体への写像を構築するように学習される。

点群は多くの場合、薄く、中空であり、円形の多様体的な構造をしている。flow-based generative model では、図1の上段に示すように、ユークリッド空間と穴のあるような多様体間の全単射が存在しないために、多様体的な構造を表現することは困難である。ニューラルネットワークによ

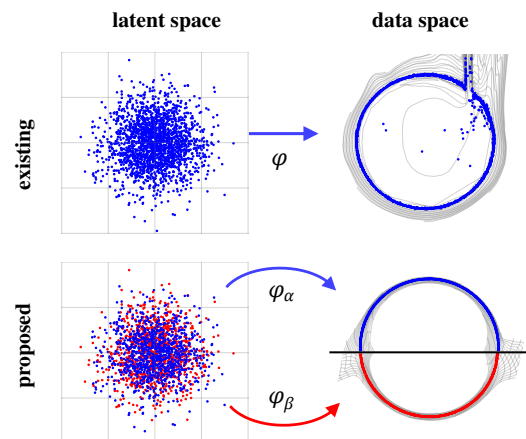


図1: 既存手法(上)と提案手法(下)の概念的な比較

てモデル化された写像 φ は、潜在空間内の2次元のボールを押しつぶし、弧を描くように引き伸ばし円を生成することで、1次元の円 S^1 上に存在する点群 X を表現する。しかし、これは外れ値や不連続性を引き起こす。この問題に対処するために、dynamic chart 法や多様体上に flow-based generative model を用いた手法が提案されてきた [17], [21]. しかし、これらの手法は多様体の幾何学的特徴が既知であり、固定されている場合にのみ有効で、さまざまな形状を持つ点群に対して常に有効であるというわけではない。さらに、点群はしばしば複数のサブパーツで構成されており、幾つかは切り離されている場合もあるため、これらを表現することは困難である。これは、ニューラルネットワークが連続写像を作る限り flow-based generative model にとどまらず GAN を用いた手法にも同じことが言える。

上記を踏まえて、我々は潜在ラベルを持つ3次元点群

^{†1} 現在, 神戸大学
Presently with Kobe University

^{†2} 現在, 大阪大学
Presently with Osaka University

^{†3} 現在, 大阪学院大学
Presently with Osaka Gakuin University

^{a)} kimura@ai.cs.kobe-u.ac.jp

^{b)} matsubara@sys.es.osaka-u.ac.jp

^{c)} kuniaki.uehara@ogu.ac.jp

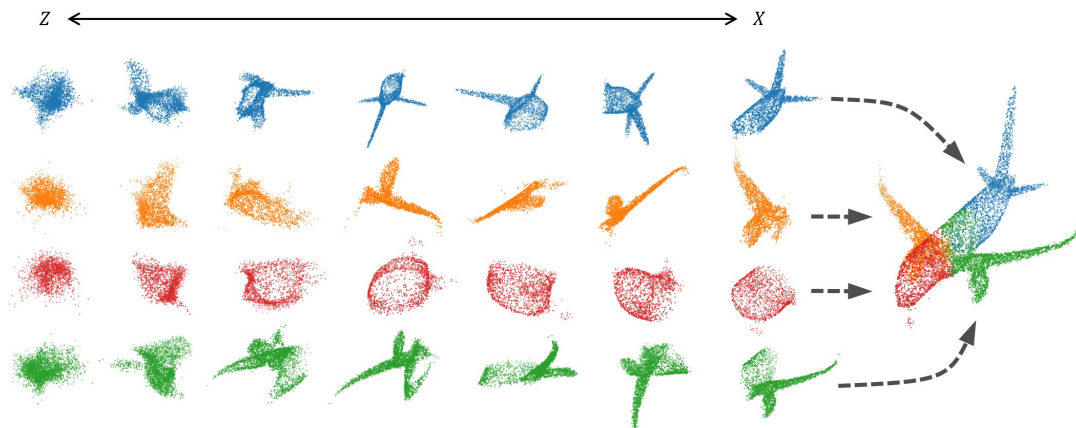


図 2: 提案手法における潜在変数空間での単純な分布 (左) から, データ空間でのオブジェクトのサブパーツ (右) への変換. 各段はチャートに対応

のための flow-based generative model を提案する. ラベルによって条件付けられた各連続写像は, 多様体におけるチャートのように, 与えられた点群の連続したサブセットに割り当てられる. そして, チャートの集合は点群全体を覆うアトラスを形成する. 図 1 を例にとると, 2つの写像 $\varphi_\alpha, \varphi_\beta$ は 2つの円弧を別々に生成し, データ空間内でそれらを連結することで連続した中空の円を生成する. より複雑な物体に関しては, 例えば, 各チャートは図 2 に示すように飛行機の機体, 右翼, 機首, 左翼のように意味的なサブパーツに割り当てられる.

本論文では点群の合成データセットと ShapeNet dataset [2] を用いて提案手法を評価した. 実験結果により, 既存手法においては生成された点群がぼやけたり, 穴の生成に失敗する傾向があったのに対し, 提案手法においてはトポロジカルな構造を詳細に保持していることが示された. 数値的な結果により, 提案手法はサンプリングベースの点群生成モデルである r-GAN [1], l-GAN [1], PC-GAN [15], PointFlow [28], SoftFlow [8] の中で最高の性能を発揮することが示された.

2. 関連研究

尤度ベースの点群生成: Yang ら [28] は順不変な処理を行う encoder と, flow-based generative model を組み合わせた PointFlow を提案した. flow-based generative model は全単射を形成し, 変数変換により尤度を求めるニューラルネットワークであり, 学習にヒューリスティックな距離関数を必要としない [3], [5], [10]. さらに, PointFlow は任意の数の点を入力として扱い, 生成することが可能である. flow-based generative model を使用し点群を生成する際は, 異なるトポロジーの多様体間の全単射が存在しないために, 曲面のような薄い構造をモデル化するには不安定になる傾向がある. Kim ら [8] はこの問題に対処するために学習時に点に摂動を与える SoftFlow を提案した.

SoftFlow はトポロジーの重要性を強調している. しか

し, 穴や, 交差, 切断など一般的なトポロジカルな構造には適応できていない. 我々はこの問題に取り組んでいる.

尤度を用いない点群生成: 点群生成のためのもう一つの手法として GAN を用いた手法がある. r-GAN は点群の全点を一度に生成し [1], l-GAN は事前に学習済みの autoencoder から抽出された特徴ベクトルに GAN を適用した [1]. 上記の GAN を用いた手法は任意の数の点を生成することは不可能である. PC-GAN [15] はこれを可能にした.

他の GAN を用いた手法は再帰的な超解像とみなすことが可能である. 各モデルは初めにスパースな点群を生成し, 次に既存の点を補間する形で点数を増やす. これを繰り返すことで点群の解像度を増加させる [6], [22], [24]. Valsesia ら [24] は互いに近い点は類似した特徴ベクトルを持つことを見つけた. また, Shu ら [22] は初めに生成された各点は, 点群の意味的な部分に関連している可能性を見つけた. これらの結果は意味的なサブパーツの重要性を示唆している. しかし, これらの研究において明示的には取り扱われていない.

深層混合生成モデル: 複数のカテゴリーのサンプルをモデル化するために, 深層生成モデルは conditional VAEs [11] や conditional GANs [18], conditional flow [3], [10] など, 混合分布に拡張されてきた. モデルに与えられるクラスラベルは, 既知, もしくは教師なしでデータから推定される.

提案手法もまた, サンプルのラベルを各点が属するチャートとしてデータから推定を行う. 上記のモデルとの主な違いは, 混合分布を仮定していないことである.

3. 背景

本節では既存手法の問題点を明らかにするために, 3次元の点群に適応する前に 2次元での結果を示す. 図 3 に示すように, circle [5], 2sines [8], double-moon [3] の 3つの合成データセットを用意し, それぞれを多数の点 $\{x_j\}$ から構成された物体 X として表す. 上段はデータセットを表し, 2段目と 3段目はバックボーンに Glow [10] を持

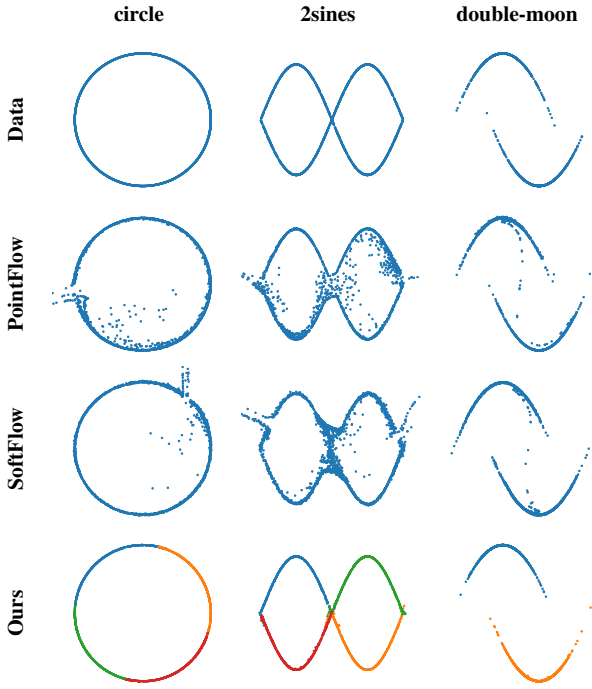


図3: PointFlow [28], SoftFlow [8], 提案手法による点群の生成. 各色は点が属するチャートを表す

つ PointFlow [28] と SoftFlow [8] の生成結果を表す. 生成された circle と 2sines には不連続性とぼやけた交点が見られ, double-moon においては紐状のアーティファクトが生じていることがわかる.

Flow-based generative model は可逆変換で構成されており, 潜在空間とデータ空間の全単射を与える. その際に, 潜在空間がユークリッド空間である限り, 点群 X は穴を持たない物体として生成される. PointFlow と SoftFlow は2次元の潜在分布を押しつぶし, 引き延ばし, 弧を描くように変形し, 薄い構造を表現しようとする. しかし, 穴や交点, 非連続な部分に対しては生成を失敗している. 一般的にニューラルネットワークは連続写像を形成するため, これはGANを用いた手法に対しても同じことが言える. これらの結果が我々の研究の動機となっている.

4. チャートを用いた flow-based generative model

ネットワーク構造: 簡単のために, まず単一の点群 X の生成モデルに焦点を当てる. point generator F を, ラベル y によって条件づけられた点 $x \in X$ のための flow-based generative model とする. point generator F はチャートとみなすことが可能であり, n 個のチャートの集合が点群 X 全体をカバーするアトラスを形成する. このとき, 点 x の条件付き対数尤度は, 変数変換によって次のように与えられる.

$$\log p_F(x|y) = \log p(z) + \log \left| \det \frac{\partial F^{-1}(x;y)}{\partial x} \right|, \quad (1)$$

ここで z は潜在変数であり $z = F^{-1}(x;y)$, また, その事前分布 $p(z)$ は標準正規分布である. 対数周辺尤度 $\log p_F(x)$ は, すべてのラベルに対して総和をとることで $\log p_F(x) = \log \sum_y p_F(x|y)p(y)$ と求めることが可能である. しかし, 我々はニューラルネットワークで実装された変分推論モデル $q_C(y|x)$ (chart predictor C とする) を用いているため, negative evidence lower bound (ELBO) $\mathcal{L}_{ELBO}(F; C; x)$ として次のように求められる.

$$\begin{aligned} \log p_F(x) &= \mathbb{E}_{q_C(y|x)} \left[\log \frac{p_F(x;y) q_C(y|x)}{p_F(y|x) q_C(y|x)} \right] \\ &\geq \mathbb{E}_{q_C(y|x)} \left[\log \frac{p_F(x|y)p(y)}{q_C(y|x)} \right] \\ &= \mathbb{E}_{q_C(y|x)} \left[\log p(z) + \log \left| \det \frac{\partial F^{-1}(x;y)}{\partial x} \right| \right] \quad (2) \\ &\quad - H[q_C(y|x)|p(y)] + H[q_C(y|x)] \\ &\triangleq -\mathcal{L}_{ELBO}(F; C; x), \end{aligned}$$

ここで $H[q_C(y|x)|p(y)]$, $H[q_C(y|x)]$ はクロスエントロピーとエントロピーを表す. 我々はラベルの事前確率分布 $p(y)$ を一様分布であると仮定しているため, クロスエントロピー $H[q_C(y|x)|p(y)]$ は定数となり, これ以降は省略する.

目的関数: ラベル y は one-hot ベクトルとして表現され, このとき negative ELBO \mathcal{L} はすべてのラベルに対して加重平均をとることによって求められる. しかし, この方法はチャートの数に比例して計算時間が増加する. そこで, Gumbel-Softmax [7] を使用することで, この問題を回避した.

$$\begin{aligned} \tilde{y} &= \text{softmax}((\log \pi_C(x) + \mathbf{g})/\tau) \\ \mathbf{g} &\sim \text{Gumbel}(0, 1), \end{aligned} \quad (3)$$

ここで \mathbf{g} はベクトルであり, その要素は Gumbel distribution $\text{Gumbel}(0, 1)$ に従う. また, $\tau \in (0, \infty)$ は softmax 関数の温度を表す. 例えば, $\tau \rightarrow +0$ であれば, 出力 \tilde{y} は one-hot ベクトルに近づき, $\tau \rightarrow +\infty$ であれば, 出力 \tilde{y} は一様分布に従うベクトルに近づく. $\pi_C(x)$ はラベルの事後分布 $q_C(y|x)$ のベクトルを表し, $\pi_C(x)_j = q_C(y_j|x)$ と書ける. Gumbel-Softmax により, ラベル \tilde{y} を微分可能な形でモンテカルロサンプリングをすることが可能となり, negative ELBO は次のように近似される.

$$\begin{aligned} \mathcal{L}_{ELBO}(F; C; x) &\simeq -\log p(z) - \log \left| \det \frac{\partial F^{-1}(x;\tilde{y})}{\partial x} \right| - H[q_C(y|x)] \quad (4) \\ &\triangleq \tilde{\mathcal{L}}_{ELBO}(F; C; x), \end{aligned}$$

ここでベクトル \tilde{y} は式 (3) によって得られる. このとき, チャート数によらず計算コストが一定となることに注意されたい.

近似された negative ELBO $\tilde{\mathcal{L}}$ を最小化する場合, 同時にエントロピー $H[q_C(y|x)]$ は最大化される. これにより, 各点のラベルに属する確率が等しく, つまり, チャートが互

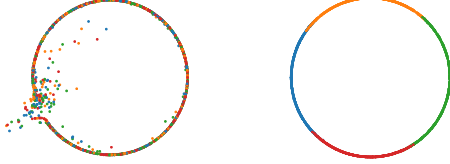


図4: 正則化項 \mathcal{L}_{MI} の影響. (左) 正則化項なし. (右) 正則化項あり. 各色はその点が属するチャートを表す

いにオーバーラップすることを引き起こす. そこで, 各チャートを多様体 (点群) の特定の連続した領域に割り当てるために, 相互情報量 $I(x; y)$ に基づく正則化項 $\mathcal{L}_{MI}(x, y)$ を導入する.

$$\begin{aligned} -I(y; x) &= -(H[q_C(y)] - H[q_C(y|x)]) \\ &= -H[\mathbb{E}_{p(\tilde{x})}[q_C(y|\tilde{x})]] + H[q_C(y|x)] \\ &\simeq -H\left[\frac{1}{|X|} \sum_{\tilde{x} \in X} q_C(y|\tilde{x})\right] + H[q_C(y|x)] \\ &\triangleq \mathcal{L}_{MI}(C; x). \end{aligned} \quad (5)$$

正則化項 \mathcal{L}_{MI} を最小化することで, negative ELBO $\tilde{\mathcal{L}}(F; C; X)$ においてエントロピー $H[q_C(y|x)]$ が最大化されることを相殺し, かつ, エントロピー $H[q_C(y)]$ が最大化されることとなる. これにより, 各サンプルは一つのチャートに属するようになり, 点群 X 全体で見た場合, すべてのチャートは一樣な確率で使用されることとなる.

独立同時分布の仮定により, 点群 X の目的関数は各点 x の和によって次のように表すことが可能である.

$$\mathcal{L}(F; C; X, \lambda) = \sum_{x \in X} \left[\tilde{\mathcal{L}}_{ELBO} + \lambda \mathcal{L}_{MI} \right], \quad (6)$$

ここで λ は正則化係数である.

合成データにおける実験: 提案手法のコンセプトを証明するために, 図3のように2次元の合成データセットを用いて実験を行った. point generator F のバックボーンとして Glow [10] を用いた. circle と 2sines のデータセットに対しては $n = 4$ チャート, double-moon には $n = 2$ チャートを用いた. 正則化係数は $\lambda = 1.05$, Gumbel-Softmax の温度は $\tau = 0.1$ と設定した. 学習は Adam optimizer [9] を用いて, 学習率を 10^{-4} , バッチサイズを 100 とし 10K iteration 行った.

同様の実験設定下で PointFlow [28] と SoftFlow [8] の学習を行った. ラベルを用いない場合, 提案手法は PointFlow と同じになることに注意されたい.

実験によって生成された点群を図3に示す. PointFlow と SoftFlow は節3で述べたように, アーティファクトを伴い, ぼやけた, 非連結な表現に失敗する点群を生成している. 対照的に, 提案手法においては不連続性のない円や, 激しくぼやけることのない交点, アーティファクトのない明確に分離された2つの円弧が生成されている. 各色はその点が属しているチャートを表している. circle と

2sines のデータセットに関しては, 4つのサブパーツが滑らかに接続され穴のある多様体を形成している. 2sines の交点はサブパーツの交点として形成されていることもわかる. double-moon データセットに関しては, 各チャートがそれぞれ1つの円弧に割り当てられ, そのため非連結な多様体を形成することが可能になっている. これらの結果は, チャートのコンセプトが同じ潜在変数分布 $p(z)$ であっても, 様々なトポロジカルな構造に対してうまく機能することを示している.

正則化項 \mathcal{L}_{MI} の影響を見るために, 正則化係数を $\lambda = 0$ とした結果を図4に示す. 正則化項 \mathcal{L}_{MI} を導入しない場合, 各ラベルはお互いにオーバーラップし, 点群全体に割り当てられ, さらに, 不連続性が見られた. これは, エントロピー $H[q_C(y|x)]$ が最大化されることにより, 事後確率分布 $q_C(y|x)$ が一樣になってしまうためである.

5. 提案手法

本節では, 節4で提案したモデルを3次元点群データに適用するために拡張する. 図5にその概念図を示す. 点群データセット \mathcal{X} は N 個のオブジェクト $\{X_1, X_2, \dots, X_N\}$ から構成されており, 各オブジェクト X_i は M_j 個の点 $\{x_1, x_2, \dots, x_{M_j}\}$ により表されるものとする.

ネットワーク構造: 提案手法では, feature encoder E と呼ばれる順不変なニューラルネットワークが, M 個の点からなる点群 X を入力として受け取る. そして, reparameterization trick [13] を用いて特徴ベクトル s_X の事後分布 $q_E(s_X|X)$ へと符号化する. ここで, 特徴ベクトル s_X は点群 X 全体の形状を表現したものであると考えられる. ガウス事前分布を用いた reparameterization trick は, 出力 s_X が入力 X を無視する posterior collapse が引き起こされることが知られている [12], [28]. そこで, 事前分布の表現力を上げるために prior flow G と呼ばれる flow-based generative model を用いて, 特徴ベクトル s_X を潜在変数 w に写像する. このとき, 特徴ベクトル s_X の学習可能な事前分布 $p_G(s_X)$ は次のように表される.

$$\log p_G(s_X) = \log p(w) + \int_{t_0}^{t_1} \log \left| \det \frac{\partial G^{-1}(s_X)}{\partial s_X} \right|. \quad (7)$$

ここで $w = G^{-1}(s_X)$ であり, 事前分布 $p(w)$ は標準正規分布である. これにより, prior flow G は点群の分布を学習する.

提案手法では, chart predictor $q_C(y|x, s_X)$ は点 $x \in X$ を入力として受け取り, 各点が属するチャートに対応するラベル y を推論する. このとき, chart predictor は特徴ベクトル s_X により条件付けられる. これは, 同じデータセットでも形状により異なるアトラスを持つことが考えられるためである. 例えば, 同じ座標の点であったとしても飛行機の幅によっては, エンジンの一部や機体を表している場

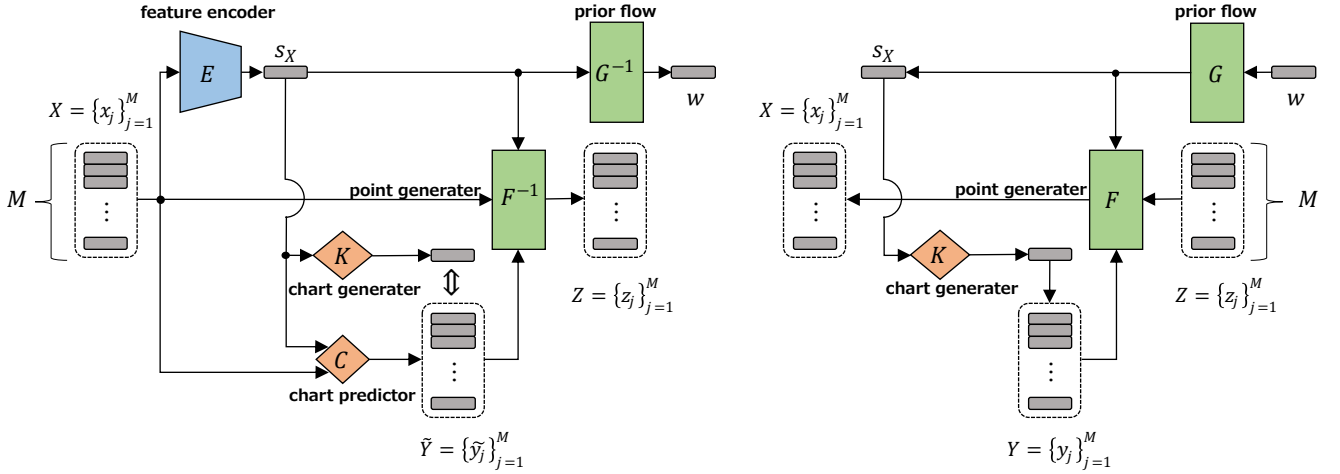


図 5: 学習時 (左) と生成時 (右) のアーキテクチャとデータフロー

合がある。

また, point generator F も同様に, 異なる形状を生成可能にするために特徴ベクトル s_X によって条件付けされる. このとき, 点 x の対数尤度は次のように表される.

$$\log p_F(x|y, s_X) = \log p(z) + \log \left| \det \frac{\partial F^{-1}(x; y, s_X)}{\partial x} \right|. \quad (8)$$

我々はラベル y の事前分布 $p(y)$ を, 節 4 と同様に一樣であると仮定している. しかし, サブパーツの大きさは点群 X に依存することが考えられる. 生成時のために, さらに特徴ベクトル s_X を入力としてラベルの事後確率 $p_K(y|s_X)$ を出力するニューラルネットワーク chart generator K を導入する. これによりラベル y の事後分布 $q_C(y|X, s_X) = \mathbb{E}_{x \in X} \sum_j q_C(y|x, s_X)$ が学習される.

目的関数: ラベルの集合を Y とする. これは, 与えられた点群 X の各点 x にそれぞれ対応している. 独立同時分布の仮定により, $p(Y) = \prod_j p(y_j)$, $q_C(Y|s_X) = \prod_j q_C(y_j|s_X)$, $p_F(X|Y, s_X) = \prod_j p_F(x_j|y_j, s_X)$ となり, negative ELBO \mathcal{L}_{ELBO} は次のように表される.

$$\begin{aligned} \log p(X) &\geq \mathbb{E}_{q_E(s_X|X)q_C(Y|X, s_X)} \left[\log \frac{p_F(X|Y, s_X)p(Y)p_G(s_X)}{q_C(Y|X, s_X)q_E(s_X|X)} \right] \\ &= \mathbb{E}_{q_E(s_X|X)} \left[\sum_j \left\{ \mathbb{E}_{q_C(y_j|x_j, s_X)} [\log p_F(x_j|y_j, s_X)] \right. \right. \\ &\quad \left. \left. - H(q_C(y_j|x_j, s_X)) \right\} \right] - H[q_E(s_X|X)|p_G(s_X)] \\ &\triangleq -\mathcal{L}_{ELBO}(F, C, E, G; X). \end{aligned} \quad (9)$$

実際は, ラベル y_j に対する期待値 $\mathbb{E}_{q_C(y_j|x_j, s_X)}$ は Gumbel-Softmax [7] によって近似され, 特徴ベクトル s_X に対する期待値 $\mathbb{E}_{q_E(s_X|X)}$ はモンテカルロサンプリング [13] により近似される. 近似された negative ELBO を $\tilde{\mathcal{L}}_{ELBO}(F, C, E, G; X)$ と表記する.

式 5 の正則化項の第 1 項は, 各オブジェクトにおいてすべてのチャートを等しく使用することを強制する. しか

し, 実際には各チャートはそれぞれの大きさを持っていることが考えられる. そこで, 我々はより柔軟な調整のために次のように係数 μ と λ を導入した.

$$\sum_j \left\{ -\mu H \left[\frac{1}{|X|} \sum_{\tilde{x} \in X} q_C(y_j|\tilde{x}) \right] + \lambda H[q_C(y_j|x_j)] \right\} \triangleq \mathcal{L}_{MI}(C; X, \mu, \lambda). \quad (10)$$

以上より目的関数は次のように表される.

$$\mathcal{L}(F, C, E, G, K; X, \mu, \lambda) = \sum_{X \in \mathcal{X}} \left[\tilde{\mathcal{L}}_{ELBO} + \mathcal{L}_{MI} \right]. \quad (11)$$

また, chart generator K は次の目的関数を最小化することで, ラベルの事後分布 $q_C(y|X)$ を推定するように別に学習される.

$$\mathcal{L}_{CP}(K; X) = D_{KL}(p_K(y|s_X) \parallel q_C(y|X)). \quad (12)$$

使用方法とタスク: 生成タスクについては, 図 5 の右図に従うことで物体が生成される. 形式的には, $p(X) = \int_{s_X} p_G(s_X) \prod_j \int_{y_j} p_F(x_j|y_j, s_X) p_K(y_j|s_X)$ と書くことができる.

再構成, 超解像のタスクにおいては prior flow G から特徴ベクトル s_X を取得する代わりに, 与えられた点群 X を feature encoder E に入力することで特徴ベクトル s_X を取得する.

提案手法では, チャート数によらず計算コストが一定であり, chart predictor は学習時のみ使用される. また, 他の構成要素 (E, F, C) は点群の点の数に比例した計算コストを必要とするのに対し, chart generator K は点群の数 (オブジェクトの数) に比例するために無視することが可能である. 以上のことより, 提案手法の生成時の計算コストは, 比較手法である PointFlow [28] と SoftFlow [8] とほぼ等しいと言える.

6. 実験結果

6.1 実験設定

データセットとして ShapeNet dataset [2] を用いた. 本実

験では、この中から Yang ら [28] に従い、airplane, chair, car の3つのカテゴリにおいて実験を行った。

学習設定は SoftFlow の公開コード [8] に従った。具体的には、Adam optimizer [9] を用い、バッチサイズ 128, 初期学習率 2.0×10^{-3} , $\beta_1 = 0.9$, $\beta_2 = 0.999$ とし、各カテゴリ 15K epoch 学習を行った。また、学習率は 5K epoch ごとに 1/4 に減衰させ、各オブジェクト X からはランダムに $M = 2,048$ 点取得した。再構成タスクにおいても、物体から 2,048 点取得し評価を行った。

Gumbel-Softmax の温度の設定として $\tau = 0.1$ とし、正則化項 \mathcal{L}_{MI} のために $\mu = 0.05$, $\lambda = 1.0$ と設定し、さらに、チャート数 n は {4, 8, 12, 16, 20, 24} の中から選択した。また、ネットワークのサイズは PointFlow [28] と SoftFlow [8] に従った。

6.2 評価指標

点群 X_1 と X_2 間の距離を測定するために、順不変な評価指標である Chamfer distance (CD) と Earth mover's distance (EMD) が一般的に使用されており [1], [8], [15], [28], 次のように表される。

$$CD(X_1, X_2) = \sum_{x \in X_1} \min_{\xi \in X_2} \|x - \xi\|_2 + \sum_{x \in X_2} \min_{\xi \in X_1} \|x - \xi\|_2, \quad (13)$$

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

ここで、点群 X_1 と X_2 は同じ点数で構成されている。 ϕ は点群 X_1 から X_2 への全単射を表し、 $\|\cdot\|_2$ は \mathbb{R}^3 上でのユークリッド距離を表す。

点群の集合 \mathcal{X}_1 と \mathcal{X}_2 間の類似度を評価するために、既存研究では Jensen-Shannon divergence (JSD), minimum matching distance (MMD), coverage (COV), 1-nearest neighbor accuracy (1-NNA) [1], [8], [15], [16], [28] が提案されている。しかし、JSD, MMD, COV, は悪いモデルに対して良いスコアを与える可能性があることが指摘されている [28]。例えば、JSD は個々の形状を考慮せずに、平均的な形状を出力するモデルに良いスコアを与える [28]。したがって、我々は 1-NNA を用いてモデルを評価した。

1-NNA は 2 標本検定において 2 つの分布が同一であるかどうかを評価しており、次のように表される。

$$1-NNA(\mathcal{X}_1, \mathcal{X}_2) = \frac{\sum_{X_1 \in \mathcal{X}_1} \mathbb{1}[N_{X_1} \in \mathcal{X}_1] + \sum_{X_2 \in \mathcal{X}_2} \mathbb{1}[N_{X_2} \in \mathcal{X}_2]}{|\mathcal{X}_1| + |\mathcal{X}_2|}, \quad (14)$$

ここで、 \mathcal{X}_1 と \mathcal{X}_2 は同数の点群で構成されており、 N_{X_\bullet} は $\mathcal{X}_1 \cup \mathcal{X}_2 - \{X_\bullet\}$ に対する X_\bullet の最近傍を与える。 $\mathbb{1}[\cdot]$ は指示関数を示す。簡単に言えば、1-NNA は与えられた点群 X を CD または EMD の観点から最も近いサンプル N_X に従って \mathcal{X}_1 もしくは \mathcal{X}_2 に分類する。1-NNA の精度が 50% に近ければ近い程 \mathcal{X}_1 と \mathcal{X}_2 は類似している。

表 1: 1-NNA による生成結果の評価。50%に近い程良い

Category	Model	CD	EMD
Airplane	r-GAN [1]	93.58	99.51
	l-GAN (CD) [1]	86.30	97.28
	l-GAN (EMD) [1]	87.65	85.68
	PC-GAN [15]	94.35	92.32
	PointFlow [28]	75.68	75.06
	SoftFlow [8]	70.92	69.44
	Ours	69.39	65.62
Chair	r-GAN [1]	71.75	99.47
	l-GAN (CD) [1]	64.43	85.27
	l-GAN (EMD) [1]	64.73	65.56
	PC-GAN [15]	76.03	78.37
	PointFlow [28]	60.88	59.89
	SoftFlow [8]	59.95	63.51
	Ours	57.89	58.31
Car	r-GAN [1]	97.87	99.86
	l-GAN (CD) [1]	63.07	88.07
	l-GAN (EMD) [1]	69.74	68.32
	PC-GAN [15]	92.19	90.87
	PointFlow [28]	60.65	62.36
	SoftFlow [8]	62.63	64.71
	Ours	58.13	58.80

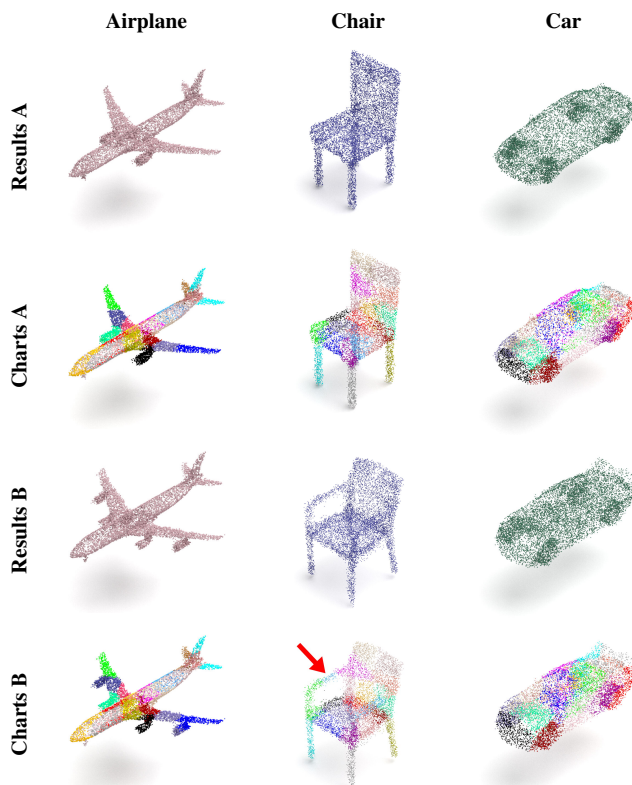


図 6: 提案手法による生成例

6.3 生成タスク

生成タスクでは、提案手法とサンプリングベースな生成手法である、r-GAN [1], l-GAN [1], PC-GAN [15], PointFlow [28], SoftFlow [8] を比較した。比較手法の結果は [28]

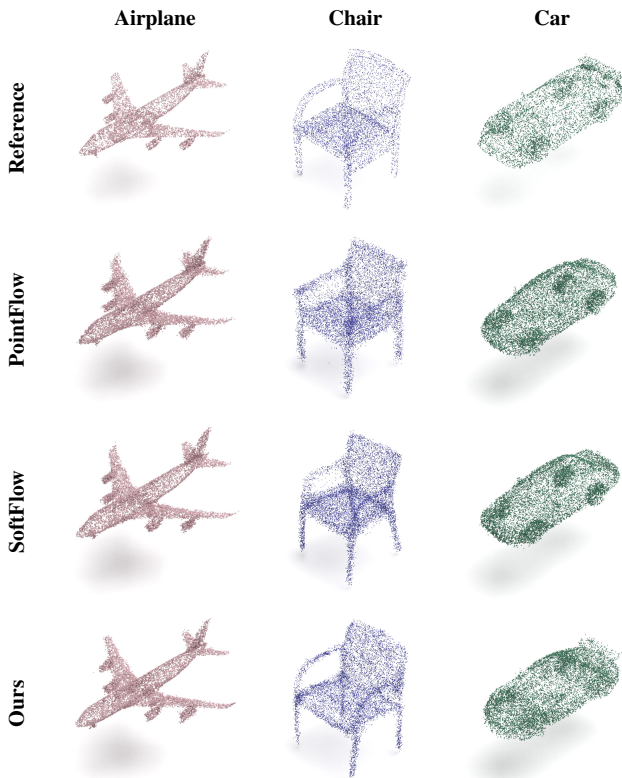


図 7: データセットから取得した参照に最も近い生成例

と [8] に基づいている。

実験結果を表 1 に示す。生成結果にはばらつきがあるために、我々の実験結果は Kim ら [8] と同様に 16 回の平均を取ったものを記載している。実験結果より、すべてのカテゴリにおいて提案手法が他の手法の性能を上回っていることがわかる。提案手法における結果は、airplane, chair, car 各カテゴリにおいてそれぞれ 24, 20, 16 のチャートを用いたものを記載している。しかし、提案手法は 12-24 チャートにおいて性能は一貫していた。

図 6 に提案手法によって生成されたオブジェクト (10,000 点) を示す。飛行機の水平尾翼、椅子の足、車の車輪のようなオブジェクトの突出したサブパーツはそれぞれ独自のチャートで表現されていることがわかる。また、同じサブパーツはオブジェクト間において同じチャートで表現されており、上段の椅子において下段の椅子のアームレストに割り当てられたチャートが使用されていないこともわかる (赤矢印参照)。これにより、提案手法は必要な場合にチャートを割り当てることで、様々なトポロジーを表現できることがわかる。

比較のために、図 7 に示すように評価サブセットから参照データを取得し、各モデルにより生成されたサンプルの中から、EMD の観点により最も近いサンプルを取得した。提案手法は他のモデルよりも近いサンプルを生成しており、さまざまな形状のサンプルを生成していることが示唆されている。

表 2: CD ($\times 10^4$), EMD ($\times 10^2$) による再構成性能の評価

Category	Model	CD	EMD
Airplane	PointFlow	1.21	2.77
	SoftFlow	1.19	2.60
	Ours	1.07	2.23
Chair	PointFlow	10.09	6.42
	SoftFlow	11.04	6.60
	Ours	8.78	4.62
Car	PointFlow	6.54	5.16
	SoftFlow	6.82	5.08
	Ours	6.20	3.96

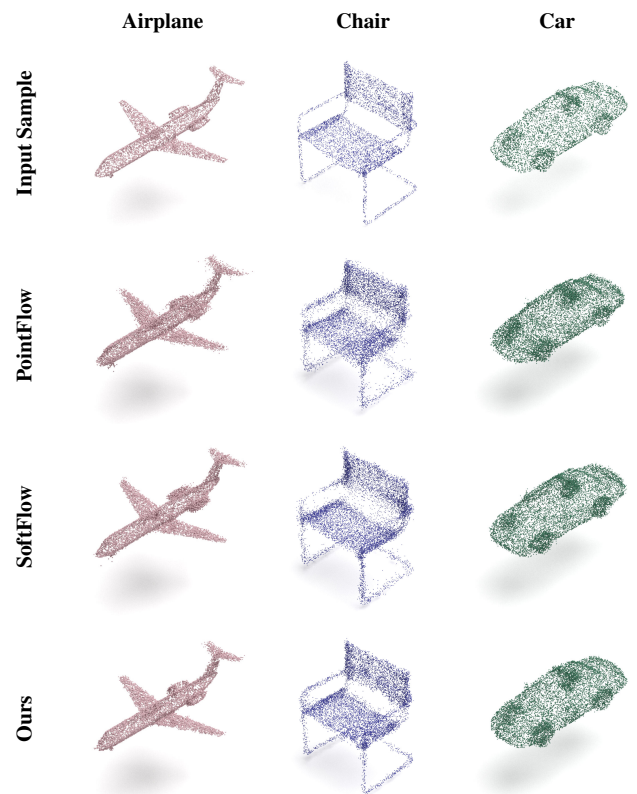


図 8: 再構成例

6.4 再構成タスク

再構成タスクでは、入力点群と再構成された点群間の距離 (CD, EMD) を測定し 5 回の平均を取り、表 2 に記載した。本節では、再構成タスクのみで学習した PointFlow と、生成タスクのみで学習した提案手法と SoftFlow を使用した。

表 2 より、すべてのカテゴリにおいて提案手法 (24 チャートを用いた) が比較手法の性能を超えていることがわかる。これは、椅子のカテゴリにおいて最も顕著であり、データセット内において足の形状や背もたれの穴の数の違い、つまりトポロジーの違いによるためであると考えられる。図 8 に提案手法がこのような形状を明確に再構成していることを示す。提案手法では飛行機の前輪や、車のミラーも再構成していることがわかる。対して、PointFlow と SoftFlow は 1 つの潜在変数から様々な形状を再構成しているため

に、椅子の背もたれや肘掛の穴の生成に失敗していることがわかる。

7. 結論

本論文では、複数のチャートを用いた点群のための flow-based generative model を提案した。適切に正規化を行うことで、各チャートは意味的なサブパーツに割り当てられ、様々なトポロジーを持つ形状を表現することが可能になった。Gumbel-Softmax により、計算コストはチャートを用いない場合と同程度に抑えることが可能になった。3つの2D合成データセットと3つの3D実用データセットを用いて性能を評価した結果、提案手法はさまざまな形状の点群を比較手法よりも優れた精度で生成可能であることが示された。チャートのコンセプトは flow-based generative model に限らず、今後は再帰的に解像度を増やすことで点群を生成する手法 [6], [22], [24] への応用を考えている。

謝辞 本研究は総務省 SCOPE (受付番号 172107101) の委託と、JSPS 科研費 (19H04172, 19K20344) の補助, JST CREST (JPMJCR1914) の支援を受けて実施された。

参考文献

- [1] Achlioptas, P., Diamanti, O., Mitliagkas, I. and Guibas, L.: Learning representations and generative models for 3d point clouds, *International Conference on Machine Learning (ICML)* (2018).
- [2] Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L. and Yu, F.: ShapeNet: An information-rich 3d model repository, *arXiv preprint arXiv:1512.03012* (2015).
- [3] Dinh, L., Sohl-Dickstein, J. and Bengio, S.: Density estimation using real NVP, *International Conference on Learning Representations (ICLR)* (2017).
- [4] Goodfellow, I. J., Pouget-abadie, J., Mirza, M., Xu, B. and Warde-farley, D.: Generative Adversarial Nets, *Advances in Neural Information Processing Systems (NIPS)* (2014).
- [5] Grathwohl, W., Chen, R. T. Q., Bettencourt, J., Sutskever, I. and Duvenaud, D.: FFDJORD: Free-form Continuous Dynamics for Scalable Reversible Generative Models, *International Conference on Learning Representations (ICLR)* (2019).
- [6] Hui, L., Xu, R., Xie, J., Qian, J. and Yang, J.: Progressive Point Cloud Deconvolution Generation Network, *European Conference on Computer Vision (ECCV)* (2020).
- [7] Jang, E., Gu, S. and Poole, B.: Categorical reparameterization with gumbel-softmax, *International Conference on Learning Representations (ICLR)* (2017).
- [8] Kim, H., Lee, H., Kang, W. H., Lee, J. Y. and Kim, N. S.: SoftFlow: Probabilistic Framework for Normalizing Flow on Manifolds, *arXiv preprint arXiv:2006.04604* (2020).
- [9] Kingma, D. P. and Ba, J.: Adam: A Method for Stochastic Optimization, *International Conference on Learning Representations (ICLR)* (2015).
- [10] Kingma, D. P. and Dhariwal, P.: Glow: Generative flow with invertible 1x1 convolutions, *Advances in Neural Information Processing Systems (NeurIPS)* (2018).
- [11] Kingma, D. P., Rezende, D. J. and Welling, M.: Semi-supervised Learning with Deep Generative Models, *Advances in Neural Information Processing Systems (NIPS)* (2014).
- [12] Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I. and Welling, M.: Improved variational inference with inverse autoregressive flow, *Advances in Neural Information Processing Systems (NIPS)* (2016).
- [13] Kingma, D. P. and Welling, M.: Auto-encoding variational bayes, *International Conference on Learning Representations (ICLR)* (2014).
- [14] Klovov, R. and Lempitsky, V.: Escape from Cells: Deep Kd-Networks for the Recognition of 3D Point Cloud Models, *International Conference on Computer Vision (ICCV)* (2017).
- [15] Li, C. L., Zaheer, M., Zhang, Y., Póczos, B. and Salakhutdinov, R.: Point cloud gan, *Deep Generative Models for Highly Structured Data, International Conference on Learning Representations (ICLR) Workshop* (2019).
- [16] Lopez-Paz, D. and Oquab, M.: Revisiting classifier two-sample tests, *International Conference on Learning Representations (ICLR)* (2017).
- [17] Lou, A., Lim, D., Katsman, I., Huang, L., Jiang, Q., Lim, S.-N. and De Sa, C.: Neural Manifold Ordinary Differential Equations, *Advances in Neural Information Processing Systems (NeurIPS)* (2020).
- [18] Mirza, M. and Osindero, S.: Conditional Generative Adversarial Nets, *arXiv preprint arXiv:1411.1784* (2014).
- [19] Qi, C. R., Su, H., Mo, K. and Guibas, L. J.: PointNet: Deep learning on point sets for 3D classification and segmentation, *Computer Vision and Pattern Recognition (CVPR)* (2017).
- [20] Qi, C. R., Yi, L., Su, H. and Guibas, L. J.: PointNet++: Deep hierarchical feature learning on point sets in a metric space, *Advances in Neural Information Processing Systems (NeurIPS)* (2017).
- [21] Rezende, D. J., Papamakarios, G., Racanière, S., Albergio, M. S., Kanwar, G., Shanahan, P. E. and Cranmer, K.: Normalizing Flows on Tori and Spheres, *International Conference on Machine Learning (ICML)* (2020).
- [22] Shu, D., Park, S. W. and Kwon, J.: 3D point cloud generative adversarial network based on tree structured graph convolutions, *International Conference on Computer Vision (ICCV)* (2019).
- [23] Su, H., Jampani, V., Sun, D., Maji, S., Kalogerakis, E., Yang, M. H. and Kautz, J.: SPLATNet: Sparse Lattice Networks for Point Cloud Processing, *Computer Vision and Pattern Recognition (CVPR)* (2018).
- [24] Valsesia, D., Fracastoro, G. and Magli, E.: Learning localized generative models for 3D point clouds via graph convolution, *International Conference on Learning Representations (ICLR)* (2019).
- [25] Wang, L., Huang, Y., Hou, Y., Zhang, S. and Shan, J.: Graph Attention Convolution for Point Cloud Semantic Segmentation, *Computer Vision and Pattern Recognition (CVPR)* (2019).
- [26] Wang, P. and Neumann, U.: Grid-GCN for Fast and Scalable Point Cloud Learning, *Computer Vision and Pattern Recognition (CVPR)* (2020).
- [27] Yan, X., Zheng, C., Li, Z., Wang, S. and Cui, S.: PointASNL: Robust Point Clouds Processing Using Nonlocal Neural Networks With Adaptive Sampling, *Computer Vision and Pattern Recognition (CVPR)* (2020).
- [28] Yang, G., Huang, X., Hao, Z., Liu, M. Y., Belongie, S. and Hariharan, B.: Pointflow: 3D point cloud generation with continuous normalizing flows, *International Conference on Computer Vision (ICCV)* (2019).
- [29] Zaheer, M., Kottur, S., Ravanbakhsh, S., Póczos, B., Salakhutdinov, R. and Smola, A. J.: Deep sets, *Advances in Neural Information Processing Systems (NeurIPS)* (2017).