CI-003

双方向シーンフロー推定と時間的サンプリングに基づく点群フレーム補間

Point Cloud Frame Interpolation based on Bidirectional Scene Flow Estimation and Temporal Sampling

松﨑 康平† Kohei Matsuzaki　野中 敬介† Keisuke Nonaka

概要

動的点群シーケンスのフレームレート向上に有効な技術として，点群フレーム補間に注目が集まっている．本稿では，双方向シーンフロー推定と時間的サンプリングに基づく点群フレーム補間手法を提案する．提案手法では，はじめに時間的に連続する二つの点群フレームから双方向シーンフロー推定によって個別に中間フレームを構築する．そして，補間時間を考慮した時間的サンプリングによって二つの中間フレームを単一のフレームに統合する．さらに，サンプリングのための真値ラベルを動的に作成する自己教師あり学習手法を導入する．三つの大規模な動的点群シーケンスデータセットを用いた評価実験を通じて，提案手法の有効性を確認する．

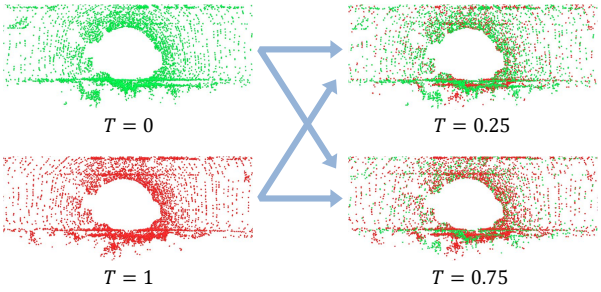

図 1. 補間時間を考慮した時間的サンプリングの概要．時間 $T = 0$ および $T = 1$ の点群から，補間時間に応じて異なる割合でを点群をサンプリングする．

1　はじめに

点群フレーム補間は，動的点群シーケンスにおいて時間的に連続するフレーム間の点群を滑らかに補間する技術である．動的点群シーケンスの取得に使用される代表的なセンサは LiDAR であり，フレームレートは一般的に 10 Hz から 20 Hz である．LiDAR はカメラや IMU のような他のセンサに比べてフレームレートが低いため，他のセンサのデータと統合する際に LiDAR のフレームが不足する可能性がある．また，LiDAR のフレームレートの低さに起因する時間的な不連続性は，物体追跡やナビゲーション [1,2] などの動的情報を活用するアプリケーションの性能を制限する可能性がある．これらの場合には動的点群シーケンスのフレームレート向上が求められ，点群フレーム補間はその解決法になり得る．

点群フレーム補間における先駆的な研究は映像フレーム補間 [3–6] 技術を活用し，時間的に連続する深度マップの中間フレームを生成することによって，疑似 LiDAR 点群のフレーム補間を実現する [7,8]．NeuralPCI [9] は四次元のニューラル場を用いて実際の LiDAR 点群に対する高精度なフレーム補間を実現するが，実行時最適化を行うため効率性が低い．PointINet [10] と FastPCI [11] は，フレーム間の点の変位を表す双方向シーンフローを推定することにより，実際の LiDAR 点群に対して効率的なフレーム補間を実現する．これらの手法は精度と効率性の両面で優れた性能を持つが，双方向シーンフロー推定によって二つの中間フレームが構築されるため，それらを単一のフレームに統合する必要がある．

二つの中間フレームを統合するために，PointINet [10] は適応的サンプリングと適応的 k 近傍クラスタリングによって得られた点集合の加重和として，新たに点を生成する点融合モジュールを導入する．ただし，適応的サンプリングは後段の処理を考慮しておらず，フレーム補間に対して最適な点をサンプリングするとは限らない．また，二つの中間フレームの統合は点の生成に基づくため，シーンフロー推定の誤差と生成に起因する誤差が累積し，点群フレーム補間性能が低下するおそれがある．FastPCI [11] は，はじめにシーンフロー推定によって構築された点群の座標を補正し，その後 PointINet の点融合モジュールを用いて二つの中間フレームを単一のフレームに統合する．したがって，PointINet と同様に，点融合モジュールによって生成に起因する誤差が発生するおそれがある．

本稿では，双方向シーンフロー推定と時間的サンプリングに基づく点群フレーム補間手法を提案する．提案手法は，はじめに双方向シーンフロー推定を用いて二つの中間フレームを構築する．シーンフロー推定誤差と生成に起因する誤差の累積を回避するために，非生成的な点群サンプリングによって二つの中間フレームを単一のフレームへ統合する．ここでは，図 1 に示すように二つの中間フレームから補間時間に応じて異なる割合で点をサンプリングする，時間的サンプリングを提案する．また，サンプリングされた点群の幾何学的品質を改善するために，注意に基づく座標補正を導入する．さらに，サンプリングモデルの学習のために，点ごとの真値ラベルを動的に作成する自己教師あり学習手法を提案する．

本研究の主な貢献は以下にまとめられる．

- 補間時間を考慮した時間的サンプリングに基づいて，双方向シーンフロー推定によって構築される中間フレームを統合する点群フレーム補間手法を提案する．
- 点群サンプリングのための点ごとの真値ラベルを動的に作成可能な自己教師あり学習手法を提案する．
- 三つの大規模な動的点群シーケンスデータセットを用いた実験により，提案手法が従来手法に比べて優れた点群フレーム補間性能を達成することを示す．

† 株式会社 KDDI 総合研究所　KDDI Research, Inc.

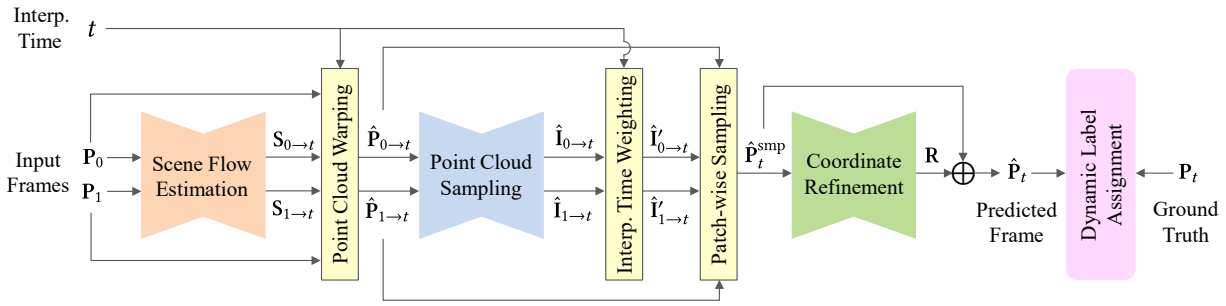

図 2. 提案手法のフレームワーク．入力は補間時間 $T = t \in (0, 1)$ および，時間 $T = 0$ および $T = 1$ に対応する二つの点群フレーム $\mathbf{P}^0 \in \mathbb{R}^{N\times 3}$ および $\mathbf{P}^1 \in \mathbb{R}^{N\times 3}$ である．出力は補間時間 $T = t$ に対応する補間点群フレーム $\hat{\mathbf{P}}_t \in \mathbb{R}^{N\times 3}$ である．動的ラベル割り当ては $\hat{\mathbf{P}}_t$ と真値点群フレーム $\mathbf{P}_t \in \mathbb{R}^{N\times 3}$ を用いて実行する．

2　関連研究

2.1　点群フレーム補間

映像フレーム補間に基づく手法では，深度マップ補間を介して疑似 LiDAR 点群を生成することにより，点群フレーム補間を実現する [7, 8]．PointINet [10] は実際の LiDAR 点群に対する点群フレーム補間を実現した初の手法であり，双方向シーンフロー推定に基づいて構築された二つの中間フレームを統合することで単一のフレームを構築する．NeuralPCI [9] はフレーム間の非線形モーションに対処するために，四次元のニューラル場を用いて点群フレームを補間する．この手法は高精度に補間フレームを予測することができるが，実行時最適化を必要とするため，推論時間が非常に長くなるという課題がある．FastPCI [11] はモーションと構造の一貫性を考慮した Pyramid Convolution-Transformer を用いてシーンフローを推定することにより，高精度かつ効率的な点群フレーム補間を実現する．しかし，シーンフロー推定によって構築された二つの中間フレームの統合時に新たに点を生成するため，生成に起因する誤差が生じる．

2.2　点群サンプリング

最も代表的な点群サンプリング手法はランダムサンプリングと最遠方点サンプリング [12, 13] である．近年では，後段のでタスクに対して最適な性能を発揮する点群を得るために，学習に基づくサンプリング手法が提案されている [14–19]．ただし，これらの手法はサンプリングの際に新たに点を生成するため，点の座標に対して生成に起因する誤差が生じる可能性がある．新たに点を生成せずに直接的に点を選択可能な，学習に基づくサンプリング手法も提案されている [20–22]．しかし，これらの手法はエッジ点や局所的に集中した点のような，特定の性質を持つ点を選択することを目的とする．そのため，点群フレーム補間に対して最適な点をサンプリングするとは限らない．さらに，上述したすべての手法は時間を考慮していない．適応的サンプリング [10] は時間を考慮したサンプリング手法であるが，ランダムサンプリングに基づいて設計されており，後段のタスクに対して効果的な点群をサンプリングするとは限らない．

3　提案手法

補間時間を考慮した時間的サンプリングおよび自己教師あり学習に基づく点群フレーム補間手法を提案する．

図 2 に提案手法のフレームワークを示す．提案手法への入力は補間時間 $T = t \in (0, 1)$ および，時間 $T = 0$ および $T = 1$ に対応する時間的に連続する二つの点群フレーム $\mathbf{P}_0 \in \mathbb{R}^{N\times 3}$ および $\mathbf{P}_1 \in \mathbb{R}^{N\times 3}$ である．ここで N は点群フレームの点の個数を表す．提案手法の出力は補間時間 $T = t$ に対応する予測された補間点群フレーム $\hat{\mathbf{P}}_t \in \mathbb{R}^{N\times 3}$ である．提案手法では，真値点群フレーム $\mathbf{P}_t \in \mathbb{R}^{N\times 3}$ を用いてモデルを学習させる．

提案手法は，はじめに双方向シーンフロー推定を実行し，点群フレーム $\mathbf{P}_0$ および $\mathbf{P}_1$ のそれぞれから時間 $T = t$ で補間するための中間フレーム $\hat{\mathbf{P}}_{0\to t} \in \mathbb{R}^{N\times 3}$ および $\hat{\mathbf{P}}_{1\to t} \in \mathbb{R}^{N\times 3}$ を構築する．ここで $0 \to t$ および $1 \to t$ はそれぞれ，時間 $T = 0$ から順方向のシーンフローおよび $T = 1$ から逆方向のシーンフローに基づいて $T = t$ のフレームを構築することを表す．次に，$\hat{\mathbf{P}}_{0\to t}$ および $\hat{\mathbf{P}}_{1\to t}$ から補間時間を考慮した時間的サンプリングによって合計で N 個の点を選択し，点群フレーム $\hat{\mathbf{P}}_t^{\mathrm{smp}} \in \mathbb{R}^{N\times 3}$ を構築する．その後，幾何学的品質を改善するために，点群フレーム $\hat{\mathbf{P}}_t^{\mathrm{smp}}$ に対して注意に基づく座標補正を実行する．そして，座標補正後の点群フレーム $\hat{\mathbf{P}}_t$ を補間点群フレームとして出力する．さらに，サンプリングモデルの学習のために，補間点群フレーム $\hat{\mathbf{P}}_t$ に対する動的ラベル割り当てによって点ごとの真値ラベルを作成する．

以下ではシーンフロー推定について第 3.1 節で，時間的サンプリングについて第 3.2 節で，注意に基づく座標補正について第 3.3 節で，自己教師あり学習について第 3.4 節で，損失関数について第 3.5 節で詳細に説明する．

3.1　シーンフロー推定

はじめに，二つの入力点群フレーム $\mathbf{P}_0$ および $\mathbf{P}_1$ から双方向のシーンフローを推定する．本研究においてはシーンフロー推定モデルとして Pyramid Convolution-Transformer [11] を使用するが，提案手法では異なるシーンフロー推定モデルも適用可能である．このモデルは入力点群から複数の階層における順方向特徴および逆方向特徴を抽出する．入力点群は階層 $l = 0$ では元々の点数 N を維持し，階層 $l = 1$ と $l = 2$ では最遠方点サンプリング法を用いてそれぞれ 4 分の 1 と 32 分の 1 の点数へダウンサンプリングされる．各階層における点群を $\mathbf{P}_0^l$ および $\mathbf{P}_1^l$ $(l = 0, 1, 2)$ と表記する．階層ごとに抽出した特徴を用いて順方向のシーンフロー $\mathbf{S}_{0\to t}^l \in \mathbb{R}^{N_l\times 3}$ および逆方向のシーンフロー $\mathbf{S}_{1\to t}^l \in \mathbb{R}^{N_l\times 3}$ を推定する．こ

こで N_l は l 番目の階層における点群の点の個数を表す．順方向のシーンフローを補間時間 $T=t$ を用いて線形補間し，入力点群フレーム $\mathbf{P}_0^l$ に加算することにより，中間フレーム $\hat{\mathbf{P}}_{0\to t}^l$ を構築する．同様に，逆方向のシーンフロー推定による中間フレーム $\hat{\mathbf{P}}_{1\to t}^l$ を $(1-t)$ を用いて構築する．$\hat{\mathbf{P}}_{0\to t}^l$ および $\hat{\mathbf{P}}_{1\to t}^l$ は次式で表される．

$$\hat{\mathbf{P}}_{0\to t}^l = \mathbf{P}_0^l + t\mathbf{S}_{0\to t}^l,\quad \hat{\mathbf{P}}_{1\to t}^l = \mathbf{P}_1^l + (1-t)\mathbf{S}_{1\to t}^l \tag{1}$$

以下では階層 $l=0$ における $\hat{\mathbf{P}}_{0\to t}^l$ および $\hat{\mathbf{P}}_{1\to t}^l$ をそれぞれ $\hat{\mathbf{P}}_{0\to t}$ および $\hat{\mathbf{P}}_{1\to t}$ と表記する．

3.2 時間的サンプリング

提案手法では，二つの中間フレーム $\hat{\mathbf{P}}_{0\to t}$ および $\hat{\mathbf{P}}_{1\to t}$ から，補間時間を考慮した時間的サンプリングによって単一の点群フレームを構築する．学習に基づく点群サンプリング手法の多くは新たに点を生成する [14–19,23] ため，シーンフロー推定誤差と生成に起因する誤差を累積し，点群フレーム補間性能が低下するおそれがある．この問題に対処するために，点ごとに重要性を予測することによって，直接的に点を選択するサンプリング手法を提案する．提案手法では，補間時間 $T=t$ における補間点群フレームに対する二つの中間フレーム $\hat{\mathbf{P}}_{0\to t}$ および $\hat{\mathbf{P}}_{1\to t}$ の類似性に基づく重要性の重み付けにより，補間時間に応じて異なる割合で点群をサンプリングする時間的サンプリングを実現する．さらに，点密度を維持するために，局所パッチごとに点をサンプリングする．

サンプリングモデルは Point Transformer [24] に基づく U-Net と予測のための線形層で構成する．はじめに二つの中間フレーム $\hat{\mathbf{P}}_{0\to t}$ および $\hat{\mathbf{P}}_{1\to t}$ から点ごとの特徴を抽出する．次に，特徴を線形層に入力し，点ごとの重要性を予測する．中間フレーム $\hat{\mathbf{P}}_{0\to t}$ および $\hat{\mathbf{P}}_{1\to t}$ に対する予測された重要性をそれぞれ $\hat{\mathbf{I}}_{0\to t}\in\mathbb{R}^N$ および $\hat{\mathbf{I}}_{1\to t}\in\mathbb{R}^N$ と表記する．

$\hat{\mathbf{P}}_{0\to t}$ および $\hat{\mathbf{P}}_{1\to t}$ は，それらに対応する入力点群フレームの時間 T が t に近いほど，真値点群フレーム $\mathbf{P}_t$ に類似する可能性がある．したがって，予測された重要性 $\hat{\mathbf{I}}_{0\to t}$ および $\hat{\mathbf{I}}_{1\to t}$ に対する，補間時間を考慮した重み付けを導入する．補間時間 $T=t$ における重み $W(t)\in\mathbb{R}$ は正規分布関数を用いて次式で定義する．

$$W(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{t^2}{2\sigma^2}} \tag{2}$$

ここで σ は調整可能なパラメータである．次式により，$W(t)$ を用いて補間時間 $T=t$ に依存する重み付けされた重要性 $\hat{\mathbf{I}}'_{0\to t}\in\mathbb{R}^N$ および $\hat{\mathbf{I}}'_{1\to t}\in\mathbb{R}^N$ を求める．

$$\hat{\mathbf{I}}'_{0\to t} = W(t)\hat{\mathbf{I}}_{0\to t},\quad \hat{\mathbf{I}}'_{1\to t} = W(1-t)\hat{\mathbf{I}}_{1\to t} \tag{3}$$

ここでは $\hat{\mathbf{I}}_{1\to t}$ は逆方向のシーンフロー推定により構築された点群に対応するため，$W(1-t)$ を使用する．

重み付けされた重要性 $\hat{\mathbf{I}}'_{0\to t}$ および $\hat{\mathbf{I}}'_{1\to t}$ の降順に二つの中間フレーム $\hat{\mathbf{P}}_{0\to t}$ および $\hat{\mathbf{P}}_{1\to t}$ から合計で N 個の点を選択する．$2N$ 個の点から N 個の点を選択するため全体のサンプリング率は 50% であるが，各中間フレームからの個々のサンプリング率は固定されない．重要性は特定の局所領域で高くなり，結果として選択される点の局所的な密度に偏りが生じるおそれがある．この問題に対処し，点密度を維持するために，パッチごとのサンプリング手法を提案する．具体的には，はじめに $\hat{\mathbf{P}}_{0\to t}$ と $\hat{\mathbf{P}}_{1\to t}$ を単純に統合し，$2N$ 個の点で構成された点群を得る．そして，効率性の高い点群のグループ化手法 [24] を用いて，統合された点群を同数の点で構成される局所パッチの集合に分割する．図 3 に局所パッチの例を示す．最後に，局所パッチごとに重み付けされた重要性の降順に 50% の点を選択し，選択されたすべての点を統合する．結果として得られた N 個の点で構成される点群フレームを $\hat{\mathbf{P}}_t^{\mathrm{smp}}\in\mathbb{R}^{N\times 3}$ と表記する．

図 3. 局所パッチの集合の可視化．同一の局所パッチに属する点を同色で表す．

3.3 注意に基づく座標補正

提案手法は注意に基づく座標補正モデルを用いて点群フレーム $\hat{\mathbf{P}}_t^{\mathrm{smp}}$ の幾何学的品質を改善する．このモデルは，はじめに Point Transformer [24] に基づく U-Net を用いて点ごとに D 次元の特徴を抽出する．結果として得られる特徴の集合を $\mathbf{F}=\{\mathbf{f}_i\in\mathbb{R}^D\}_{i=1}^N$ と表記する．その後，点群フレーム $\hat{\mathbf{P}}_t^{\mathrm{smp}}$ の各点 $\hat{\mathbf{p}}_i^{\mathrm{smp}}$ の k 近傍点を探索し，インデックスの集合 $\mathcal{N}(\hat{\mathbf{p}}^{\mathrm{smp}})$ を得る．そして，次式のように k 近傍点との座標の残差に位置符号化 [25] を適用し，多層パーセプトロン (Multi-Layer Perceptron, MLP) を用いて D 次元の特徴を得る．

$$\mathbf{E} = \Theta_{\mathbf{E}}\big(\gamma(\hat{\mathbf{p}}_i^{\mathrm{smp}} - \hat{\mathbf{p}}_j^{\mathrm{smp}})\big),\ \forall j\in\mathcal{N}(\hat{\mathbf{p}}_i^{\mathrm{smp}}) \tag{4}$$

ここで $\gamma(\cdot)$ は位置符号化関数，$\Theta_{\mathbf{E}}$ は正規化線形ユニット (Rectified Linear Unit, ReLU) [26] 関数を持つ二層の MLP である．個別の線形層 Φ_Q，Φ_K，Φ_V を用いてクエリ $\mathbf{Q}\in\mathbb{R}^{N\times D}$，キー $\mathbf{K}\in\mathbb{R}^{N\times D\times k}$，バリュー $\mathbf{V}\in\mathbb{R}^{N\times D\times k}$ を次式で求める．

$$\mathbf{Q}=\Phi_{\mathbf{Q}}(\mathbf{f}_i),\ \mathbf{K}=\Phi_{\mathbf{K}}(\mathbf{f}_j),\ \mathbf{V}=\Phi_{\mathbf{V}}(\mathbf{f}_j),\ \forall j\in\mathcal{N}(\hat{\mathbf{p}}_i^{\mathrm{smp}}) \tag{5}$$

k 近傍の次元にわたって複製した $\mathbf{Q}\in\mathbb{R}^{N\times D\times k}$ および $\mathbf{E}\in\mathbb{R}^{N\times D\times k}$ を用いて，次式で注意重み $\mathbf{A}\in\mathbb{R}^{N\times D\times k}$ を得る．

$$\mathbf{A} = \rho\big(\Theta_{\mathbf{A}}(\mathbf{Q}-\mathbf{K})+\mathbf{E}\big) \tag{6}$$

ここで $\rho(\cdot)$ はソフトマックス関数，$\Theta_{\mathbf{A}}$ は ReLU 関数を持つ二層の MLP である．注意重みを用いて集約した特徴 $\mathbf{G}\in\mathbb{R}^{N\times D}$ および座標補正ベクトル $\mathbf{R}\in\mathbb{R}^{N\times 3}$ を次式で求める．

$$\mathbf{G} = \mathbf{A}^\top(\mathbf{V}+\mathbf{E}),\ \mathbf{R}=\Phi_{\mathbf{R}}(\mathbf{G}) \tag{7}$$

ここで $\Phi_{\mathbf{R}}$ は座標補正ベクトル $\mathbf{R}$ を予測するための線形層を表す．最後に，$\mathbf{R}$ を用いて点群フレーム $\hat{\mathbf{P}}_t^{\mathrm{smp}}$ を次式で補正する．

$$\hat{\mathbf{P}}_t = \hat{\mathbf{P}}_t^{\mathrm{smp}} + \mathbf{R} \tag{8}$$

提案手法は補間時間 $T = t$ に対応する予測された補間点群フレームとして $\hat{\mathbf{P}}_t$ を出力する．

3.4 自己教師あり学習

提案手法はシーンフロー推定モデル，サンプリングモデル，座標補正モデルを一括で学習させる．しかし，モデルの学習時に構築される点群フレームに対して効果的なサンプリングを実現するための，点ごとの真値ラベルを事前に作成することは困難である．この問題に対処するために，学習時に点ごとの真値ラベルを作成する自己教師あり学習手法を提案する．提案手法では，予測された補間点群フレーム $\hat{\mathbf{P}}_t$ の各点に対して重要性を表す点ごとの二値ラベルを動的に割り当てる．具体的には，真値点群フレーム $\mathbf{P}_t$ の点 $\mathbf{p}$ の最近傍に位置する点 $\hat{\mathbf{p}} \in \hat{\mathbf{P}}_t$ に高い重要性を与える．点 $\hat{\mathbf{p}}$ に対する重要性の真値ラベルを $\mathbf{I}(\hat{\mathbf{p}}) \in \{0, 1\}$ と表記する．学習中に次式で動的に点 $\hat{\mathbf{p}}$ にラベル $\mathbf{I}(\hat{\mathbf{p}})$ を割り当てる．

$$\mathbf{I}(\hat{\mathbf{p}}) = \begin{cases} 1 & \text{if } \mathbf{p} \in \mathbf{P}_t \wedge \hat{\mathbf{p}} \in \arg\min\limits_{\hat{\mathbf{p}} \in \hat{\mathbf{P}}_t} \|\mathbf{p} - \hat{\mathbf{p}}\|_2^2 \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

図 4 にこの動的ラベル割り当ての模式図を示す．点 $\hat{\mathbf{p}}$ は $\mathbf{P}_t$ 内の複数の点に対する最近傍点になる可能性がある一方，$\mathbf{P}_t$ 内のいかなる点にとっても最近傍点にはならない可能性もある．

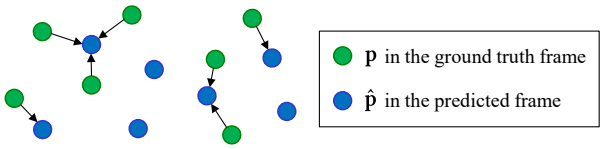

図 4. 動的ラベル割り当ての模式図．点 $\hat{\mathbf{p}}$ が点 $\mathbf{p}$ の最近傍点の場合には 1，それ以外の場合には 0 を割り当てる．

点 $\hat{\mathbf{p}}$ に対する予測された重要性を $\hat{\mathbf{I}}(\hat{\mathbf{p}}) \in \mathbb{R}$ と表記する．$\hat{\mathbf{I}}(\hat{\mathbf{p}})$ はサンプリングされた点群 $\hat{\mathbf{P}}_t^{\mathrm{smp}}$ の点のインデックスを用いて $\hat{\mathbf{I}}_{0\to t}$ および $\hat{\mathbf{I}}_{1\to t}$ から取得する．点群サンプリングの損失として $\mathbf{I}(\hat{\mathbf{p}})$ と $\hat{\mathbf{I}}(\hat{\mathbf{p}})$ の間で次式で表される重み付き二値交差エントロピー損失 $\mathcal{L}_{\mathrm{bce}}$ を計算する．

$$\begin{aligned} \mathcal{L}_{\mathrm{bce}} = -\frac{1}{|\hat{\mathbf{P}}_t|} \sum_{\hat{\mathbf{p}} \in \hat{\mathbf{P}}_t} \Big[& \lambda \mathbf{I}(\hat{\mathbf{p}}) \log\Big(s\big(\hat{\mathbf{I}}(\hat{\mathbf{p}})\big)\Big) \\ & + \big(1 - \mathbf{I}(\hat{\mathbf{p}})\big) \log\Big(1 - s\big(\hat{\mathbf{I}}(\hat{\mathbf{p}})\big)\Big) \Big] \end{aligned} \tag{10}$$

ここで λ と $s(\cdot)$ はそれぞれ重みパラメータとシグモイド関数を表す．

3.5 損失関数

損失の計算には次式で定義される Chamfer Distance (CD) [27] を使用する．

$$d_{\mathrm{CD}}(\mathbf{X}, \mathbf{Y}) = \frac{1}{|\mathbf{X}|} \sum_{\mathbf{x} \in \mathbf{X}} \min_{\mathbf{y} \in \mathbf{Y}} \|\mathbf{x} - \mathbf{y}\|_2^2 + \frac{1}{|\mathbf{Y}|} \sum_{\mathbf{y} \in \mathbf{Y}} \min_{\mathbf{x} \in \mathbf{X}} \|\mathbf{x} - \mathbf{y}\|_2^2 \tag{11}$$

ここで $\mathbf{X} \subseteq \mathbb{R}^3$ および $\mathbf{Y} \subseteq \mathbb{R}^3$ は二つの点群を表す．FastPCI [11] と同様に，補間時間 $T = t$ における予測された補間点群フレーム $\hat{\mathbf{P}}_t$ の真値点群フレーム $\mathbf{P}_t$ に対する予測損失 $\mathcal{L}_{\mathrm{intp}}$ を次式で定義する．

$$\mathcal{L}_{\mathrm{intp}} = d_{\mathrm{CD}}(\mathbf{P}_t, \hat{\mathbf{P}}_t) \tag{12}$$

順方向の中間フレーム $\hat{\mathbf{P}}_{0\to t}$ および逆方向の中間フレーム $\hat{\mathbf{P}}_{1\to t}$ のサイクル一貫性損失 $\mathcal{L}_{\mathrm{cyc}}$ を次式で定義する．

$$\mathcal{L}_{\mathrm{cyc}} = d_{\mathrm{CD}}(\mathbf{P}_t, \hat{\mathbf{P}}_{0\to t}) + d_{\mathrm{CD}}(\mathbf{P}_t, \hat{\mathbf{P}}_{1\to t}) \tag{13}$$

シーンフロー推定モデルにおける階層ごとの予測損失の総和を表すマルチスケール損失 $\mathcal{L}_{\mathrm{ms}}$ を次式で定義する．

$$\mathcal{L}_{\mathrm{ms}} = \sum_l \alpha_l d_{\mathrm{CD}}(\mathbf{P}_t^l, \hat{\mathbf{P}}_{0\to t}^l) \tag{14}$$

ここで l は階層番号，α_l は l 番目の階層に対する重み，$\mathbf{P}_t^l$ は真値点群フレーム $\mathbf{P}_t$ から得られる l 番目の階層の点群フレームを表す．評価実験では $\alpha_0 = 0.05$, $\alpha_1 = 0.1$, $\alpha_2 = 0.2$ に設定する．

さらに，式 (10) で定義される点群サンプリングのための重み付き二値交差エントロピー損失 $\mathcal{L}_{\mathrm{bce}}$ も導入する．したがって，次式で定義される損失 $\mathcal{L}$ を最小化するようにモデルを学習させる．

$$\mathcal{L} = \mathcal{L}_{\mathrm{intp}} + \mathcal{L}_{\mathrm{cyc}} + \mathcal{L}_{\mathrm{ms}} + \beta \mathcal{L}_{\mathrm{bce}} \tag{15}$$

ここで β は他の損失項とのバランスを調整するためのパラメータである．評価実験では $\beta = 0.01$ に設定する．

4 評価実験

4.1 実験設定

4.1.1 評価用データセット

点群フレーム補間性能を評価するために，大規模な動的点群シーケンスデータセットである KITTI Odometry データセット [30]，Argoverse 2 Sensor データセット [31] および nuScenes データセット [32] を使用する．KITTI Odometry および Argoverse 2 Sensor データセットは 10 Hz の LiDAR センサを用いて取得された点群で構成される．nuScenes データセットは 20 Hz の LiDAR センサを用いて取得された点群で構成されるため，他の二つのデータセットとフレームレートを揃えるために 10 Hz にダウンサンプリングする．すべてのデータセットにおいて，フレーム補間手法への入力は 10 Hz から 2.5 Hz でサンプリングされた二つのフレームである．これらの二つの入力フレームの間で補間する三つのフレームをそれぞれ Frame-1, Frame-2 および Frame-3 と表記する．これらの補間フレームに対応する補間時間はそれぞれ $T = 0.25$, 0.5, 0.75 である．すべてのデータセットにおいて，各点群フレームを構成する点の個数は $N = 8192$ である．

4.1.2 評価指標

定量的な評価指標として CD と Earth Mover's Distance (EMD) [27] を使用する．これらは二つの点群間の類似性を測るための指標である．CD は一方の点群の各点からもう一方の点群内の最近傍点までの距離の平均を表し，式 (11) で定義される．EMD は一方の点群を移動させることによってもう一方の点群に変換するための，各点の最小平均距離を表す．点数の等しいの二つの点群

表 1. 点群フレーム補間性能の比較．Frame-1 から Frame-3 は入力フレーム間の三通りの補間フレームを表す．CD と EMD は数値が低いほど良い結果であることを表す．

Datasets	Methods	Frame-1		Frame-2		Frame-3		Average	
		CD ↓	EMD ↓	CD ↓	EMD ↓	CD ↓	EMD ↓	CD ↓	EMD ↓
KITTI Odometry	PointPWC-Net [28]	0.64	71.14	0.80	91.91	0.91	60.35	0.78	74.46
	NSFP [29]	0.58	70.53	0.68	84.76	1.95	115.42	1.07	90.24
	PointINet [10]	0.72	55.25	0.82	77.87	0.83	73.74	0.79	69.19
	NeuralPCI [9]	0.64	52.61	0.85	62.73	0.64	52.15	0.71	55.83
	FastPCI [11]	0.54	51.23	0.61	59.84	0.58	50.27	0.58	53.78
	Ours	**0.49**	**47.74**	**0.50**	**46.52**	**0.47**	**47.79**	**0.49**	**47.35**
Argoverse 2 Sensor	PointPWC-Net [28]	0.90	56.44	1.07	79.86	1.26	65.32	1.07	67.20
	NSFP [29]	0.72	62.30	0.85	73.89	2.14	98.99	1.24	78.40
	PointINet [10]	0.83	57.89	1.25	67.73	1.06	62.97	1.05	62.86
	NeuralPCI [9]	0.68	55.03	0.88	65.93	0.69	55.30	0.75	58.75
	FastPCI [11]	0.68	54.99	0.77	64.29	0.74	54.59	0.73	57.95
	Ours	**0.66**	**51.52**	**0.76**	**54.32**	**0.65**	**53.07**	**0.69**	**52.97**
nuScenes	PointPWC-Net [28]	1.52	172.31	1.39	224.64	2.16	181.39	1.69	192.78
	NSFP [29]	1.10	173.03	1.33	212.32	4.37	319.56	2.27	234.97
	PointINet [10]	1.48	183.03	1.67	202.10	1.50	186.51	1.55	190.54
	NeuralPCI [9]	**1.00**	163.10	1.37	205.24	1.06	168.98	1.15	179.11
	FastPCI [11]	1.02	162.78	1.28	205.75	1.03	152.39	1.11	173.64
	Ours	**1.00**	**147.75**	**1.13**	**151.81**	**1.01**	**152.23**	**1.05**	**150.60**

$\mathbf{X} \subseteq \mathbb{R}^3$ と $\mathbf{Y} \subseteq \mathbb{R}^3$ の EMD は次式で定義される．

$$d_{\mathrm{EMD}}(\mathbf{X}, \mathbf{Y}) = \min_{\phi:\mathbf{X}\to\mathbf{Y}} \frac{1}{|\mathbf{X}|} \sum_{\mathbf{x}\in\mathbf{X}} \|\mathbf{x} - \phi(\mathbf{x})\|_2 \qquad (16)$$

ここで $\phi : \mathbf{X} \to \mathbf{Y}$ は $\mathbf{X}$ から $\mathbf{Y}$ への単射を表す．

4.1.3 実装の詳細

提案手法は PyTorch [33] を用いて実装する．モデルの学習には，初期学習率 0.0001 の AdamW Optimizer [34] およびコサインアニーリングスケジューラ [35] を使用する．バッチサイズを 4 に設定し，KITTI Odometry，Argoverse 2 Sensor，nuScenes データセットのそれぞれについて 700，800，900 エポックの学習を行う．時間的サンプリングおよび座標補正のための Point Transformer における出力特徴次元はどちらも 32 に設定する．式 (2) におけるパラメータは $\sigma = 1/\sqrt{2}$ に設定する．パッチごとのサンプリングのためのパッチサイズは nuScenes データセットでは 16，その他のデータセットでは 8 に設定する．座標補正における k 近傍数を $k = 4$ に設定する．式 (10) におけるパラメータは $\lambda = 10$ に設定する．すべての実験は NVIDIA RTX 6000 Ada GPU，AMD Ryzen Threadripper PRO 7985WX CPU (3.20 GHz) および 128 GB の RAM を搭載したコンピュータで実施する．

4.2 従来手法との比較

提案手法と最先端の点群フレーム補間手法である PointINet [10]，NeuralPCI [9] および FastPCI [11] の定量的な性能評価を行う．また，シーンフロー推定手法である PointPWC-Net [28] と NSFP [29] も本評価に含める．表 1 にすべての手法の点群フレーム補間性能を示す．提案手法はすべてのデータセットにおいて最高の性能を達成することがわかる．提案手法は二番目に優れた性能を達成する FastPCI と比べて，Frame-1 から Frame-3 のすべてにわたって性能を改善する．その理由としては，FastPCI では二つの中間フレームを統合する際に点の生成に起因する誤差が発生するのに対して，提案手法は非生成的なサンプリングによって中間フレームを統合することにより，統合時に誤差の発生を回避するためと考えられる．さらに，中間フレームを統合する前に点群の座標を補正する FastPCI とは対称的に，提案手法は中間フレームの統合後に点群の座標を補正することにより，効果的に幾何学的品質を改善する．

表 1 で優れた結果を示した NeuralPCI，FastPCI および提案手法の定性的な評価を行う．図 5 にこれらの手法によって予測された補間点群フレームと真値点群フレームを示す．NeuralPCI は鋭いエッジを含む点群を予測する一方で，赤枠で強調される矩形領域のように部分的な欠損や歪が生じることがある．FastPCI は大域的な構造を忠実に予測できるが，予測された点群にはしばしばノイズが発生することがわかる．対称的に，提案手法は忠実な大域的構造を持つノイズの少ない点群を予測しており，最も真値点群フレームに近い結果が得られる．

また，NeuralPCI，FastPCI および提案手法の推論時間を比較する．表 2 にこれらの手法で単一のフレームを補間する場合の平均推論時間を示す．NeuralPCI は実行時最適化を必要とするため，他の手法に比べて推論時間が大幅に長くなる．FastPCI は計算効率を重視してモデル

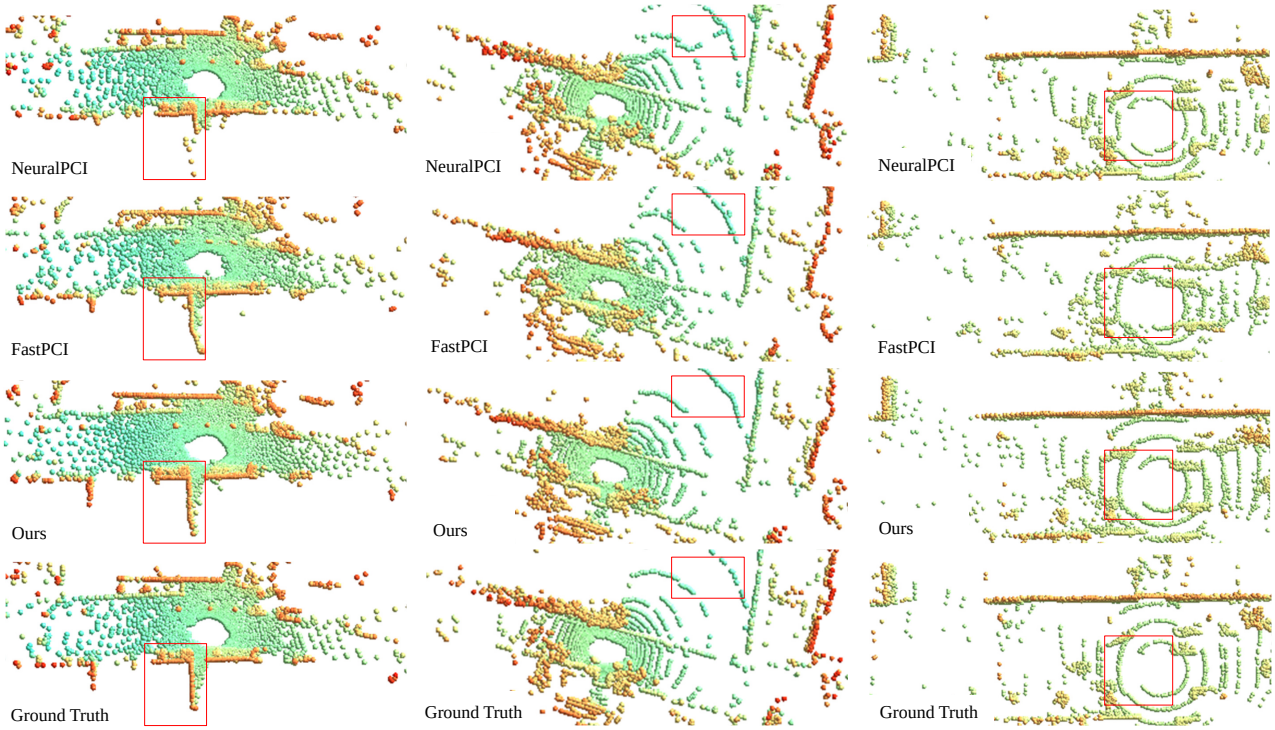

図 5. 点群フレーム補間結果の可視化．(a)，(b)，(c) はそれぞれ KITTI Odometry，Argoverse 2 Sensor および nuScenes データセットでの結果に対応する．点の色は鉛直方向の高さを表す．

表 2. 平均推論時間 [ms] の比較.

	NeuralPCI [9]	FastPCI [11]	Ours
Inference Time	30843 ms	72 ms	96 ms

が設計されているため，最も短い推論時間を達成する．提案手法は FastPCI に比べて低速だが同等の桁数の推論時間を達成しており，効率的な推論が可能である．

4.3 構成要素の影響

提案手法の主な構成要素である補間時間重み付け (Interpolation Time Weighting, ITW)，パッチごとのサンプリング (Patch-Wise Sampling, PWS)，注意に基づく座標補正 (Attention-based Coordinate Refinement, ACR)，自己教師あり学習 (Self-Supervised Learning, SSL) の点群フレーム補間性能への影響を評価する．表 3 に KITTI Odometry データセットでの点群フレーム補間性能を示す．一行目は提案手法の結果を示しており，最良の結果を達成することが確認できる．二行目は提案手法から ITW を除外した場合の結果を示しており，Frame-1 および Frame-3 で大きな性能低下が見られた．その原因は二つの中間フレーム $\hat{\mathbf{P}}_{0\to t}$ および $\hat{\mathbf{P}}_{1\to t}$ は対応する入力点群フレームの時間 T が t に近いほど真値点群フレーム $\mathbf{P}_t$ に類似する可能性があるにもかかわらず，時間情報を十分に活用できていないためである．Frame-2 は二つの入力点群フレームの中間の時間に対応するため，比較的影響が少ない．三行目は PWS を除外した場合の結果を示しており，特に Frame-2 で大幅に性能が低下することがわかる．これは，Frame-2 では二つの中間フレームから均等に近い点数がサンプリングされるため，局所的な点密度を維持することが困難なためである．図 6 は Frame-2 において一行目と三行目の設定で予測された補間点群フレームと真値点群フレームを可視化しており，PWS によって幾何学的品質が改善することが確認できる．四行目は ACR を除外した場合の結果を示しており，Frame-1 から Frame-3 にわたって大幅に性能が低下することがわかる．したがって，ACR によってサンプリングされた点群の幾何学的品質が改善されることが確認できる．五行目は提案手法の SSL における重み付き二値交差エントロピーの代わりに，Probabilistic Chamfer Distance [36] を用いてモデルを学習させる場合の結果を示している．Frame-1 から Frame-3 にわたって性能が低下しており，提案手法の SSL の有効性が確認できる．Probabilistic Chamfer Distance では測定された距離に予測値が乗算されるため，点と点の距離を正確に測定することが困難になり，結果として補間性能に悪影響を与えると考えられる．対照的に，提案手法の SSL における重み付き二値交差エントロピーは距離を含まないため，補間性能に悪影響を与えない．

4.4 点群サンプリング結果の分析

提案手法の時間的サンプリングによる，二つの中間フレーム $\hat{\mathbf{P}}_{0\to t}$ および $\hat{\mathbf{P}}_{1\to t}$ からの点群サンプリングの結果を分析する．表 4 に KITTI Odometry，Argoverse 2 Sensor，nuScenes データセットにおけるサンプリング率および二つの中間フレームの補間性能の平均値を示す．

表 3. 提案手法の構成要素の影響．補間時間重み付け (IWT)，パッチごとのサンプリング (PWS)，注意に基づく座標補正 (ACR)，自己教師あり学習 (SSL) を評価する．CD と EMD は数値が低いほど良い結果であることを表す．

Indices	ITW	PWS	ACR	SSL	Frame-1		Frame-2		Frame-3		Average	
					CD ↓	EMD ↓	CD ↓	EMD ↓	CD ↓	EMD ↓	CD ↓	EMD ↓
1	✓	✓	✓	✓	**0.49**	**47.74**	**0.50**	**46.52**	**0.47**	**47.79**	**0.49**	**47.35**
2	✗	✓	✓	✓	0.52	49.76	0.51	47.92	0.51	50.03	0.51	49.24
3	✓	✗	✓	✓	0.50	54.09	0.60	79.17	0.49	50.96	0.53	61.41
4	✓	✓	✗	✓	0.58	49.18	0.61	48.66	0.56	50.65	0.58	49.50
5	✓	✓	✓	✗	0.59	53.07	0.55	50.64	0.58	52.46	0.57	52.06

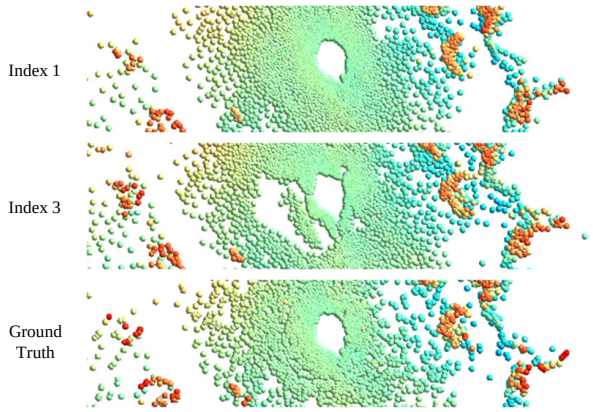

図 6. 提案手法から PWS を除外した場合の点群フレーム補間結果の可視化．点の色は鉛直方向の高さを表す．

Frame-1 に対しては $\hat{\mathbf{P}}_{1\to t}$ からのサンプリング率に比べて $\hat{\mathbf{P}}_{0\to t}$ からのサンプリング率の方が高く，Frame-3 に対してはその逆の傾向が見られる．これは主に提案手法における補間時間重み付けの効果である．ただし，Frame-2 では補間時間重み $W(t)$ と $W(1-t)$ が同値であるにもかかわらず，$\hat{\mathbf{P}}_{0\to t}$ からのサンプリング率の方が高い．これは Frame-2 では $\hat{\mathbf{P}}_{1\to t}$ に比べて $\hat{\mathbf{P}}_{0\to t}$ の方が CD が低い，すなわち最近傍距離の平均が短いためである．$\hat{\mathbf{P}}_{0\to t}$ に比べて $\hat{\mathbf{P}}_{1\to t}$ の方が EMD は低いが，提案手法における動的ラベル割り当てでは最近傍探索に基づいてラベルを割り当てるため，EMD よりも CD に類似した設計である．そのため，提案手法では CD が低いフレームの点に対してより高い重要性を与え，結果としてサンプリング率が高くなると考えられる．Frame-1 と Frame-3 においてもサンプリング率は対称ではなく，$\hat{\mathbf{P}}_{1\to t}$ に比べて CD が低い $\hat{\mathbf{P}}_{0\to t}$ からのサンプリング率の方が高い．これらの結果は提案手法が点群フレーム補間に対する点の重要性を学習可能であることを示している．

5 まとめ

本稿では，双方向シーンフロー推定と時間的サンプリングに基づく点群フレーム補間手法を提案した．提案手法では時間的に連続する二つの点群フレームから双方向シーンフロー推定によって個別に中間フレームを構築し，補間時間を考慮した時間的サンプリングによってそれらを単一のフレームに統合する．その後，提案手法は注意に基づく座標補正を実行し，統合されたフレームの幾何学的品質を改善する．さらに，モデルの学習時にサンプリングのための真値ラベルを作成する自己教師あり学習手法も導入した．大規模な動的点群シーケンスデータセットを用いた評価実験により，提案手法が従来手法を上回る点群フレーム補間性能を達成することを示した．今後の課題として，より空間的解像度の高い点群や，属性情報を持つ点群に対する点群フレーム補間の適用が挙げられる．

表 4. 二つの中間フレーム $\hat{\mathbf{P}}_{0\to t}$ および $\hat{\mathbf{P}}_{1\to t}$ からのサンプリング率と点群フレーム補間性能．

Metrics	Frame-1	Frame-2	Frame-3
Sampling rates from $\hat{\mathbf{P}}_{0\to t}$	82.4%	54.6%	19.4%
Sampling rates from $\hat{\mathbf{P}}_{1\to t}$	17.6%	45.4%	80.6%
CD ↓ between $\mathbf{P}_t$ and $\hat{\mathbf{P}}_{0\to t}$	0.73	0.98	1.27
EMD ↓ between $\mathbf{P}_t$ and $\hat{\mathbf{P}}_{0\to t}$	88.09	115.45	142.35
CD ↓ between $\mathbf{P}_t$ and $\hat{\mathbf{P}}_{1\to t}$	1.57	1.17	0.77
EMD ↓ between $\mathbf{P}_t$ and $\hat{\mathbf{P}}_{1\to t}$	129.89	106.40	85.35

謝辞

本研究成果は，国立研究開発法人情報通信研究機構（NICT）の委託研究（JPJ012368C06801）により得られたものです．

参考文献

[1] Y. Lu, J. Nie, Z. He, H. Gu, and X. Lv, “VoxelTrack: Exploring multi-level voxel representation for 3D point cloud object tracking,” Proceedings of the 32nd ACM International Conference on Multimedia (MM), pp.6345–6354, 2024.

[2] S. Li, S. Wang, Y. Zhou, Z. Shen, and X. Li, “Tightly coupled integration of GNSS, INS, and LiDAR for vehicle navigation in urban environments,” IEEE Internet of Things Journal (IoT-J), vol.9, no.24, pp.24721–24735, 2022.

[3] Z. Liu, R.A. Yeh, X. Tang, Y. Liu, and A. Agarwala, “Video frame synthesis using deep voxel flow,” Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp.4463–4471, 2017.

[4] H. Jiang, D. Sun, V. Jampani, M.H. Yang, E. Learned-Miller, and J. Kautz, “Super SloMo: High quality estimation of multiple intermediate frames for video interpolation,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.9000–9008, 2018.

[5] T. Peleg, P. Szekely, D. Sabo, and O. Sendik, “IM-Net for high res-

olution video frame interpolation," Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition (CVPR), pp.2398–2407, 2019.
[6] W. Bao, W.S. Lai, C. Ma, X. Zhang, Z. Gao, and M.H. Yang, "Depth-aware video frame interpolation," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.3703–3712, 2019.
[7] H. Liu, K. Liao, C. Lin, Y. Zhao, and M. Liu, "PLIN: A network for pseudo-LiDAR point cloud interpolation," Sensors, vol.20, no.6, p.1573, 2020.
[8] H. Liu, K. Liao, C. Lin, Y. Zhao, and Y. Guo, "Pseudo-LiDAR point cloud interpolation based on 3D motion representation and spatial supervision," IEEE Transactions on Intelligent Transportation Systems (T-ITS), vol.23, no.7, pp.6379–6389, 2021.
[9] Z. Zheng, D. Wu, R. Lu, F. Lu, G. Chen, and C. Jiang, "NeuralPCI: Spatio-temporal neural field for 3D point cloud multi-frame nonlinear interpolation," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.909–918, 2023.
[10] F. Lu, G. Chen, S. Qu, Z. Li, Y. Liu, and A. Knoll, "PointINet: Point cloud frame interpolation network," Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), pp.2251–2259, 2021.
[11] T. Zhang, G. Qian, J. Xie, and J. Yang, "FastPCI: Motion-structure guided fast point cloud frame interpolation," Proceedings of the European Conference on Computer Vision (ECCV), pp.251–267, Springer, 2024.
[12] Y. Eldar, M. Lindenbaum, M. Porat, and Y.Y. Zeevi, "The farthest point strategy for progressive image sampling," IEEE Transactions on Image Processing (TIP), vol.6, no.9, pp.1305–1315, 1997.
[13] C. Moenning and N.A. Dodgson, "Fast marching farthest point sampling," tech. rep., University of Cambridge, Computer Laboratory, 2003.
[14] O. Dovrat, I. Lang, and S. Avidan, "Learning to sample," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.2760–2769, 2019.
[15] I. Lang, A. Manor, and S. Avidan, "SampleNet: Differentiable point cloud sampling," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.7578–7588, 2020.
[16] Y. Lin, Y. Huang, S. Zhou, M. Jiang, T. Wang, and Y. Lei, "DA-Net: Density-adaptive downsampling network for point cloud classification via end-to-end learning," Proceedings of the International Conference on Pattern Recognition and Artificial Intelligence (PRAI), pp.13–18, IEEE, 2021.
[17] X. Wang, Y. Jin, Y. Cen, C. Lang, and Y. Li, "PST-NET: Point cloud sampling via point-based transformer," Proceedings of the International Conference on Image and Graphics (ICIG), pp.57–69, Springer, 2021.
[18] Y. Qian, J. Hou, Q. Zhang, Y. Zeng, S. Kwong, and Y. He, "Task-oriented compact representation of 3D point clouds via a matrix optimization-driven network," IEEE Transactions on Circuits and Systems for Video Technology (TCSVT), vol.33, no.11, pp.6981–6995, 2025.
[19] X. Wang, Y. Jin, Y. Cen, T. Wang, B. Tang, and Y. Li, "LighTN: Light-weight transformer network for performance-overhead tradeoff in point cloud downsampling," IEEE Transactions on Multimedia (TMM), vol.27, pp.832–847, 2023.
[20] C. Wu, J. Zheng, J. Pfrommer, and J. Beyerer, "Attention-based point cloud edge sampling," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.5333–5343, 2023.
[21] K. Matsuzaki and K. Nonaka, "Point cloud sampling preserving local geometry for surface reconstruction," Proceedings of the British Machine Vision Conference (BMVC), pp.1–14, 2023.
[22] K. Matsuzaki and K. Nonaka, "Learnable point cloud sampling considering seed point for neural surface reconstruction," IEEE Access, vol.12, pp.190945–190958, 2024.
[23] J. Liu, J. Li, K. Wang, H. Guo, J. Yang, J. Peng, K. Xu, X. Liu, and J. Guo, "LTA-PCS: Learnable task-agnostic point cloud sampling," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.28035–28045, 2024.
[24] X. Wu, L. Jiang, P.S. Wang, Z. Liu, X. Liu, Y. Qiao, W. Ouyang, T. He, and H. Zhao, "Point transformer v3: Simpler, faster, stronger," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.4840–4851, 2024.
[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," Proceedings of the Advances in Neural Information Processing Systems (NIPS), 2017.
[26] V. Nair and G.E. Hinton, "Rectified linear units improve restricted Boltzmann machines," Proceedings of the International Conference on Machine Learning (ICML), pp.807–814, 2010.
[27] H. Fan, H. Su, and L. Guibas, "A point set generation network for 3D object reconstruction from a single image," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.605–613, 2017.
[28] W. Wu, Z.Y. Wang, Z. Li, W. Liu, and L. Fuxin, "PointPWC-Net: Cost volume on point clouds for (self-) supervised scene flow estimation," Proceedings of the European Conference on Computer Vision (ECCV), pp.88–107, Springer, 2020.
[29] X. Li, J. Kaesemodel Pontes, and S. Lucey, "Neural scene flow prior," Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), pp.7838–7851, 2021.
[30] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.3354–3361, IEEE, 2012.
[31] M.F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, and J. Hays, "Argoverse: 3D tracking and forecasting with rich maps," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.8748–8757, 2019.
[32] H. Caesar, V. Bankiti, A.H. Lang, S. Vora, V.E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A multimodal dataset for autonomous driving," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.11621–11631, 2020.
[33] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), 2019.
[34] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," Proceedings of the International Conference on Learning Representations (ICLR), 2019.
[35] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," Proceedings of the International Conference on Learning Representations (ICLR), 2017.
[36] R.A. Potamias, S. Ploumpis, and S. Zafeiriou, "Neural mesh simplification," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.18583–18592, 2022.