論文

PointpartNet++：対応点の決定によるサイズが異なる3次元点群位置合わせの精度向上*

顔　世荀** Sarthak PATHAK† 梅田和昇†

PointpartNet++: Accuracy Improvement of 3D Point-Cloud Registration via Determination of Corresponding Points

Shixun YAN, Sarthak PATHAK and Kazunori UMEDA

This paper improves a deep learning model for point cloud registration of different sizes. Point cloud registration is an important task for applications such as 3D modeling, self-positioning of mobile robots, and environment inspection. Recently, deep learning has started to deal with point clouds. PointNet was the first deep learning model for point cloud classification and semantic segmentation. Since then, methods based on PointNet for tasks such as point clouds registration have also been proposed. However, these methods are only suitable for identical or nearly identical point clouds. The size of the point cloud varies with measurement distance, sensor type, environment, and many other factors. Therefore, point clouds that need to be registered may have very different sizes or dimensions. Conventional methods cannot cope with such situations. In the past, we proposed "PointpartNet", a new deep neural network for point cloud registration based on partial feature extraction. This network specialized in searching the matching region between point clouds of different sizes, followed by a simple registration method. In this paper, we improve this model to increase the accuracy and robustness of point clouds registration. This is done by extracting local point cloud features and passing them through an explicit feature matching network that calculates correspondences between the point clouds to be registered. In qualitative experiments, we demonstrate that this vastly decreases the registration error and increases the success rate of point cloud registration by as much as 15 to 16 percent, as compared to previous research.

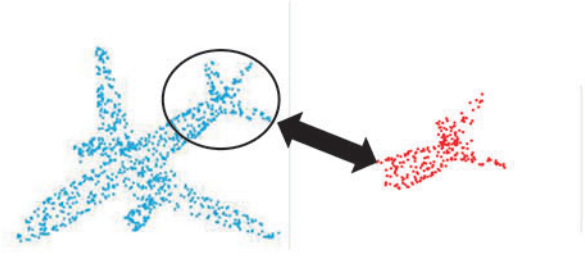
Key words: point cloud; deep learning; registration

1. 緒　　言

距離画像センサは，3次元計測，SLAM（Simultaneous Localization and Mapping），点検，環境認識などのタスクに有用である．しかし，どの種類の距離画像センサも，距離，精度，測定条件などにそれぞれ制限がある．そこで，ロバスト性を向上させるために，複数の距離画像センサを融合させることが有効である[1)2)]．このとき，異なる距離画像センサから得られた点群をロバストかつ高精度に位置合わせをすることは重要である．本研究は特に異なるサイズの点群の位置合わせに注目する．ここで本研究における点群のサイズとは，点群が取得されている範囲の広さを表している．距離画像センサによって計測できる画角や計測レンジが異なり，また計測時の対象までの距離も都度変化することから，取得できる点群のサイズが異なる．例えば，**図1**の(a)の点群は(b)と比べサイズが大きいことがわかる．

近年，深層学習技術は様々な領域で優れた成果を上げている．PointNet[3)]は，点群を直接扱う最初の分類用深層学習モデルである．これは，入力点群の点数や順序に依存しないモデル構造を持っている．PointNet[3)]の後，それに基づく他のモデルがいくつか提案された[4)-8)]．

その中で，PointNet[3)]をベースとした位置合わせモデルも多く提案されている[9)10)]．その代表的なモデルの一つがPointNetLK[9)]である．これは，点群それぞれからPointNet[3)]により求めた大域特徴から点群間の剛体変換行列を求めるモデルである．このモデルでは，点群にノイズが多い場合や，点群のサイズが異なる場合，位置の精度が低下する傾向にある．この問題を解決するために，MaskNet[11)]と呼ばれるモデルが提案されている．これは，点群間のマッチング領域を探索するモデルである．しかし，このモデルは，点群のサイズの差が大きすぎると，ロバスト性や精度に欠ける．そこで我々はサイズの大きい点群を切り分けし，各サブ点群の大域特徴を抽出し，各サブ点群とサイズの小さい点群のマッチング尤度スコアを算出するニューラルネットワークPointpartNet[12)]を提案した．このスコアは，サイズの小さい点



(a) point cloud with large size　　(b) point cloud with small size

Fig. 1　Point clouds with different sizes

* 原稿受付 令和4年5月13日
掲載決定 令和4年9月14日
** 学生会員 中央大学大学院（東京都文京区春日1-13-27）
† 正会員 中央大学（東京都文京区春日1-13-27）

群がサイズの大きい点群のどの箇所に類似するかを予測する．この結果をもとに，別のニューラルネットワークを用いて，実際の位置合わせを行う．また，ローカルミニマムに陥らないように，ラフ位置合わせと精密位置合わせに分割して位置合わせを行っている．このモデルでは精密位置合わせの際，特徴空間で点群同士の各点の対応関係を求めている．マッチング領域探索時には，最近傍点で点群切り分けを行った上で，切り分けられた点群を用いて位置合わせを行っている．例として，ある距離画像センサにより取得した点群と，別の距離画像センサにより取得した点群から最近傍点での切り分けで得られた点群を**図 2** に示す．このように，ある距離画像センサにより取得した点群の形状と別の距離画像センサにより取得した点群から最近傍点での切り分けで得られた点群の形状が異なるため，1 対 1 の点の対応関係を求めることが困難である．従って，本研究では PointpartNet [12]に点の対応関係を求める深層学習モデルを導入することで，サイズの異なる点群を対象とした位置合わせ精度を向上することを目的とする．

2. 関 連 研 究

これまでにも多くの点群位置合わせ手法が提案されている [13)-15)]．ICP（Iterative Closest Point）[13)]は，反復的な点群位置合わせ手法である．この手法は，簡便なことから広く利用されている一方で，外れ値がある場合に精度が低下し，また位置合わせの成否が点群の初期の位置と姿勢に強く依存する．正規分布変換（NDT）[14)]は，点群空間をボクセルとグリッドで等間隔に分割し，それぞれについて求めた正規分布を用いて点群を表現し照合する手法である．計算コストが小さく，また，NDT マッピングは，サイズが大きく異なる点群間の位置合わせにも使用することができる．しかし，NDT マッピングは，上から見ると正確であるが，横から見ると高さ方向に持続的な変形がある [16)]．また，近年，深層学習の発展に伴い，以下に示すように点群ベースの深層学習モデルも多く提案されている．

・深層学習を用いた点群処理：PointNet [3)]は，点群の分類とセマンティックセグメンテーションのための最初の深層学習モデルである．PointCNN [4)]は，点群中の点の並び順を考慮した分類とセマンティックセグメンテーションのための深層学習モデルである．X-Conv [4)]は，点群から代表点を選択し，隣接する点の情報を畳み込む手法である．このモデルは PointNet [3)]よりもロバストである．

・位置合わせ：PointNet [3)]に基づく手法として，DirectNet [17)]や PointNetLK [9)]が提案された．DirectNet [17)]は，ニューラルネットワークで抽出した点群特徴を MLP（Multilayer Perceptron）の入力として扱い，剛体変換を算出するモデルである．しかし，このモデルはラフ位置合わせしか行うことができない．PointNetLK [9)]は，PointNet [3)]を用いて点群大域特徴を抽出し，対象となる点群の特徴が互いに同じになるように剛体変換行列を算出するモデルである．しかし，この方法は，二つの点群がほぼ同一であることが前提となる．点群のサイズが大きく異なる場合（例えば，一方の点群が他方の点群の一部である場合）には，位置合わせ精度が低下する．

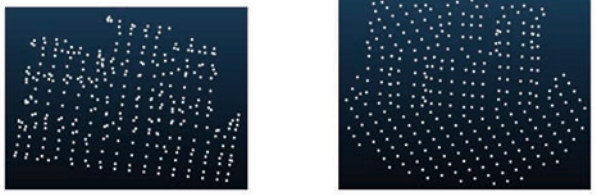
・マッチング領域探索：点群同士のマッチング領域を算出するために，MaskNet [11)]というモデルが提案された．このモデルでは，点群全体のみを考慮し，部分領域を見ない．そのため，MaskNet [11)]も，サイズが大きく異なる点群や，姿勢が大きく異なる点群には対応できず，マッチングに失敗することが多い．

・PointpartNet [12)]：PointpartNet [12)]は，最近傍点で点群を切り分けし，部分的特徴を用いてマッチング領域探索を行い，その上で点群位置合わせを行うモデルである．このモデルは，サイズの異なる点群を対象としたマッチング領域探索に特化しているため，点群位置合わせに関しては簡単な対応点探索を行った後に SVD（Singular Value Decomposition）で剛体変換行列を計算している．しかし，SVD を用いて剛体変換行列を算出する場合，対応関係が正確に得られていることが必要である．また，点群を最近傍点で探索し切り分ける処理を行うため，図 2 のように切り分けられた点群と実際の距離画像センサにより取得した点群とに差異が生じる．そのため，マッチングできない点を排除せず，マッチングした点群間で対応関係を求めると，誤対応が発生してしまう．それを用いて SVD で剛体変換行列を計算すると誤差が発生し，PointpartNet [12)]の位置合わせ精度が低下する原因となる．

これらを踏まえて，本研究は，PointpartNet [12)]に点の対応関係を求める深層学習モデルを導入することで，より正確な対応点を求め，サイズの異なる点群に対しより高精度な位置合わせが可能な深層学習モデル PointpartNet++を提案する．

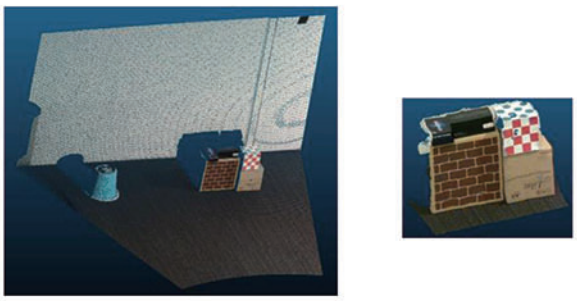
3. PointpartNet++

本研究では，位置合わせする 2 つの点群について，以下の条件を想定する：1. LiDAR のようなワイドレンジスキャナで取得した広範囲の点群，2. RGB-D カメラなどで撮影した狭い範囲の点群．点群のサンプルを**図 3** に示す．ここで，サイズの大きい点群を **full**，サイズの小さい点群を **part** と呼ぶ．PointpartNet++の基本コンセプトは，マッチングしていると判断された点群同士の対応点関係を深層学習モデルで算出することで，高精度な剛体変換行列を計算することである．PointpartNet++の概要は以下の通りに，2 つのステップで構成されている：1. 部分点群の



(a) point cloud of real data　　(b) partitioned point cloud

Fig. 2 Differences between raw point cloud and partitioned point cloud captured by another range image sensor



(a) **full**　　(b) **part**

Fig. 3 A sample of point clouds that we aim to register in this research

特徴抽出とマッチング領域探索，2. ステップ 1 で得られたマッチング点群からの対応点探索と位置合わせ．PointpartNet++の構成を**図 4** に示す．部分点群の特徴抽出，マッチング領域探索及びラフ位置合わせに関しては，PointpartNet [12] は良好な結果を得られたため，同じ手法を用いる．一方，本手法では位置合わせの精度向上のために，精密位置合わせにおいて点の対応関係を求める深層学習モデルを導入する． PointpartNet [12] の精密位置合わせは，3 次元点の対応関係に誤りが多い状態から SVD を用いて剛体変換行列を計算する．これに対し，本モデルでは，深層学習モデルで対応点を求めることで，より正しい 3 次元点の対応関係の下で SVD を用いて剛体変換行列を計算することが可能となり，より高精度でロバストな位置合わせが実現できることが期待できる．

以下，PointpartNet++の概要を説明する．

(1)部分点群の特徴抽出とマッチング領域探索：**part** と一致する **full** の領域を見つけるために，**full** を分割し，部分特徴を抽出する．マッチング領域探索では，**part** とのマッチング尤度が最大となる **full** の領域を求めて，ラフ位置合わせの並進 $\mathbf{t}_G$ を得る．

(2)ラフ位置合わせ：深層学習モデルによるラフ位置合わせを行い，ラフ位置合わせの回転 $\mathbf{R}_G$ を算出する．

(3)精密位置合わせ（提案手法）：深層学習モデルを用いて点群同士の対応関係を求める．得られた対応関係から SVD を用いて精密位置合わせを行い，精密位置合わせの回転 $\mathbf{R}_L$ と精密位置合わせの並進 $\mathbf{t}_L$ を計算する．

3.1 部分的特徴抽出とマッチング領域探索

まず，**full** と **part** の点群の密度が同等になるようにダウンサンプリングする．本研究ではボクセルグリッドフィルタ [18] を用いてダウンサンプリングを行う．ダウンサンプリング後の点群を $\mathbf{full} = \{\mathbf{f}_1, ..., \mathbf{f}_g\}$, $\mathbf{part} = \{\mathbf{p}_1, ..., \mathbf{p}_n\}$とする．点 $\mathbf{f}_i = (x_i, y_i, z_i)$, $\mathbf{p}_j = (x_j, y_j, z_j)$はそれぞれ **full**, **part** に含まれる 3 次元点，g, n は **full**, **part** に含まれる 3 次元点の数である．

ダウンサンプリング後，**full** の各部分点群の特徴を抽出するため **full** の切り分けを行う． **full** の各 3 次元点に対して，n 個の最近傍点を選択し，g 個のサブ点群の集合 $\mathbf{full}_{group} = \{\mathbf{full}_{group\,i} \mid i = 1, ... g\}$を作成する． $\mathbf{full}_{group\,i}$ は **full** から切り分けられたサブ点群である．ここで，**full** と **part** はすでにダウンサンプリングし，同じ密度になっている．そのため，**part** と同じ点数で **full** を切り分けることで **part** と似た領域の切り分けが可能となる．

切り分け処理の後，PointNet [3]を用いて $\mathbf{full}_{group}$ と **part** の大域特徴を抽出する．$\mathbf{full}_{group}$ から得られた特徴を $\mathbf{Feature_f} = \{\mathbf{F}_{fi} \mid \mathbf{F}_{fi} \in \mathbb{R}^d,\ i = 1, ... g\}$，**part** から得られた特徴を $\mathbf{Feature_p} = \{\mathbf{F_p} \mid \mathbf{F_p} \in \mathbb{R}^d\}$とする．d は PointNet [3]から出力された特徴の次元数である．PointNet [3]に入力する際には，位置に依存しない特徴を得るため，入力点群の重心位置が原点に一致するように並進移動させる．さらに，スケールに対するロバスト性を高めるために，**full** に応じたユニットボックス [9]への正規化も行う．

特徴抽出後，**part** に最も類似している **full** の領域を探索する．また，この処理では，ラフ位置合わせの並進 $\mathbf{t}_G$ も計算する．$\mathbf{Feature_p}$ と $\mathbf{Feature_f}$ を MLP_1 に入力し，マッチング尤度 $\mathbf{Score} = \{Score_i \mid i = 1, ... g\}$を計算する．ただし，$MLP_1$ の層数は 4，各層のサイズは (256, 128, 128, 1)とした．$Score_i$ は，0 から 1 の値をとり，$\mathbf{full}_{group\,i}$ が **part** と類似しているほど 1 に近づく．最後に，最も高いスコア $Score_H$ をとる $\mathbf{full}_{group\,H}$ を **part** にマッチングする領域と見なす．また，このときの $\mathbf{full}_{group\,H}$ と **part** の重心位置の差をラフ位置合わせの並進 $\mathbf{t}_G$ とする．

3.2 点群位置合わせ

推定した領域 $\mathbf{full}_{group\,H}$ に対し，点群間の位置合わせを行う．位置合わせは，ラフ位置合わせと精密位置合わせの 2 つに分けられる．ラフ位置合わせは，高速で大まかな点群位置合わせを行う処理である．精密位置合わせは，高精度に点群位置合わせを行う処理である．ラフ位置合わせ，精密位置合わせにより得られた剛体変換行列を用いて点群を式（1）に従ってに座標変換する．

$$\mathbf{part'}_i = \mathbf{R} \cdot \mathbf{part}_i + \mathbf{t} \qquad i = 1, 2, ..., n \tag{1}$$

ここで，$\mathbf{R}$ は回転行列，$\mathbf{t}$ は並進ベクトルである．また，$\mathbf{part}_i$ は **part** の 3 次元点，$\mathbf{part'}_i$ は座標変換後の **part** の 3 次元点である．

3. 2. 1 ラフ位置合わせ

マッチング領域検索では，$\mathbf{full}_{group\,H}$ と **part** のラフ位置合わせの並進 $\mathbf{t}_G$ を推定した．ラフ位置合わせでは，$\mathbf{full}_{group\,H}$ と **part** の大まかな姿勢関係であるラフ位置合わせの回転 $\mathbf{R}_G$ を求める．この処理では，高速に大まかな姿勢を求めることとなる．ラフ位置合わせの回転を求めるには図 4 に示している DirectNet [17]を用いる．$\mathbf{full}_{group\,H}$ と **part** の大域特徴を入力とし，MLP_2 によりオイラー角を計算する． MLP_2 により求めたオイラー角を用いてラフ位置合わせの回転 $\mathbf{R}_G$ を表す．ただし，MLP_2 の層数は 5，各層のサイズは（512, 256, 128, 64, 3）とした．

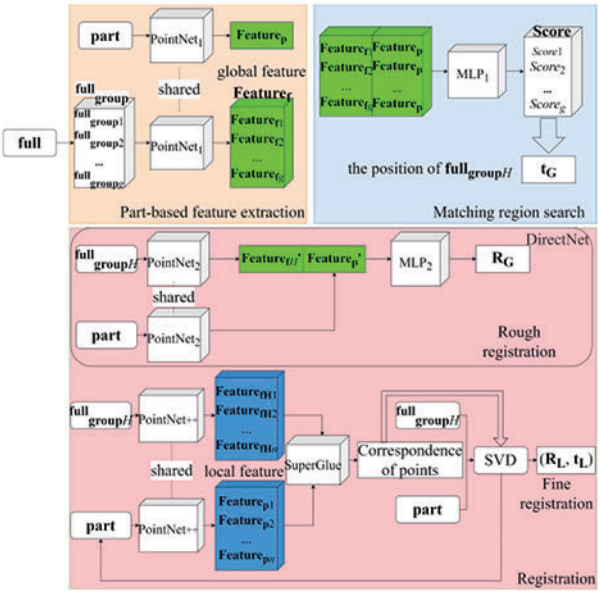


Fig. 4 PointpartNet++ architecture

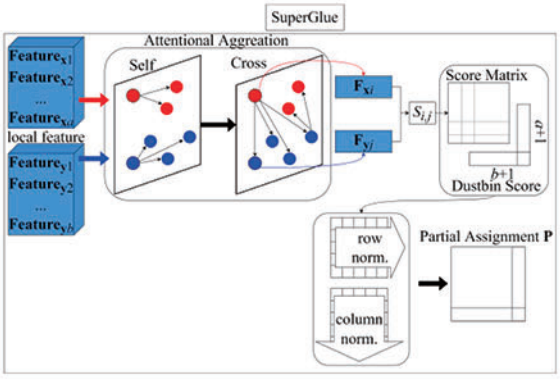


Fig. 5 SuperGlue [19] architecture

3. 2. 2　3 次元点の対応関係を用いた精密位置合わせ

前項までで，ラフ位置合わせの並進 $\mathbf{t}_G$，ラフ位置合わせの回転 $\mathbf{R}_G$ により $\mathbf{full}_{groupH}$ と **part** の位置姿勢の関係を大まかに推定した．この項では，$\mathbf{full}_{groupH}$ と **part** の位置合わせ精度を向上させる処理を行う．PointpartNet [12]では，$\mathbf{full}_{groupH}$ と **part** の点の対応関係を特徴空間で最も類似しているものから求めた．しかし，$\mathbf{full}_{groupH}$ は最近傍点で切り分けた点群であるため，**part** と差異が生じる．そのため，単純に点の特徴のみで対応関係を求めると，SVD により剛体変換行列を計算する場合に精度が低下する．従って，本モデルでは，$\mathbf{full}_{groupH}$ と **part** の点の対応関係を求める際に対応点が存在するか否かを判断して不要な点を排除した上で，SVD により剛体変換行列を算出する．これにより，より正確な剛体変換行列を得られると考えられる．この処理を行うために，本研究では，SuperGlue [19]を用いる．SuperGlue [19] のイメージ図を**図 5** に示す．SuperGlue [19]とは本来画像中の特徴点の特徴を入力とし，Attention を用いて 2 枚の画像の特徴点のマッチング確率を含む Soft Assignment 行列 $\mathbf{P} \in \mathbb{R}^{(a+1)\times(b+1)}$ を出力するモデルである．a と b はそれぞれの画像の特徴点の点数である．SuperGlue [19]は，点の対応関係を算出するだけでなく，そもそもマッチングできる点が存在するかどうかの判断も行っている．存在する場合，$P_{u,v}$ の値が 1 に近づく．u と v は対応する点のインデックスである．一方，存在しない場合には，$P_{x,b+1}$ あるいは $P_{a+1,y}$ の値が 1 に近づく．x と y は対応点が存在しない点のインデックスである．本研究では，画像特徴点の特徴を 3 次元点の特徴に変更した上で SuperGlue [19]を用いることで，点群同士の局所特徴から対応点を算出する．まず，図 4 に示すように PointNet++ [5]から得られた $\mathbf{full}_{groupH}$ と **part** の局所特徴を抽出する．局所特徴とは，点群に含まれる各点の特徴である．このとき，PointNet++ [5] は 256 次元の特徴を出力する．抽出された局所特徴を SuperGlue [19]に入力し，$\mathbf{full}_{groupH}$ と **part** の点の対応関係を求める．そして，算出された点の対応関係と 3 次元点の座標を SVD に入力し剛体変換行列を算出する．また，求めた剛体変換行列を適用し，再度上記の処理を行い新たな剛体変換行列を得る．これを剛体変換が収束するまで繰り返す．

3.3　学　　習

本手法では，マッチング領域探索，ラフ位置合わせ，精密位置合わせについてそれぞれ教師あり学習を行う．学習にはデータセットを用いる．このとき，**full** には全体の点群，**part** には **full** を最近傍点で切り分けた点群を用いる．また，学習処理の効率化のため，ラフ位置合わせの学習時には，$\mathbf{full}_{groupH}$ ではなく，マッチング領域探索の真値に相当する $\mathbf{full}_{groupgt}$ を入力としている．データセットから得られた **part** は $\mathbf{full}_{groupgt}$ と完全に一致する点群となる．しかし，このまま精密位置合わせの学習を行うと，マッチングできない点を排除する学習ができない．従って，精密位置合わせの学習時には，$\mathbf{full}_{groupgt}$ ではなく $\mathbf{full}_{groupgt}$'を用いる．$\mathbf{full}_{groupgt}$'はマッチング領域探索の真値である $\mathbf{full}_{groupgt}$ の最近傍点の点数を n から $1.2n$ に拡大してえられる点群である．この学習に関しては，はじめにマッチング領域探索とラフ位置合わせを学習率 0.001 で行い，次に，精密位置合わせを学習率 0.0001 で行う．

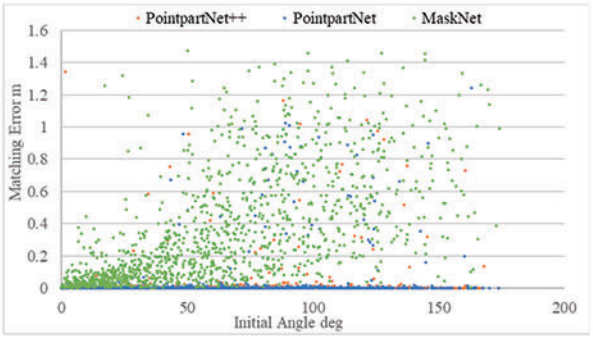
3.4　損失関数

本モデルは，以下の 3 つのネットワークを学習させる：1. 最も高いスコアを選択するマッチング領域探索，2. ラフ位置合わせ，3. 精密位置合わせの対応関係．マッチング領域探索の損失関数をマッチング損失 $loss_m$，ラフ位置合わせの損失関数をラフ位置合わせ損失 $loss_r$，精密位置合わせの損失関数を精密位置合わせ損失 $loss_{SG}$ と定義する．マッチング領域探索及びラフ位置合

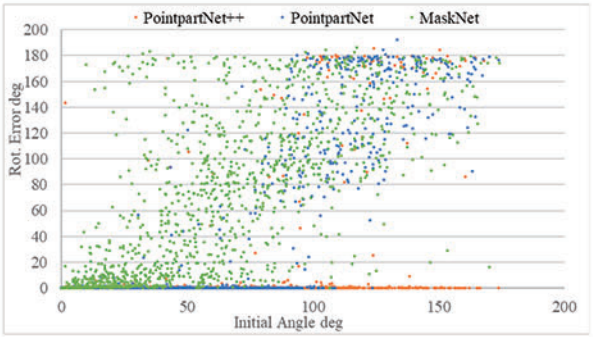
Fig. 6　Point cloud samples of the dataset: blue: **full**, red: **part**

Table 1　Experimental conditions

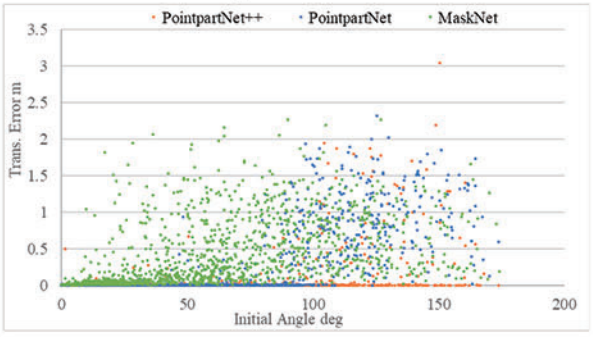
	batch size	number of points in **full**	number of points in **part**
training	16	256	64
testing	1	1024	256



(a) Matching error



(b) Rotation error



(c) Translation error

Fig. 7　Registration results (scatter plot)

わせの学習では，$loss_m$ と $loss_r$ の和を最小にするように学習する．精密位置合わせの学習では，$loss_{SG}$ を最小にするように学習する．

3. 4. 1　マッチング損失とラフ位置合わせ損失

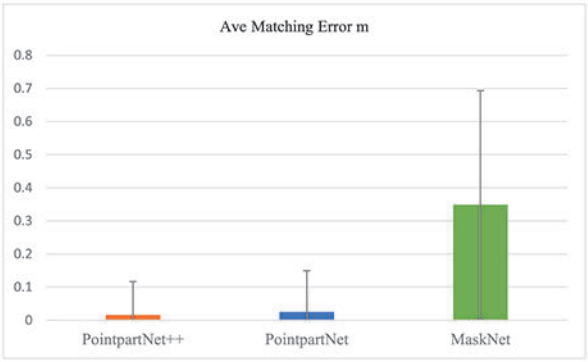
マッチング損失 $loss_m$ は分類問題でよく使われる負の対数尤度損失で与える．損失関数 $loss_m$ の式を式（2）に示す．

$$loss_m = -\log\left(\frac{\exp(Score_{label})}{\sum_{i=1}^{g}\exp(Score_i)}\right) \tag{2}$$

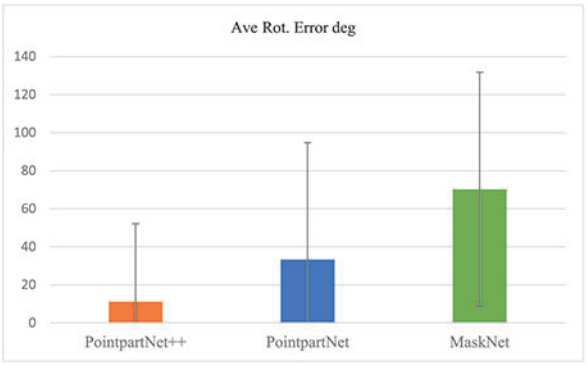
ここで，$label$ はマッチング領域探索の真値のインデックス，$Score_{label}$ は真値で算出されたマッチング尤度スコアである．一方，ラフ位置合わせ損失 $loss_r$ は剛体変換行列 $\mathbf{G_G}$ と剛体変換行列の真値 $\mathbf{G_{gt}}$ との差を最小にする損失関数である．$\mathbf{G_G}$ と算出されたラフ位置合わせの回転 $\mathbf{R_G}$ とラフ位置合わせの並進 $\mathbf{t_G}$ の関係を式（3）に示す．

$$\mathbf{G_G} = \begin{pmatrix} \mathbf{R_G} & \mathbf{t_G} \\ 0 & 1 \end{pmatrix} \tag{3}$$

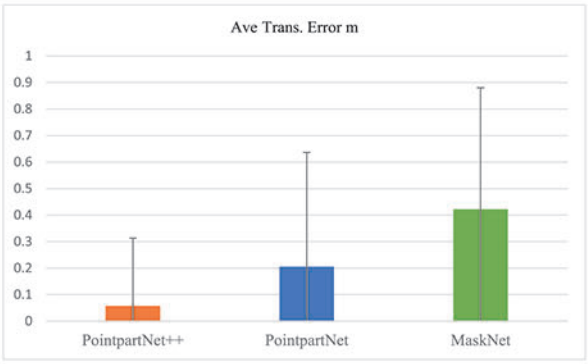
MSE（Mean Square Error）を用いて以下のように表現する．



(a) Matching error



(b) Rotation error



(c) Translation error

Fig. 8　Mean errors and standard deviations on ModelNet40 [20]

$$loss_r = \left\|(\mathbf{G_G})^{-1}\cdot\mathbf{G_{gt}} - \mathbf{I_4}\right\|_F \tag{4}$$

3. 4. 2　精密位置合わせ損失

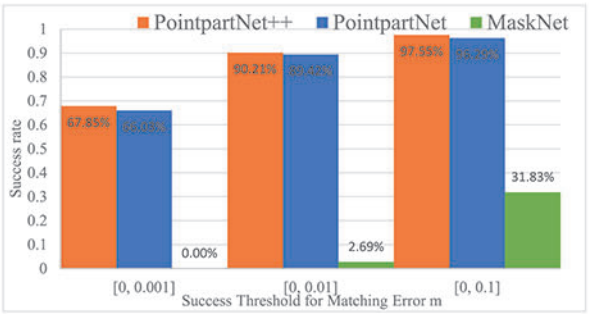
精密位置合わせ損失 $loss_{SG}$ は精密位置合わせの際の対応点算出の損失である．損失は以下の二つを含む：1.対応点が存在する際の対応関係の損失，2.対応点が存在しない際に存在しないと判断する損失．$loss_{SG}$ は以下の式に示すように負の対数尤度損失で与える．

$$loss_{SG} = -\sum_{(k,l)\in\mathbf{M}}\log\mathbf{P}_{k,l} - \sum_{k\in\mathbf{N}}\log\mathbf{P}_{k,n+1} \tag{5}$$

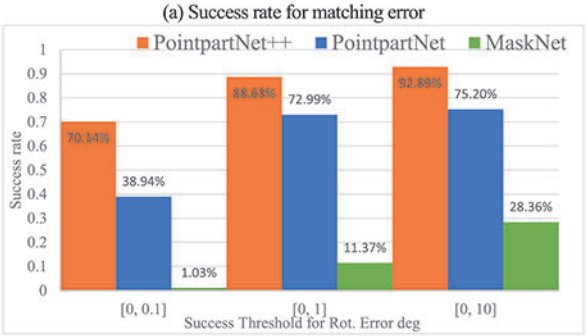
ただし，対応できる点とできない点の確率を同時に学習するため，$\mathbf{full}_{\mathbf{group}gt}$'と **part** を用いる．$\mathbf{full}_{\mathbf{group}gt}$'はマッチング領域探索の真値である $\mathbf{full}_{\mathbf{group}gt}$ の最近傍点の点数を n から $1.2n$ に拡大してえられる点群である．ここで，**M** は $\mathbf{full}_{\mathbf{group}gt}$'と **part** の対応する点のインデックスの集合，**N** は $\mathbf{full}_{\mathbf{group}gt}$'が **part** とマッチングできない点のインデックスの集合である．また，$\mathbf{P}_{k,l}$ と $\mathbf{P}_{k,n+1}$ はそれぞれ $\mathbf{full}_{\mathbf{group}gt}$'と **part** の対応できる組み合わせと対応できない組み合わせから計算された特徴点のマッチング確率である．

4.　実　　験

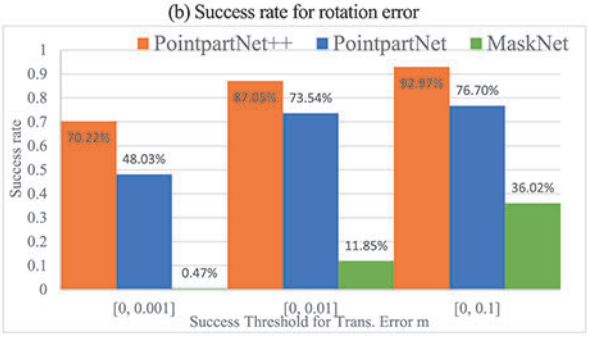
本モデルの有用性を検証するために以下の実験を行った．1. 従来手法の MaskNet [11]，PointpartNet [12]との比較実験，2. 距離画



(a) Success rate for matching error



(b) Success rate for rotation error



(c) Success rate for translation error

Fig. 9　Success rates for each criterion

像センサで取得した点群を用いた精度検証実験．実験の基準として，姿勢推定は回転誤差が 10 °以下，位置推定は並進誤差が 0.1m 以下であれば成功とみなす．また，マッチング領域探索を評価するためのマッチング誤差は，マッチング領域探索の際に推定された $\mathbf{full}_{group/f}$ と真値 $\mathbf{full}_{group/gt}$ との重心の距離である．マッチング誤差は 0.1m 以下であれば成功とみなす．使用したライブラリは python 3.6.10 と pytorch 1.7.0 である．また，使用したパソコンの CPU は 3.60GHz Intel i7-9700K，GPU は GeForce RTX 2070 SUPER である．

4.1 従来手法との比較実験

実験 1 の学習は，40 種類の CAD モデルからなる ModelNet40[20] データセットからリサンプリングで得られた点群を用いて行った．学習されていないオブジェクトの種類に対する本モデルのロバスト性を示すために，学習セットのうち 20 セットでモデルを学習し，テスト時はテストセットの他 20 セットでテストした．**full** は，ModelNet40 [20]のデータからリサンプリングされた点群をそのまま使用した．一方，**part** は，**full** からランダムに 1/4 切り取った点群の各点に平均 0 m，分散 0.05 m^2 の正規分布に従うノイズを加えたものを使用した．点群サンプルを**図 6** に示す．学習に用いた剛体変換行列の真値は，任意に選んだ軸を中心に[0, 90] °の間で回転させ，[0, 1.57] m の間でランダムに並進させることで与えた．未学習位置姿勢に対する本モデルのロバスト性を検証するため，剛体変換行列の真値は，学習時よりも範囲を拡げ，[0, 180] °の間の回転と[0, 3.14] m の間の並進をランダムに生成したものである．また，MaskNet [11]はマッチング領域探索のみなので，位置合わせには従来手法の PointpartNet [12]の位置合わせを用いた．実験条件を**表 1** に示す．学習時のバッチサイズは 16，MaskNet [11]と PointpartNet [12]の学習エポック数は 200 とした．一方，本モデルのエポック数は 400 とした．前半の 200 はマッチング領域探索とラフ位置合わせの学習，後半の 200 は精密位置合わせの学習である．また，学習に使用した点群の点数は，**full** 256 点，**part** 64 点，テスト時は **full** 1024 点，**part** 256 点とした．実験結果を**図 7~9** と**表 2** に示す．また，各プロセスの処理時間を**表 3** に示す．

図 7 は各モデルの実験結果の散布図，図 8 は各モデルの実験結果の平均誤差と標準偏差である．図 9 は各モデルの各判断基準の成功率である．マッチング領域探索と位置推定は 0.1m，0.01m と 0.001m で評価している．姿勢推定は 10 °，1°と 0.1 °で評価している．どの評価基準でも提案モデルである

Table 2 Experimental results for each method

Matching	Success rate	Mean error	Mean error on success
MaskNet [11]	31.83%	0.349 m	0.046 m
PointpartNet [12]	96.29%	0.025 m	0.003 m
PointpartNet++	97.55%	0.016 m	0.003 m
Rotation			
MaskNet [11]	28.36%	70.20 °	3.12 °
PointpartNet [12]	75.20%	33.26 °	0.20 °
PointpartNet++	92.89%	11.09 °	0.21 °
Translation			
MaskNet [11]	36.02%	0.422 m	0.034 m
PointpartNet [12]	76.70%	0.205 m	0.003 m
PointpartNet++	92.97%	0.058 m	0.003 m

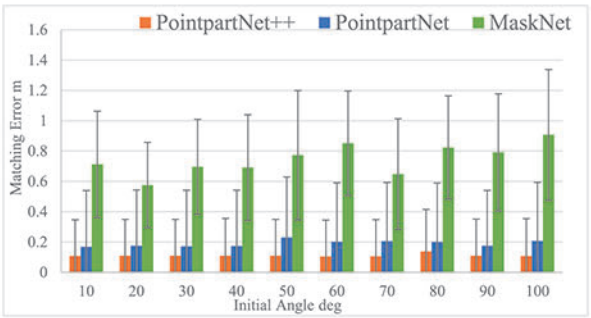
Registration success rate	Rotation	Translation
MaskNet [11]	68.98%	75.68%
PointpartNet [12]	78.10%	79.16%
PointpartNet++	95.06%	95.06%

Table 3 Processing time of each process

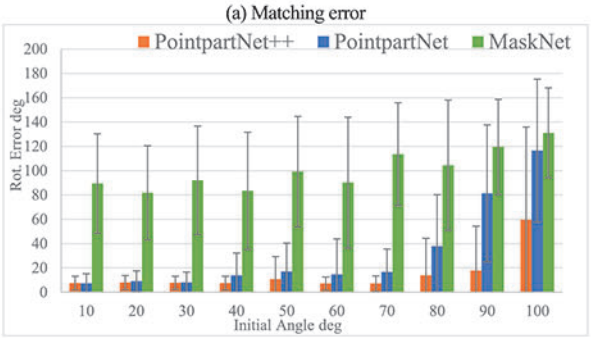
Time s	Matching region search	Global registration	Local registration	Total
PointpartNet [12]	0.089	0.006	0.251	0.345
PointpartNet++	0.086	0.007	1.478	1.574

(a) Before registration　(b) After registration

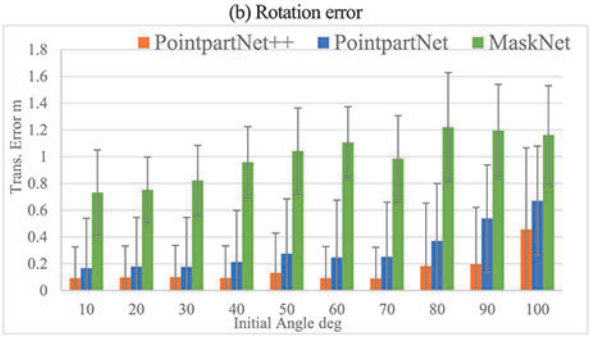
Fig. 10 Registration sample from ModelNet40 [20]. blue: **full**, red: **part**



(a) Matching error



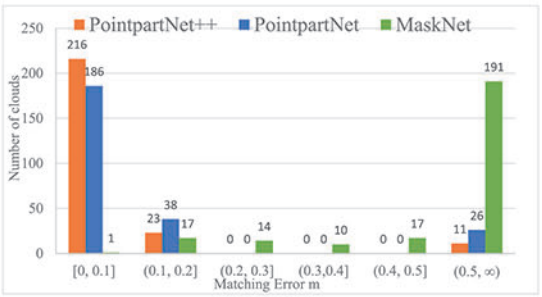
(b) Rotation error



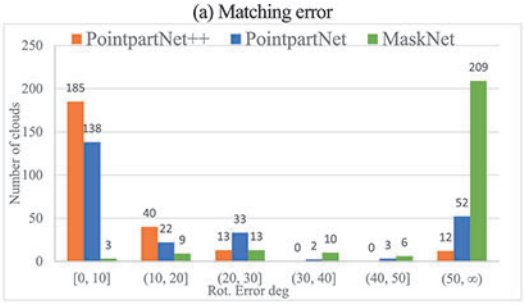
(c) Translation error

Fig. 11 Mean errors and standard deviations on real data

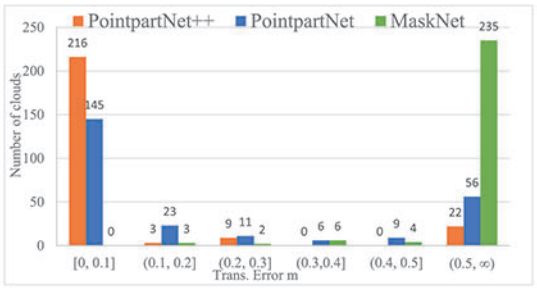
PointpartNet++が最も高く，MaskNet が最も低い成功率となった．実験結果から，MaskNet [11]と比べマッチング領域探索の精度が大幅に安定していることが分かる．また，PointpartNet [12]と比べ修正後の本モデルは異なる姿勢に対し姿勢推定と位置推定がよりロバストであることが示されている．なお，PointpartNet [12]とマッチング領域探索の結果が異なるのは，両方とも end-to-end で学習し，異なる学習パラメータが得られたためである．また，PointpartNet [12]では 1200 個以上のサンプルでマッチング領域探索が成功したなかで，姿勢推定と位置推定は約 950 個のサンプルしか成功していない．それに対し，本モデルでは同様に 1200 個以上のサンプルでマッチング領域探索が成功したなかで，姿勢推定と位置推定はいずれも 1100 個以上のサンプルで成功した．表 2 は各推定の成功率と平均誤差である．まず，全体の姿勢推定と位置推定の成功率は，従来モデルに比べ，本モデルの方が高い．また，姿勢推定と位置推定の平均誤差も大幅に減少した．また，マッチング領域探索が成功した時の姿勢推定と位置推定の成功率も従来モデルが 80%弱に対し，本モデルでは約 95%まで向上した．以上より，本モデルは従来手法と比べより高精度でロバストな手法であることが示されている．また，表 3 は各プロセスの処理時間を示す．本モデルの精密位置合わせの処理時間は従来手法の PointpartNet [12]に比べ 0.251 s から 1.478 s に増えた．これは，SuperGlue [19]の処理に時間がかかったためと考えられる．PointpartNet++を用いた位置合わせのサンプルを**図 10** に示す．正確な位置合わせが実現できていることが分かる．



(a) Matching error



(b) Rotation error

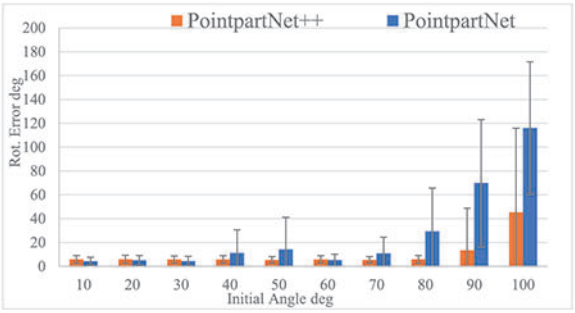


(c) Translation error

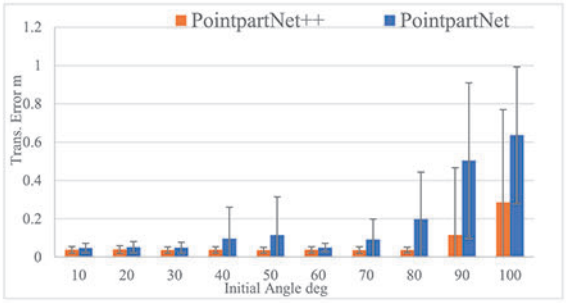
Fig. 12 Registration results on real data (histogram)

4.2 実データを対象とした精度検証実験

距離画像センサにより取得した点群を用いた実データでの精度検証実験を行った．**full** はレーザスキャナ FARO FOCUS^{M}70，**part** はステレオカメラ RealSense D435i から取得した点群を使用した．ダウンサンプリング前の **full** の点数は約 30 万点で，密度は約 5.5 万点/m^2，**part** の点数は約 50 万点で，密度は約 60 万点/m^2 である．ダウンサンプリング後の **full** の点数は約 2500 点，**part** の点数は約 300 点で，密度はいずれも約 400 点/m^2 である．

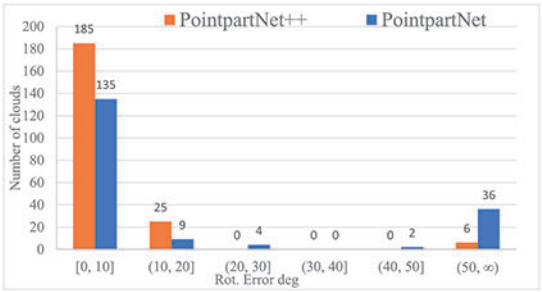


(a) Rotation error

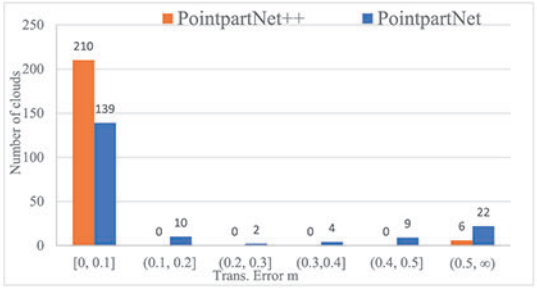


(b) Translation error

Fig. 13 Mean errors and standard deviations on real data when matching region search is successful



(a) Rotation error



(b) Translation error

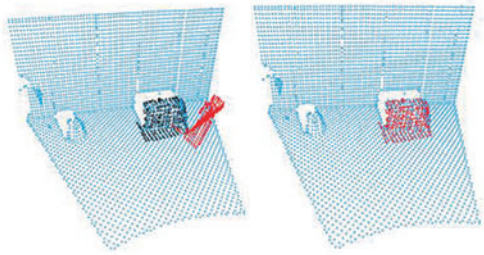
Fig. 14 Registration results on real data when matching region search is successful (histogram)

また，モデルの学習パラメータは実験 1 で学習したものを用いた．実験手順としては，まずそれぞれのセンサにより点群を取得した．取得した点群のサンプルは図 3 に示したものである．次に，取得した **part** の点群を手作業で真の位置姿勢に移動させた．そこから，**part** を 10 °，0.1 m 単位で合計 10 回回転と並進をさせたものを実験対象とした．MaskNet [11]，PointpartNet [12]，PointpartNet++それぞれの実験結果を**図 11~14** と**表 4** に示す．なお，点群は一辺 0.05m のボクセルでダウンサンプリングを行うため，ダウンサンプリング後の点群内の点の間隔は約 0.05m であった．この値を考慮し，位置推定成功の基準を 0.1m にした．また，姿勢推定成功の基準は，図 9 において位置推定と同等の成功率の 10 °にした．

図 11 は実験結果の平均誤差と標準偏差，図 12 は実験結果のヒストグラムである．図 11，12 より，MaskNet [11]はマッチング領域探索がほぼ失敗しているのに対し提案モデルは 8 割以上の成功率が得られたことが示されている．その上，マッチング推定，姿勢推定，位置推定の誤差も大幅に減少している．また，PointpartNet [12]の実験結果と比べ，実データに対する点群マッチング領域探索は似た結果を示し，その成功率は 86.40%であった．また，各姿勢に対し同等の精度が得られていることから，異なる姿勢に対しロバストな手法であることが示されている．さらに，姿勢推定と位置推定を評価するために，PointpartNet [12]と PointpartNet++の実験結果からのマッチング領域探索に成功した結果のみを抽出した．結果を図 13，図 14 と表 4 に示す．図 13 と図 14 はそれぞれマッチング領域探索成功時の平均誤差，標準偏差とヒストグラムである．従来手法と比べ各姿勢に対しよりロバストな結果が得られている．初期姿勢が 0-80 °では安定した結果が得られている一方，初期姿勢が 100 °では推定に失敗することが増えた．これは学習時の初期姿勢が[0, 90] °であるためと考えている．表 4 はマッチング領域探索成功時の推定成功率と平均誤差である．姿勢推定成功率は従来の 72.58%から 85.65%に向上した．姿勢推定の平均誤差は 10.30 °，姿勢推定成功時の平均誤差は 4.89 °である．位置推定成功率は従来の 74.73%から 97.22 %に向上した．位置推定の平均誤差は 0.069 m，位置推定成功時の平均誤差は 0.037 m である．これらの結果から，SuperGlue [19]を用いたことで，PointpartNet [12]と比べ点の対応関係がより正確になり，姿勢と位置推定がより高精度よりロバストであることが示された．PointpartNet++を用いた位置合わせのサンプルを**図 15** に示す．なお，本実験で，モデルの学習パラメータは FARO と RealSense から取得した点群ではなく ModelNet40 [20]で学習した結果であるため，これらの学習重みは FARO と RealSense の点群に特化していない．一般的に教師あり学習では，学習したデータセットと似たものでは高い精度が得られる一方，異なるものでは精度が低下する傾向がある．本実験では，学習したデータセットと異なる実際の距離画像センサにより取得した点群を対象に実験を行った．それにもかかわらず，ロバストに比較的小さな誤差での位置姿勢推定を実現したことから，実用面での提案手法の有効性が示されたと言える．

Table 4 Experimental results when matching region search is successful

Real data		success rate	mean error	mean error on success
PointpartNet [12]	Rotation	72.58%	25.80 °	4.09 °
	Translation	74.73%	0.176 m	0.046 m
PointpartNet++	Rotation	85.65%	10.30 °	4.89 °
	Translation	97.22%	0.069 m	0.037 m



(a) before registration　　(b) after registration

Fig. 15 Registration sample from real data. blue: **full**, red: **part**, black: ground truth of **part**

5. 結　論

本論文では，異なるサイズの点群に対して，部分的な特徴抽出を行う深層学習を用いた位置合わせ手法 PointpartNet [12]を改善し，推定精度を向上した．本手法は従来の点群を切り分けし部分点群の特徴を抽出することでマッチング領域探索とラフ位置合わせを行った．さらに，SuperGlue [19]を用いて点群同士の対応点を算出し精密位置合わせを行うことで，サイズの異なる点群の位置合わせを実現した．また，従来手法と比較し本手法がよりロバストかつ精度が高いことを示した．また，実データを対象とした精度検証実験でも良好な結果を得られた．今後は，色情報も考慮することにより点群位置合わせをさらに高精度化することを目標とする．

謝　辞

本研究は JSPS 科研費 19H04191 の助成を受けたものである．

参 考 文 献

1) X. Zhao et al.: LiDAR-ToF-Binocular depth fusion using gradient priors, 2020 Chinese Control And Decision Conference (CCDC), (2020) 2024.
2) P. Hambarde, A. Dudhane, P. W. Patil, S. Murala and A. Dhall: Depth Estimation From Single Image And Semantic Prior, 2020 IEEE International Conference on Image Processing (ICIP), (2020) 1441.
3) C. R. Qi, H. Su, K. Mo and L. J. Guibas: Pointnet: Deep learning on point sets for 3d classification and segmentation, Proc. Computer Vision and Pattern Recognition (CVPR), (2017) 77.
4) Y. Li, R. Bu, M. Sun, W. Wu, X. Di and B. Chen: Pointcnn: Convolution on x-transformed points, Advances in neural information processing systems,, (2018) 820.
5) C. R. Qi, L. Yi, H. Su, and L. J. Guibas: Pointnet++: Deep hierarchical feature learning on point sets in a metric space, In Advances in Neural Information Processing Systems, (2017) 5099.
6) H. Cao, R. Zhan, Y. Ma, C. Ma and J. Zhang: LFNet: Local Rotation Invariant Coordinate Frame for Robust Pointcloud Analysis, in IEEE Signal Processing Letters, **28**, (2021) 209.
7) Z. Zhang, B. Hua, D. W. Rosen and S. Yeung: Rotation Invariant Convolutions for 3D Pointclouds Deep Learning, 2019 International Conference on 3D Vision (3DV), (2019) 204.
8) Y. Rao, J. Lu and J. Zhou: Spherical Fractal Convolutional Neural Networks for Pointcloud Recognition, 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), (2019) 452.
9) Y. Aoki, H. Goforth, R. A. Srivatsan and S. Lucey: PointNetLK: Robust & Efficient Pointcloud Registration Using PointNet, 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), (2019) 7156.
10) Y. He and C. Lee: An Improved ICP Registration Algorithm by Combining PointNet++ and ICP Algorithm, 2020 6th International Conference on Control, Automation and Robotics (ICCAR), (2020) 741.
11) V. Sarode, A. Dhagat, R. A. Srivatsan, N. Zevallos, S. Lucey and H. Choset: MaskNet: A Fully-Convolutional Network to Estimate Inlier Points, 2020

International Conference on 3D Vision (3DV), (2020) 1029.
12) S. Yan, S. Pathak, and K. Umeda: PointpartNet: 3D Point-cloud Registration via Deep Part-based Feature Extraction, Advanced Robotics, DOI: 10.1080/01691864.2022.2084346.
13) P. J. Besl and N. D. McKay: A method for registration of 3-D shapes, in IEEE Transactions on Pattern Analysis and Machine Intelligence, **14**, 2, (1992) 239.
14) P. Biber and W. Strasser: The normal distributions transform: a new approach to laser scan matching, Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) , **3**, (2003) 2743.
15) V. Golyanik, S. A. Ali and D. Stricker: Gravitational Approach for Point Set Registration, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (2016) 5802.
16) A. Carballo, S. Seiya, J. Lambert, H. Darweesh, P. Narksri, L. Y. Morales, N. Akai, E. Takeuchi, and K. Takeda: End-to-end autonomous mobile robot navigation with model-based system support, Journal of Robotics and Mechatronics, **30**, 4, (2018) 563.
17) A. Kurobe, Y. Sekikawa, K. Ishikawa and H. Saito: CorsNet: 3D Pointcloud Registration by Deep Neural Network, in IEEE Robotics and Automation Letters, **5**, 3, (2020) 3960.
18) M. Munaro, F. Basso and E. Menegatti, "Tracking people within groups with RGB-D data," 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, (2012) 2101.
19) P. -E. Sarlin, D. DeTone, T. Malisiewicz and A. Rabinovich: SuperGlue: Learning Feature Matching With Graph Neural Networks, 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), (2020) 4937.
20) Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao: 3d shapenets: A deep representation for volumetric shapes, In Proceedings of the IEEE conference on computer vision and pattern recognition, (2015) 191.