# 複合現実によるロボットの空間認識可視化のための Semantic-ICPを用いたキャリブレーション

Calibration System Using Semantic-ICP for Visualization of Robot Spatial Perception Through Mixed Reality

> 中村仁 Hitoshi Nakamura

エルハフィロトフィ Lotfi El Hafi 萩原良信 Yoshinobu Hagiwara 谷口忠大 Taniguchi Tadahiro

# 立命館大学

### Ritsumeikan University

To achieve symbiosis between humans and robots, it is important to know what the robots recognize in their environment. Such information can be displayed using a Mixed Reality (MR) head-mounted device to provide an intuitive understanding of a robot perception. However, a robust calibration system is required because the robot and head-mounted MR device have different coordinate systems. In this paper, we develop a semantic-based calibration system for human-robot interactions in MR using Semantic-ICP. We show that the calibration system using Semantic-ICP is better than using GICP SE(3) when the accuracy of the semantic labels is high.

## 1. はじめに

家庭環境内で人間と共同で動作するロボットが, ヘッドマ ウントディスプレイを介して人間とインタラクションを行う 場合, ロボットとデバイス間のキャリブレーションが必要であ る. 近年, 複合現実や拡張現実を用いたロボットの内部表現の 可視化や, それらの情報を元にした追加学習を行う研究が行わ れている [Collett 06,Liu 18]. このような研究においてロボッ トとヘッドマウントディスプレイが異なる座標系を持っている 場合, キャリブレーションを行わなければロボット自身の座標 系に基づいた内部表現をヘッドマウントディスプレイの座標系 に基づいた現実空間の該当する位置に可視化することが出来 ない.

既存手法の多くでは AR マーカを用いてキャリブレーショ ンを行う手法が用いられてきた [Liu 18]. しかし, AR マーカ を用いる場合は, AR マーカが設置された範囲内でしかキャリ ブレーションを行うことができないため,環境内に多数の AR マーカを配置する必要があり,またそれには手間がかかる.こ の問題を解決するためには,ロボットとヘッドマウントディス プレイのそれぞれから得られる Point Cloud を用いたキャリ ブレーションが利用できる.しかしながら,両デバイス間にお ける Point Cloud の取得方法が異なっている場合,正確なキャ リブレーションは困難である.

そこで本研究では、より正確な位置にロボットの内部表現 を描画するために、Point Cloud に画像特徴量としてセマン ティックラベルの確率分布を付加したものをキャリブレーショ ンに用いる手法を提案する.図1に、2.4節で詳しく述べる Semantic-ICPを用いたロボットとヘッドマウントディスプレ イにおけるキャリブレーション手法の図を示す.

## 2. 関連研究

### 2.1 キャリブレーション手法

Liu らは、ロボットの関節上にマーカを描画する際に、ロ ボットの正確な位置を認識するために AR マーカを使用して いた [Liu 18]. この研究では、ヘッドマウントディスプレイを 装着した人間は常にロボットを見ているため、ロボットに搭載



図 1: Semantic-ICP を用いたロボットとヘッドマウントディ スプレイ間のキャリブレーション

されたディスプレイ上に AR マーカを表示することによって キャリブレーションを行っていた. AR マーカを用いたキャリ ブレーション手法ではロボットが移動したり人間が移動したり して AR マーカを捕捉できなくなった場合,キャリブレーショ ン誤差が発生してしまう.また人間が誤差を軽減するために何 度もタグを見て再キャリブレーションを行う必要があるという 欠点がある.

本研究ではこれらの欠点を踏まえ、人間のキャリブレーショ ンに関わる作業コストを削減し、誤差の少ないキャリブレー ションを実現するために、互いの Point Cloud を用いた自動 でキャリブレーションを行う手法を提案する.

### 2.2 ICP と Generalized-ICP

二つの Point Cloud をマッチングする手法として Iterative Closest Point (ICP) [Besl 92] がよく用いられる.また, Segal らは ICP を応用し,確率的なモデルを使った Generalized-ICP (G-ICP) を提案した [Segal 09]. ICP, G-ICP は大きく分け て以下の 3 つのステップで処理が行われる.

STEP 1 X<sub>t</sub>, X<sub>s</sub>の対応関係 I を計算する

- STEP 2 最適な変換行列 T\* を求める
- **STEP 3** 前サイクルの最適な変換行列 **T<sup>old</sup>** と **T**<sup>\*</sup> を比較し, その誤差が *ϵ* より小さければ終了する

ただし,  $\mathbf{X}_t = \left\{ \mathbf{x}_1^t, \mathbf{x}_2^t \dots \mathbf{x}_n^t \right\}$ ,  $\mathbf{X}_s = \left\{ \mathbf{x}_1^s, \mathbf{x}_2^s \dots \mathbf{x}_n^s \right\}$ を Point Cloud 集合, I を二つの Point Cloud の対応関係集合,

連絡先: 中村 仁, 立命館大学情報理工学研究科, 滋賀県草津市 野路東 1-1-1, nakamura.hitoshi@em.ci.ritsumei.ac.jp

**T**を二つの Point Cloud 間の変換行列,  $\epsilon$ を変換行列後の二 つの Point Cloud 間の許容する誤差とする.まず STEP 1 に おいて **X**<sub>t</sub>, **X**<sub>s</sub> 内の点一つ一つの対応関係を計算する.最近傍 探索手法によって式 (1)を用いることによって最も近い点を探 索する.関数 *d* は **x**<sup>t</sup><sub>i</sub>( $\in$  **x**<sub>t</sub>)と一番近い距離にある点 **x**<sup>s</sup><sub>i</sub>( $\in$  **x**<sub>s</sub>) を取得するものである.式 (1) によって **x**<sub>t</sub> と **x**<sub>s</sub> の全ての点 の対応を求めたものの集合が I である.

$$d(\mathbf{x}_{i}^{t}, \mathbf{X}_{s}) = \min_{\mathbf{x}_{i}^{s} \in \mathbf{x}_{s}} \|\mathbf{x}_{i}^{s} - \mathbf{x}_{i}^{t}\|$$
(1)

STEP 2 では最適な変換行列を求める計算を行う. ここでの 変換行列は  $\mathbf{T} = \mathbf{Rx}_{i}^{t} + \mathbf{p}$ と表される. ここで,  $\mathbf{R}$ はユークリッ ド空間での回転行列,  $\mathbf{p}$ は並進ベクトルを表している.  $\mathbf{x}_{s}$ を固 定して  $\mathbf{x}_{t}$  の中から  $\mathbf{x}_{i}^{t}$ を取得し, その点  $\mathbf{x}_{i}^{t}$ を  $\mathbf{R}$ によって回転 させ,  $\mathbf{p}$ によって平行移動を行う. ICP の場合, 式 (2)を用いて 最適な変換行列を求める. ここで  $\omega_{i}$  は,  $\|\mathbf{x}_{i}^{t} - \mathbf{T} \cdot \mathbf{x}_{i}^{s}\| \leq \mathbf{d}_{\max}$ を満たす時に  $\omega_{i} = 1$ を, それ以外は  $\omega_{i} = 0$ を取る変数であ る. また,  $d_{max}$  は, 計測誤差によるマッチング精度が減少す るのを防ぐパラメータである. 対応付けられた 2 つの点の距 離が  $d_{max}$  以上であった場合, 計測誤差として計算対象外とす るためである.

$$\mathbf{T}^* \Leftarrow \operatorname{argmin}_{\mathbf{T}} \sum_{\mathbf{i}} \omega_{\mathbf{i}} \| \mathbf{T} \cdot \mathbf{x}_{\mathbf{i}}^{\mathbf{s}} - \mathbf{x}_{\mathbf{i}}^{\mathbf{t}} \|^2$$
(2)

### 2.3 GICP-SE(3)

Parkison らは G-ICP を Absil らの手法 [Absil 08] を参考に して特殊ユークリッド群 (SE(3)) に適応させた GICP-SE(3) を提案した [Parkison 18]. G-ICP との違いとして最適な変 換行列を求める式が  $f_{GICP}$  に変更されている. ここで  $C_k$  は  $C_k \triangleq \sum_{k}^{t} + R \sum_{k}^{s} R^T$  と表される.  $\sum_{k}^{t} \sum_{k}^{s}$  は各 Point Cloud が正規分布から得られていると仮定した際の分散である. また  $f_{GICP}$  は式 (3) である. ここで  $P_{\alpha}$  関数はコーシー損失関数で あり,  $P_{\alpha}(x) = \alpha^2 log(1 + \frac{x}{\alpha^2})$ とする. また,  $\alpha$  は損失が直線 的に減少し始める場所を制御するパラメータ, n はマッチング 対象の点の数である. また, 式 (3) 内の変換行列 **T**  $\in$  **SE**(3) は式 (4) で示される.

$$\mathbf{T}^{*} \leftarrow argmax_{\mathbf{T}\in\mathbf{SE}(\mathbf{3})}f_{GICP}$$
$$f_{GICP} = \sum_{k}^{n} p_{\alpha}(\|\mathbf{x}_{k}^{t} - \mathbf{T}(\mathbf{x}_{k}^{s})\|_{\mathbf{C}_{k}}^{2}) \qquad (3)$$

$$\mathbf{T} \in \mathbf{SE}(\mathbf{3}) = \begin{pmatrix} \mathbf{R} & \mathbf{p} \\ 0 & 1 \end{pmatrix}$$
(4)

### 2.4 Semantic-ICP

Parkison らは GICP-SE(3) にセマンティックラベルの確率分 布を付加した Semantic-ICP を提案した [Parkison 18]. GICP-SE(3) と異なる点として,式(3) に示した関数  $F_{GICP}$  に,式 (5) によって得られる重み  $\omega_{\mathbf{k}}$  を新たなパラメータとして加え た関数  $F_{SICP}$  を用いている点である.

$$\omega_{\mathbf{k}} \triangleq \sum_{\mathbf{s}_{\mathbf{k}} \in \mathbf{C}} p(\mathbf{r}_{\mathbf{k}} | \mathbf{i}_{\mathbf{k}}, \mathbf{x}_{\mathbf{k}}^{\mathbf{t}}, \mathbf{X}_{\mathbf{k}}^{\mathbf{s}}; \mathbf{T}^{\mathbf{old}}) \mathbf{p}(\mathbf{s}_{\mathbf{k}} | \mathbf{i}_{\mathbf{k}}, \mathbf{x}_{\mathbf{t}})$$

$$p(\mathbf{s}_{\mathbf{k}} | \mathbf{i}_{\mathbf{k}}, \mathbf{x}_{\mathbf{s}}) \mathbf{p}(\mathbf{i}_{\mathbf{k}} | \mathbf{x}_{\mathbf{k}}^{\mathbf{t}}, \mathbf{x}_{\mathbf{k}}^{\mathbf{s}}) \tag{5}$$

式 (5) は式 (6) の式より導かれている. *F*<sub>SICP</sub> を式 (7) の ように表す. 式 (6) は二つの Point Cloud が与えられた時の それらの対応関係,セマンティックラベル情報,変換行列のパ ラメータが得られる条件付き確率はベイズ則を用いて右辺のよ うに近似できることを示している.ここで, R は T の残差パ ラメータ, C をセマンティックラベル集合, S を環境内にある セマンティックラベル集合, N をセマンティックラベルの数と する.

$$p(\mathcal{R}, \mathbf{S}, \mathbf{I} | \mathbf{X}_{t}, \mathbf{X}_{s}) \propto p(\mathcal{R} | \mathbf{I}, \mathbf{X}_{t}, \mathbf{X}_{s}) \mathbf{p}(\mathbf{S} | \mathbf{I}, \mathbf{X}_{t}) \quad (6)$$

$$p(\mathbf{S} | \mathbf{I}, \mathbf{X}_{s}) \mathbf{p}(\mathbf{I} | \mathbf{X}_{t}, \mathbf{X}_{s})$$

$$\mathbf{T}^{*} \leftarrow argmax_{\mathbf{T} \in \mathbf{SE}(3)} f_{SICP}$$

$$f_{SICP} = \sum_{n \neq N}^{n \neq N} p_{\alpha}(\omega_{k} \| \mathbf{x}_{t}^{t} - \mathbf{T}(\mathbf{x}_{t}^{s}) \|_{\mathbf{C}}^{2}) \quad (7)$$

$$\tilde{\mathbf{x}}$$
 (5) でのセマンティックラベルごとの Point Cloud

また,式 (5) でのセマンティックラベルごとの Point Cloud の対応関係  $\mathbf{i}_{\mathbf{k}}$  の条件付き確率  $p(\mathbf{i}_{\mathbf{k}}|\mathbf{x}_{\mathbf{k}}^{t},\mathbf{x}_{\mathbf{k}}^{s})$  は式 (8) によっ て示される. ここで  $nn(x_{k}^{t})$  は  $x_{k}^{t}$  の  $x_{k}^{s}$  内にある近傍点の数 を求める関数である.本研究では図 1 に示したように,この Semantic-ICP を用いて実験を行う.

$$p(\mathbf{i}_{\mathbf{k}}|\mathbf{x}_{\mathbf{k}}^{\mathbf{t}}, \mathbf{x}_{\mathbf{k}}^{\mathbf{s}}) \triangleq \begin{cases} \frac{1}{N} & if \ nn(\mathbf{x}_{\mathbf{k}}^{\mathbf{t}}) = \mathbf{N} \\ 0 & otherwise \end{cases}$$
(8)

## 3. 提案手法

### 3.1 概要

Point Cloud を用いたキャリブレーションとして、一般的に ICP がよく用いられる.しかしながら、本研究ではヘッドマ ウントディスプレイとロボットが異なるセンサーを用いて異な る手法で Point Cloud を取得している為、一般的な ICP では 良い結果が得られないと考えられる.そのため、本研究では、 ヘッドマウントディスプレイとロボットのキャリブレーション を、互いのカメラセンサから得られる Point Cloud に画像特 徴量として得られるセマンティックラベルの確率分布を付加し たものを用いてキャリブレーションする手法を提案する.セ マンティックラベルとは、画像内に含まれるオブジェクトの名 前のことである.ここで、セマンティックラベルの確率分布を 付加した Point Cloud を Semantic Point Cloud と定義する. Semantic Point Cloud は以下の 3 つのステップによって生成 する.また提案手法では CNN として PSPNet [Zhao 17] を用 いる.

- **STEP 1** RGB-D カメラを使用して Point Cloud, RGB 画 像を取得する
- **STEP 2** RGB 画像を PSPNet を用いてセマンティックセグ メンテーションを行いピクセルごとにセマンティックラ ベルの確率分布を取得する
- **STEP 3** STEP 1,2 で得られたピクセルごとのセマンティッ クラベルの確率分布から得られる最も生成確率の高いセマ ンティックラベルを Point Cloud を対応させ, Semantic Point Cloud を生成する

### **3.2** ロボット側の Semantic Point Cloud の生成手法 本手法では、ロボットは環境内全体の地図を Semantic Point

本手法では、ロボットは環境内全体の地図を Semantic Point Cloud として保有しているものとする.本手法ではロボットの Semantic Point Cloud を用いた 3D の地図作成に Semantic SLAM [Zhang 18] を改変して用いた. Semantic SLAM に用 いられている RGB-D カメラの位置情報を把握するために用い られている ORB-SLAM を用いずに,ロボットの標準機能と して取得できる RGB-D カメラの位置情報をそのまま用いる. ORB-SLAM よりもより精度の高い位置情報が得られるためで ある.また本手法では 3D 地図の作成に Octomap [Hornung 13] を用いた. Octomap で得られたボクセルマップを, Semantic Point Cloud 形式に変換することによって,環境内の Semantic Point Cloud の 3D の地図を作成することができる. ここで 環境内の Semantic Point Cloud の 3D の地図のことを 3D Semantic Map と定義する.

# **3.3** ヘッドマウントディスプレイ側の Semantic Point Cloud の生成手法

本手法で用いるヘッドマウントディスプレイとして HoloLens を用いる. HoloLens は RGB 画像は取得できるが, Point Cloud を取得することが出来ないため,以下の4つのステッ プによって Point Cloud を取得する.

STEP 1 RGB 画像を撮影する

- STEP 2 各ピクセルごとにカメラ間のベクトルを生成する
- **STEP 3** 各ベクトルをカメラが向いている方向に延長し, Spatial Mapping に接触した場合,そのベクトルの長さをそ のピクセルの深度として取得する
- **STEP 4** そのピクセルの座標と深度を用いて Point Cloud と する

以上の4ステップの後,得られた Point Cloud を 3D Semantic Map 作成に用いた同様の PSPNet のモデルを使用して最も生 成確率が高いセマンティックラベルを付加した Semantic Point Cloud を生成する.

### 3.4 Semantic-ICP を用いたマッチング

ロボットによって得られる 3D の地図データの Semantic Point Cloud と HoloLens が現在見ている部分の Semantic Point Cloud を Semantic-ICP を使ってマッチングさせるこ とによってキャリブレーションを行う. Semantic Point Cloud の照合によって,互いの Semantic Point Cloud 間における最 適な変換行列がわかり,同時に互いの座標系間における座標 系変換を推定することが可能となる.本研究では 2.2 節で示 した変数  $\mathbf{X}_{\mathbf{t}}$  をロボットの 3D Semantic Map,  $\mathbf{X}_{\mathbf{s}}$  をヘッド マウントディスプレイの Semantic Point Cloud,  $\mathbf{S}$  を互いの Point Cloud 内に存在するセマンティックラベルとして考える. 式 (6) で示した重み  $\omega_{\mathbf{k}}$  の  $p(\mathbf{s}_{\mathbf{k}}|\mathbf{i}_{\mathbf{k}},\mathbf{x}_{\mathbf{s}})$  は CNN から得られるセマンティックラベルごとの生成確率を扱うが, 本研究では各 Point Cloud ごとに一番生成確率が高かったラ ベルの確率を 1, それ以外のラベルの確率を 0 にして重み  $\omega_k$ を計算する.

# 4. 実験

# 4.1 概要

本研究では、家庭環境を模した環境で、Semantic-ICP を用 いたロボットとヘッドマウントディスプレイ間のキャリブレー ションを行う実験を行う.



図 2: 実験で用いた環境の画像

#### 4.2 実験条件

本実験では家庭環境を模した実環境で実験を行う.またロ ボットとしてトヨタ自動車株式会社の Human Support Robot (HSR), ヘッドマウントディスプレイとして Microsoft 社の HoloLens を使用する.ロボットは家庭環境内の 3D の地図 データを事前に保持しているものとし,その地図データの原点 がわかっているものとする.また,ヘッドマウントディスプレ イは装着者が向いている方向の Point Cloud を取得すること ができるものとする.ロボットの座標系が右手系であり,ヘッ ドマウントディスプレイの座標系が左手系であることも既知と する.

### 4.3 本実験の目的

既存手法として,Point Cloud のマッチング手法としてよく 用いられている GICP-SE(3) を用いたキャリブレーションと, 提案手法である Semantic-ICP キャリブレーション同士の精度 の比較を行い,提案手法の妥当性を評価する.ここでのキャリ ブレーションの精度とは,互いの Point Cloud をマッチングさ せる際に得られた最適な変換行列  $\mathbf{T}^* \in \mathbf{SE}(3)$  の精度である.

### 4.4 実験方法

はじめに,実験データ作成のために,HSR を用いて家庭環 境の 3D Semantic Map を作成する.その後 HoloLens を用い て家庭環境内の四箇所 (テレビ前,作業机,本棚前,ダイニン グ)で Semantic point cloud の取得を行う.図 2 に四箇所の 実際の環境を示す.

これら四箇所の場所で得られた 3D Semantic Map と Semantic point cloud を用いて GICP-SE(3) と Semantic-ICP を実行し、キャリブレーション精度の比較を行う.またグランドトゥルースとして、HoloLens の座標系と HSR の座標系の 3 次元距離をメジャーで測定し取得する.グランドトゥルースと 各キャリブレーション手法の精度の比較として、Parkison らが Semantic-ICP の評価の際に使用した変換行列 ( $\mathbf{T} \in \mathbf{SE(3)}$ ) 誤差を用いる.ここで  $\mathbf{T_{GT}}$  はグランドトゥルースの変換行列 を、d は誤差を計算する関数を表す.またこれら 3 項目の定義 を式 (9) に示す.

 $d_{SE(3)}(\mathbf{T}^*; \mathbf{T}_{\mathbf{GT}})) \triangleq \|\log(\mathbf{T}^*\mathbf{T}_{\mathbf{GT}})^{-1})\| \qquad (9)$ 

### 4.5 実験結果

HoloLens と HSR の Semantic Point Cloud を Semantic-ICP と GICP-SE(3) でそれぞれマッチングした結果を表1に 示す.なお,表に示された数値は全て小数第三位を四捨五入し たものである.また二つの手法の結果を比較し,誤差が小さ い方の数値を太字で示す.作業机とダイニングでは Semantic-ICP の方が変換行列誤差が既存手法より小さく,精度の向上が 認められた.本棚前とテレビ前では,GICP-SE(3)と比較し, Semantic-ICP の方が大きな誤差が得られた.

### 5. 考察

精度の悪かったテレビ前と本棚前の HoloLens と HSR の Semantic Point Cloud を図 3, 図 4 に示す.図 3a と図 3b を 比較すると、テレビの前のテーブルで異なったラベルが貼られ ていることがわかる.同様に、本棚前でも、図 4a と図 4b を 比較すると本棚に異なったラベルが張られている.よって、作 業机とダイニングにおいてマッチングの精度が向上し、作業机 と本棚前では精度の低下したことから、HSR と HoloLens で 得られたセマンティックラベルが異なっている領域が大きい時 に精度が下がると考えられる.



図 3: テレビ前の Semantic Point Cloud





(a) HoloLens

(b) HSR

図 4: 本棚前の Semantic Point Cloud

# 6. おわりに

本研究では、ロボットやヘッドマウントディスプレイのキャ リブレーションを、AR マーカを用いずにキャリブレーション 精度を向上させる手法として Semantic-ICP を用いた.本実験

表	1:	変換行列誤差	$d_{SE(3)}$
---	----	--------	-------------

キャリブレーシ ョン手法	テレビ前	作業机	本棚前	ダイニング
GICP-SE(3)	0.22	5.42	0.08	0.06
Semantic-ICP	8.68	0.05	0.76	0.04

では HoloLens と HSR のカメラから取得した Point Cloud に PSPNet の出力値のセマンティックラベルを付加した Semantic Point Cloud を用いてマッチングを行った. GICP-SE(3), Semantic-ICP のキャリブレーション精度の比較を行った. 実 験の結果,互いの Semantic Point Cloud 内の一致しているセ マンティックラベルの領域が広い場合,精度が向上する傾向に あることがわかった.今後の展望としては,各ラベルの生成 確率を扱う手法の改善や,セマンティックラベル以外の特徴量 を用いた場合に精度の向上が見られるかの検証を行う予定で ある.

# 参考文献

- [Collett 06] T.H.J Collett and B.A MacDonald, "Augmented reality visualisation for player", In Proceedings 2006 IEEE International Conference on Robotics and Automation (ICRA), 2006, pp.3954-3959
- [Liu 18] Hangxin Liu, Yaofang Zhang, Wenwen Si, Xu Xie, Yixin Zhu, and Song-Chun Zhu, "Interactive Robot Knowledge Patching Using Augmented Reality", In Proceedings 2018 IEEE International Conference on Robotics and Automation (ICRA), 2018, pp.1947-1954
- [Besl 92] P.J. Besl and Neil D. McKay, "A method for registration of 3-D shapes", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.14, No.2, 1992, pp.239-256
- [Segal 09] Aleksandr V Segal, Dirk Haehnel, and Sebastian Thrun, "Generalized-ICP", Robotics:Science and Systems 2009, Vol.2, 2009
- [Absil 08] P.-A. Absil, R Mahony, and R Sepulchre, "Optimization Algorithms on Matrix Manifolds", Princeton University Press, Princeton, NJ, 2008
- [Parkison 18] Steven A. Parkison, Lu Gan, Maani Ghaffari Jadidi, and Ryan M. Eustice, "Semantic Iterative Closest Point through Expectation-Maximization", Proceedings of the British Machine Vision Conference, 2018, pp.1-17
- [Zhao 17] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia, "Pyramid Scene Parsing Network", In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp.6230-6239
- [Zhang 18] Xuan Zhang, "Real-time voxel based 3d semantic mapping with a hand held rgb-d camera", https://github.com/floatlazer/semantic\_slam, Accessed:2019-01-02
- [Hornung 13] Armin Hornung, Kai M Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard, "OctoMap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees", Autonomous Robots, 2013