

次世代型知的電動車いすの開発に関する研究

小西 亮介

鳥取大学大学院工学研究科情報エレクトロニクス専攻電気電子工学コース

Study on the development of next generation type electric wheel chair

Ryosuke KONISHI

Electrical & Electronic Engineering Course, Information Electronics Major

Tottori University

Abstract: In this paper, I describe about the development of next generation type electric wheel chair. Speech recognition technique is auxiliarily used for start and stop. Wheel chair after start moves toward target by which information is obtained from EKF-SLAM algorithm every second moment. EKF-SLAM is calculated from both distance length and direction angle obtained from LRF, which put in the wheel chair. By using value of EKF-SLAM, this wheel chair automatically move into calculated length and angle, while map of neighbouring surrounding is drawn simultaneously . . .

Key Words: LRF, speech recognition, EKF-SLAM algorithm

1. はじめに

ここでは、著者が鳥取大学に在籍した40年間のうち、特に最近約10年間に興味をもったインテリジェント（画像・音声技術）な技術を利用した室内移動ロボット（電動車いす）の開発に関する研究に絞って報告する。特に身体の不自由な人や日常に介助を必要とする高齢者や障害者の方々により安全で快適な車いすを提供することを目的としている。これまでにこの分野で行われてきた研究は大きく以下の2つに大別することができる。

一つはユーザインタフェースに関する研究である。人間の最も自然な情報伝達手段の一つに音声^{1, 2, 3, 4}や画像処理を用いて顔の方向を検出し電動車いすを制御している⁵。筋電位センサを用いるもの、舌先の動きを検出する舌圧センサを利用したものなど様々な電動車いすを制御する手法が提案されている。これらのなかで音声は人間にとって最も自然な情報伝達手段であり、音声インタフェースは、直感的で分かりやすく利便性も高いがこれまでの研究はインタフェースに関する議論が中心であるため、安全性に関する議論が少ないのが現状である。もう一方は、センサ等を用いてユーザの操作あるいは車いすの走行を補助する研究である。これにより人間の操作に対する負担を減らし、安全な走行を実現できる。後藤はレーザレンジファインダ（LRF）を車いすに搭載し、LRFの情報か

ら方向の安全圏を計算することでユーザのジョイスティック操作を補助するシステムを提案している⁶。外界センサやカメラを用いる研究は、自律移動ロボットの分野で盛んに取り組まれている^{7, 8, 9}。ここでは上記研究開発の動向を踏まえて、操作インタフェースとして音声を用いるとともにLRFで周囲環境地図を生成しながら、自律走行を可能にする機能を加味した電動車いすの開発について述べる。

2. システム構成

本システムではSUZUKI株式会社製電動車いすMC2000Sを用いる。車いす周囲の環境を取得するセンサとしてLRF、LRFの信号を処理するノートPC、車いすを制御するためにPICを用いる。LRFは車いすの中心線上かつ車いすの駆動中心の88cm前方、床面からの高さ49cmの位置に床面と平行になるように取り付けた。PICはMASTERとしてPIC18F2620を一つ、また、SLAVEとしてモータ制御用に一つ、左右エンコーダ信号取得のためのパルスカウンタ用に二つ、合計三つのPIC16F876を用いる。システム構成をFig.1に示す。システムの大まかな流れは以下ようになる。

1. LRFにより周囲環境情報を取得する。
2. PCはLRFのデータを取り込み、PICから送

られるオドメトリ情報と組み合わせSLAMを適用する。

3. SLAMの結果からPCからPICへ車いすの動作信号を送信する。
4. PICはPCから受信した動作信号をモータ制御用のPICを通して車いす本体のメインコントローラへ送信し、車いすを動作させる。

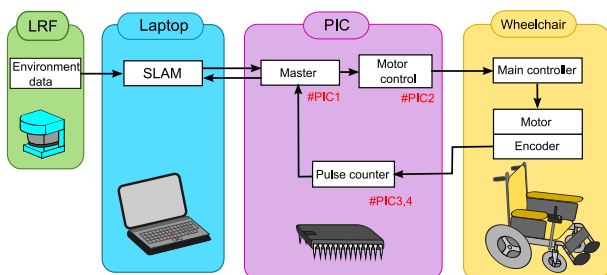


Fig.1 システムの構成

2.1 LRF

本システムで使用するLRFとしてSICK社製レーザレンジファインダLMS-200を用いる。外観をFig.2に示す。LMS200の仕様をTable1に示す。



Fig.2 LRFの外観

本システムで使用するLRFの測定原理はTOF (Time-of-Flight; 飛行時間) である。TOFとはレーザを発射してから測定対象にあたり、反射したレーザが再びLRFに戻ってくるまでの時間から対象物までの距離を求める手法である。測定対象物までの距離を d とすると、 d は光速 c 、レーザを発射し

てから戻ってくるまでの時間 Δt を用いて式 (1) で表される。

$$d = \frac{c\Delta t}{2} \tag{1}$$

また、本研究で用いるLMS-200は測定可能距離0~8mあるいは0~80m、スキャン角度180°、角度分解能1.00°、0.50°、0.25° (0.25°の場合はスキャン角度100°) と測定条件を選択することが可能である。本研究では距離分解能が高く、なるべく高い分解能を維持したまま広い視野を確保できることが望ましい。したがって、測定距離0~8m、スキャン角度180°、角度分解能0.50°と設定した。また、LRFとPCはシリアル通信方式により通信する。通信速度は9600bpsと設定した。なお、通信速度は最大38400bps (専用機器を使用した場合は500kbps) まで設定が可能である。

Table 1 LRFの仕様

項目	仕様内容
型式	LMS200-30106
測定距離	0 - 80[m]
視野角度	0 - 180[deg]
角度分解能	0.25 - 1.00[deg]
システム誤差	± 15[mm]
統計的誤差	5[mm]
モータ回転速度	75[Hz]
レーザ光	905[nm]
レーザクラス	クラス1 (アイ・セーフ)
電源電圧	24[V]
消費電力	20[W]
重量	4.5[kg]
外形寸法	156 x 155 x 210 [mm]

実際の操作をFig.3にフローチャートで示す。本システムでは距離分解能が高く、なるべく高い分解能を維持したままPCでLRFを駆動し周囲環境から距離情報を取得する。そのデータをもとにEKF-SLAMアルゴリズムにもとづく処理計算を行い車いすの位置・姿勢情報を得る。その後、ユーザが音声命令を行うと、その命令に従って進むべき方向 (速度・角度) が計算されその方向への初期動作を行う。その後、EKF-SLAM処理によって、車いすの自己位置を推定すると同時に地図生成を行う。それを繰り返すことによって車いすを走行するシステムである。

LRFにより、周囲環境情報を取得し、そこから環境中の特徴量を抽出する。その後、抽出した特徴点をランドマークとして対応づける。特徴点データとPICから送られたオドメトリ情報に対して

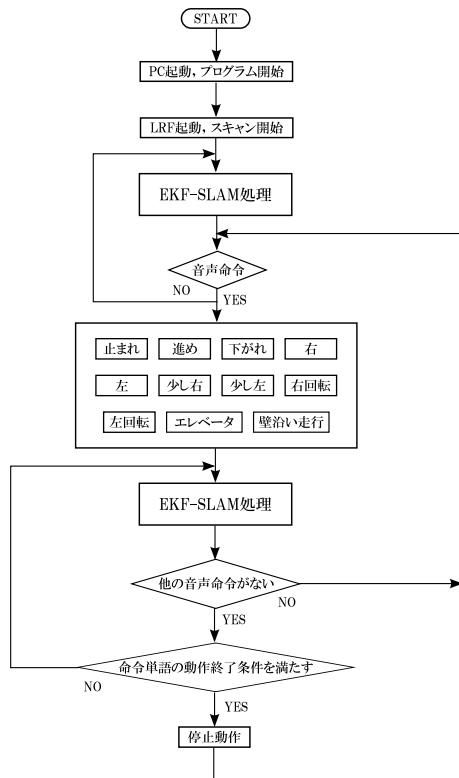


Fig.3 操作手順

EKF-SLAMを適用する。EKFは車いすの自己位置および姿勢情報を修得するが、周辺環境に対する自己位置を推定するためにはSLAMアルゴリズムが必要である。SLAMは外界センサを用いて自身の位置および周辺環境を同時に推定する手法である。SLAMでは事前にロボットに環境情報を入力する必要がないという利点がある。

2.2 EKF-SLAMアルゴリズム

LRFから得られた周囲環境情報からまず、環境中の特徴点を抽出する。その後、抽出した特徴点と過去に抽出された特徴点をランドマークとして対応づける。その後、特徴点データとオドメトリ情報に対して、EKF-SLAMを適用し、車いすの自己位置及び周辺環境を推定する。

2.2.1 特徴点抽出

本研究では環境の幾何形状に着目して計測データから特徴点を抽出する。抽出する特徴点として廊下の角¹⁰⁾や、平面の壁などの線^{11), 12)}などがある。本研究では比較的抽出が容易であることから、廊下の角を抽出し特徴点として用いる。特徴点抽

出処理として、角検出アルゴリズムと不連続点検出アルゴリズムを提案する。例としてFig.4に示す環境では角の形状が陽に表われる場所 (Fig.4の右上) と LRF のレーザ光が届かず廊下の角付近において死角となり、角の形状が不連続な点として表される場所 (Fig.4左上) が存在する。本研究ではこれら2種類の点を前者については角検出アルゴリズムにより抽出し、後者は不連続点検出アルゴリズムにより抽出する。以下に二つのアルゴリズムの詳細について述べる

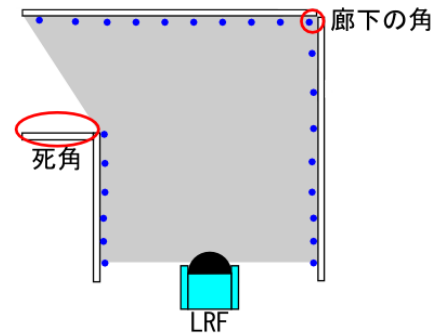


Fig.4 廊下の角の種類

2.2.2 角検出アルゴリズム

角検出アルゴリズムの処理をFig.5を用いて説明する。処理の流れは以下のようにになっている。また、フローチャートをFig.6に示す。

1. LRFから得られた N 個の計測データ群 S の最初の点からある距離 a 離れた点を始点とし、最後の点から a 離れた点を終点とする。
2. LRFから得られた N 個の計測データ群 S から i 番目の点 s_i (最初は始点) からその前後に a 離れた2点 s_{i-n} , s_{i+m} の3点を取り出す。
3. 3点のなす角 ϕ_i を式(2)より求める。
4. eq.3より求めた α_i がしきい値 α_{th} より小さく、かつ極小値の場合、 s_i を廊下の角の点とする。
5. s_i が終点でないなら i を1増加させ2に戻る。

$$\phi_i = \cos^{-1} \left(\frac{\overrightarrow{s_i s_{i-n}} \cdot \overrightarrow{s_i s_{i+m}}}{|\overrightarrow{s_i s_{i-n}}| |\overrightarrow{s_i s_{i+m}}|} \right) \quad (2)$$

$$\alpha_i = |90 - \phi_i| \tag{3}$$

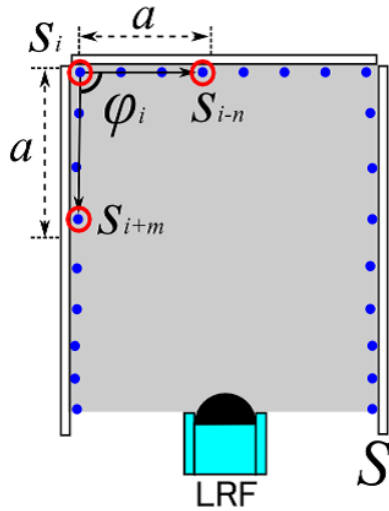


Fig.5 角検出アルゴリズム

Fig.7 に実際の環境を計測したデータ例を示す. Fig.8 はこの計測データについて, 対象物の方向 θ における α_i の推移を示している. 図に示すように角点では α_i の値が急激に変化していることがわかる. 結果より, 角点が正確に抽出できていることがわかる.

2.2.3 不連続点の検出

不連続点アルゴリズムの処理を Fig.9 を用いて説明する. 同図のような環境ではレーザ光が届かず死角となる部分において LRF の計測データが急激に変化する. 不連続点検出アルゴリズムではこの点を検出する. 処理の流れは以下のようになっている. また, フローチャートを Fig.10 に示す.

1. N 個の計測データ群 S から i 番目の点 s_i の距離データを d_i とする ($0 < i < N - 1$).
2. 式 (4) を用いて d_i と d_{i+1} の差分 Δd を求める.
3. Δd がしきい値 d_{th} より大きければ, s_i, s_{i+1} を不連続点とする.
4. s_i が LRF の最後の点から一つ前の点でない場合 i を 1 つ増加させ 1. に戻る.

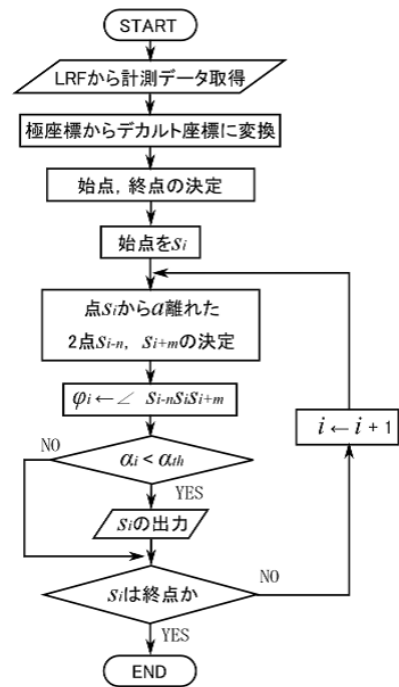


Fig.6 角検出アルゴリズムの処理の流れ

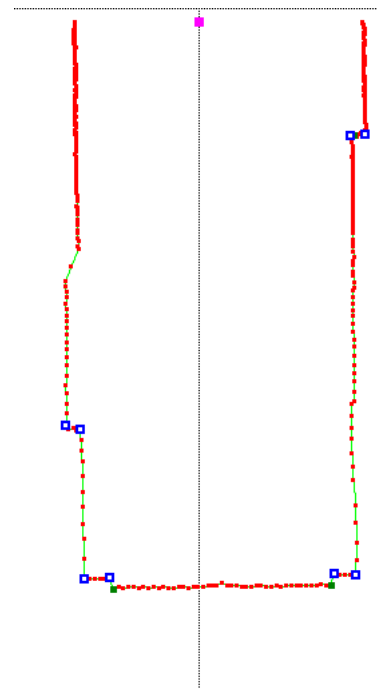


Fig.7 計測データの例

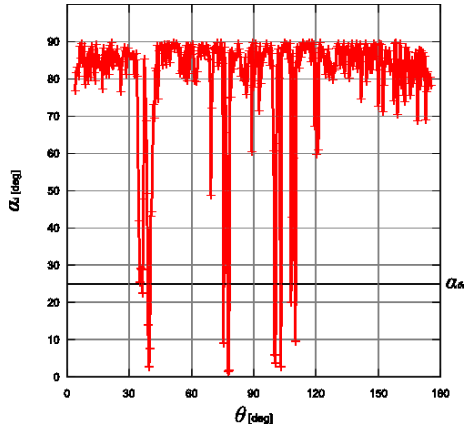


Fig.8 Fig.7における α_i の推移

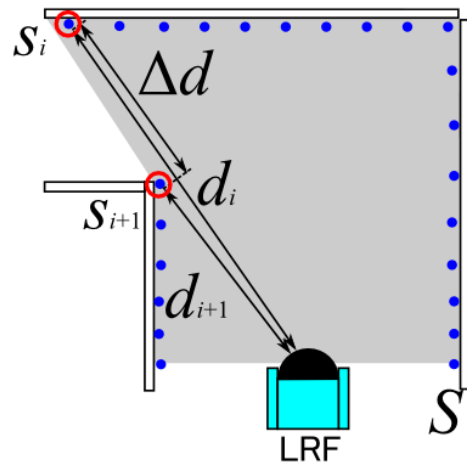


Fig.9 不連続点検出アルゴリズム

$$\Delta d = |d_i - d_{i+1}| \quad (4)$$

Fig.11 に実際の環境を計測したデータ例を示す. Fig.12 はこの計測データについて, 対象物の方向 θ における Δd の推移を示している. 図のように不連続点では, Δd の値が急激に変化していることがわかる. 不連続点検出アルゴリズムを適用した結果を Fig.13 に示す. 同図における○点はそれぞれ計測点, 角点, 不連続点を表している. 結果より, 不連続点が抽出できていることがわかる. しかし, d_{th} によっては Fig.13 のように計測点の間隔が広いところで過検出が起きてしまうことが考えられる. そこで, 不連続点検出アルゴリズムに改良を加え, 過検出を減らすことを考える. 詳細は省略する. なお, しきい値 d_{th} については 4.1 で述べる実験結果により決定する.

2.2.4 EKF-SLAM における状態方程式

EKF では状態ベクトルが車いすの位置及び姿勢であった. EKF-SLAM では, 状態ベクトルが車いすの位置及び姿勢だけでなく, グローバル座標におけるランドマークの位置が含まれる. SLAM における状態ベクトルは以下のように表される.

$$\mu_k = (x_k \quad y_k \quad \theta_k \quad m_{0,x} \quad m_{0,y} \quad 1 \quad \dots \quad m_{N,x} \quad m_{N,y} \quad 1) \quad (5)$$

ランドマークの総数を N とすると, 状態ベクトルの次元は $(3N + 3)$ である. また, その共分散行

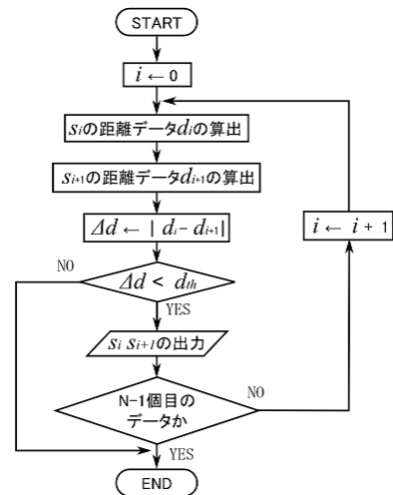


Fig.10 不連続点検出の処理の流れ

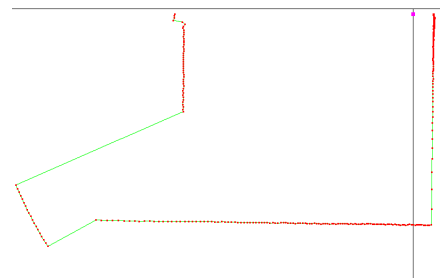


Fig.11 計測データ例 2

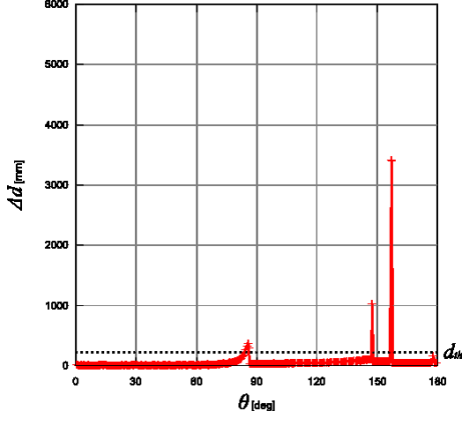


Fig.12 Fig.11における推移

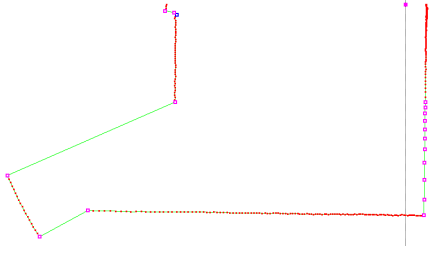


Fig.13 Fig.11における不連続点検出結果

列 Σ_k の大きさは $(3N+3) \times (3N+3)$ となる．したがって，SLAMにおける状態方程式は以下のようになる

$$\begin{aligned} \mu_k &= \mu_{k-1} \\ &+ \begin{pmatrix} v_k \cos(\theta_{k-1} + \frac{\omega_k \Delta t}{2}) \Delta t \\ v_k \sin(\theta_{k-1} + \frac{\omega_k \Delta t}{2}) \Delta t \\ \omega_k \Delta t \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \epsilon_k \end{aligned} \quad (6)$$

$$\begin{aligned} \mu_k &= \mu_{k-1} \\ &+ \mathbf{F}_x^T \begin{pmatrix} v_k \cos(\theta_{k-1} + \frac{\omega_k \Delta t}{2}) \Delta t \\ v_k \sin(\theta_{k-1} + \frac{\omega_k \Delta t}{2}) \Delta t \\ \omega_k \Delta t \end{pmatrix} \\ &+ \epsilon_k \end{aligned} \quad (7)$$

$$\mathbf{F}_x = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \end{pmatrix} \quad (8)$$

ここで，EKF-SLAMにおいては最初，ランドマークの座標は全て未知である．したがって，状態ベクトル及びその共分散の初期値は以下のように設定する．

$$\mu_0 = \begin{pmatrix} x_0 & y_0 & \theta_0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{pmatrix} \quad (9)$$

$$\Sigma_0 = \begin{pmatrix} 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \infty & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \infty \end{pmatrix} \quad (10)$$

EKF-SLAMではロボットの初期位置がグローバル座標系の原点として扱われる．本研究では車いすの初期位置及び姿勢を $(x_0, y_0, \theta_0) = (0, 0, \frac{\pi}{2})$ とした．初期位置はわかっているため共分散行列の車いすの状態に関する部分はゼロ行列となり，ランドマークの位置は未知であるためランドマークの位置に関する部分の値は無限大である．ここでに見られるように，共分散の値はカルマンゲインに影響を与える．カルマンゲインによって修正ステップにおける修正量の寡多が決定するため，共分散の値が極端に高い値であるとき，発散してしまう可能性がある．そこで，本研究では，初期値の値をある値 A として以下のように設定する．

$$\Sigma_0 = \begin{pmatrix} 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & A & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & A \end{pmatrix} \quad (11)$$

2.2.5 EKF-SLAMにおける観測方程式

EKF-SLAMにおける観測ベクトルはEKFのときと同様にLRFから見た特徴点の距離 r と方向 ϕ である．したがって，観測ベクトル，観測方程式は以下のようになる．

$$\begin{aligned} \mathbf{z}_k &= \begin{pmatrix} r_k \\ \phi_k \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} \sqrt{(m_{j,x} - x)^2 + (m_{j,y} - y)^2} \\ \text{atan2}(m_{j,y} - y, m_{j,x} - x) - (\theta - \theta_0) \\ 1 \end{pmatrix} \\ &+ \delta_k \end{aligned} \quad (12)$$

2.2.6 EKF-SLAMにおける予測, 修正, 更新の流れ

EKF-SLAMにおける各ステップの計算は以下のようになる。

$$\mu_k = \mu'_{k-1} + \mathbf{F}_x^T \begin{pmatrix} v_k \cos(\theta_{k-1} + \frac{\omega_k \Delta t}{2}) \Delta t \\ v_k \sin(\theta_{k-1} + \frac{\omega_k \Delta t}{2}) \Delta t \\ \omega_k \Delta t \end{pmatrix} \quad (13)$$

$$\Sigma_k = \mathbf{G}_k \Sigma'_{k-1} \mathbf{G}_k^T + \mathbf{F}_k \mathbf{R}_k \mathbf{F}_k^T \quad (14)$$

$$\mathbf{S}_k = \mathbf{H}_k \Sigma_k \mathbf{H}_k^T + \mathbf{Q}_k \quad (15)$$

$$\mathbf{K}_k = \Sigma_k \mathbf{H}_k^T (\mathbf{S}_k)^{-1} \quad (16)$$

$$\mu_k = \mu_k + \mathbf{K}_k (\mathbf{z}_k - \mathbf{h}_k) \quad (17)$$

$$\Sigma_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \Sigma_k \quad (18)$$

$$\mu'_k = \mu_k \quad (19)$$

$$\Sigma'_k = \Sigma_k \quad (20)$$

式(13), 式(14)では状態方程式に従い, 状態ベクトル及び共分散を予測する. ヤコビ行列 \mathbf{G}_k は以下のようになる.

$$\mathbf{G}_k = \mathbf{I} + \mathbf{F}_x^T \begin{pmatrix} v_k \cos(\theta_{k-1} + \frac{\omega_k \Delta t}{2}) \Delta t \\ v_k \sin(\theta_{k-1} + \frac{\omega_k \Delta t}{2}) \Delta t \\ \omega_k \Delta t \end{pmatrix} \mathbf{F}_x \quad (21)$$

また, 式(14)の右辺第2項は \mathbf{R}_k を状態ベクトルの次元まで拡張したものである. EKF-SLAMでは特徴点-ランドマーク間の対応関係が複数得られたときは, 式(15)~式(18)を対応関係が得られた特徴点の個数分繰り返す. ここで, ヤコビ行列 \mathbf{H}_k は車いすの状態と対応関係のとれたランドマークの座標にしか依存していない. 式(19), 式(20)は修正後の状態及びその共分散をその時刻における最適な推定値として更新する.

2.2.7 新しく見つかったランドマークへの対応

ここで, 特徴点間の対応付けアルゴリズムで対応関係が推定されなかった特徴点への対応について述べる. 対応関係が推定されなかった特徴点は新しく観測されたランドマークとして状態ベクトルに追加される. このとき, 共分散行列も2.2.4で述べた初期値 A を設定して同時に拡張される. ここで, 状態ベクトルを追加すると共分散行列も増加するため, 時間の経過に従い, EKF-SLAMの演算量は時間の経過とともに増加する. また, 対応関係が推定されない特徴点は新しく見つかった特徴点だけでなく, 過検出による点であることも考

えられる. したがって, 過検出点まで含めて状態ベクトルを拡張してしまうと, 演算量が大幅に増加してしまうことが考えられる. そこで, 本研究では, これらの対応関係が得られなかった点を状態ベクトルに追加する前に一時的に保存しておく領域を確保し, その領域から新たに状態ベクトルに追加すべきランドマークを選び出す. 本研究ではこの領域を暫定ランドマークリストと呼ぶ. また, 暫定ランドマーク中に登録された特徴点を暫定ランドマークと呼ぶ.

暫定ランドマークリストを用いた状態ベクトルの追加は以下のようになる. まず, 新たに得られた特徴点を暫定ランドマークリストに登録する. その後, 次の対応関係推定時において, モデルとして状態ベクトルに登録されているランドマークだけでなく, 暫定ランドマークに登録されている暫定ランドマークを含めてモデルとし, 対応関係を推定する. ここで, 状態ベクトルに追加すべき暫定ランドマークは以降の観測データでも特徴点として抽出される可能性が高い. したがって, 各暫定ランドマークに評価値を設定し, 対応関係推定時に現在観測された特徴点と対応関係が得られるたびに評価値を足していく. そして, あるしきい値を設け, しきい値を超えた評価値をもつ暫定ランドマークを新しいランドマークとして状態ベクトルに追加する. この暫定ランドマークリストによって, 不要な特徴点が状態ベクトルに追加されることを防ぐことができる.

3. 地図生成

前述のEKF-SLAMを適用することにより, 車いすの状態に関する最適な推定量を得ることができる. よって, LRFのスキャンデータを用いて車いすの初期位置を原点とする周囲環境地図を構成することができる. 本研究では, LRF座標系で表現されたLRFのスキャンデータを車いすの現在位置, 及び姿勢を用いてグローバル座標に写像し, これを重ね合わせることで, 周囲環境地図を構成する. 処理の流れは以下のようになる

$$\begin{pmatrix} sx'_{k,l} \\ sy'_{k,l} \end{pmatrix} = \begin{pmatrix} \cos(\mu_{k,\theta} - \theta_0) & -\sin(\mu_{k,\theta} - \theta_0) \\ \sin(\mu_{k,\theta} - \theta_0) & \cos(\mu_{k,\theta} - \theta_0) \end{pmatrix} \begin{pmatrix} sx_{k,l} \\ sy_{k,l} \end{pmatrix} + d_{LRF} \begin{pmatrix} \cos \mu_{k,\theta} \\ \sin \mu_{k,\theta} \end{pmatrix} + \begin{pmatrix} \mu_{k,x} \\ \mu_{k,y} \end{pmatrix} \quad (22)$$

処理結果をFig.14に示す. より, 正しく周囲環境が表現できていることがわかる. しかし, 単に重ね

合わせるだけでは、移動するに従いデータ数が膨大になる。また、後述する自動走行のための走行領域の検出時における処理の複雑性が増す。そこで、本研究では環境を格子状に区切り、その一つ一つの格子が占有されているかどうかを確率的に表現する占有格子地図 (Occupancy grid map) を用いる。ここで、各格子のことをセルと呼ぶ。占有格子地図におけるセルはその占有度合いに応じて 0~1 の値をもっており、数値が高いほど、そのセルが占有されている確率が高い。この占有格子地図を用いることで、セルの占有度合いを参照することで障害物の有無を判定でき、効率的に処理することが可能になる。占有格子地図の例を Fig.15 に示す。

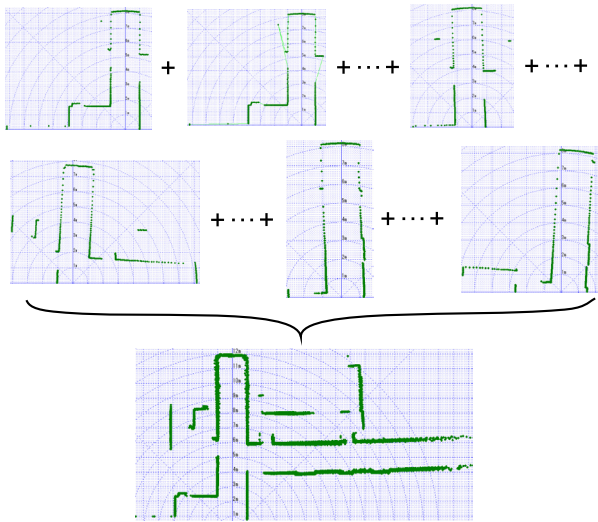


Fig.14 重ねあわせによる地図生成

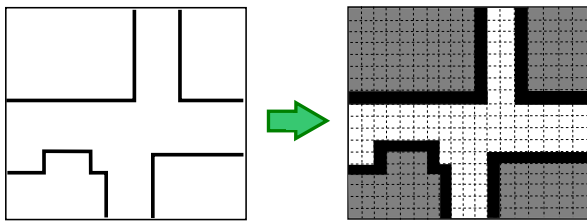


Fig.15 占有格子地図の概要

占有格子地図について述べる前に、センサのモデルを設定する必要がある。本研究で使用する LRF は非常に指向性が高い。したがって、Fig.16 のような直線状のビームモデルとする。このとき、時刻 $t = k$ における (i, j) 番目のセルの占有度 $p_{k,i,j}$ は以下の式で更新される。

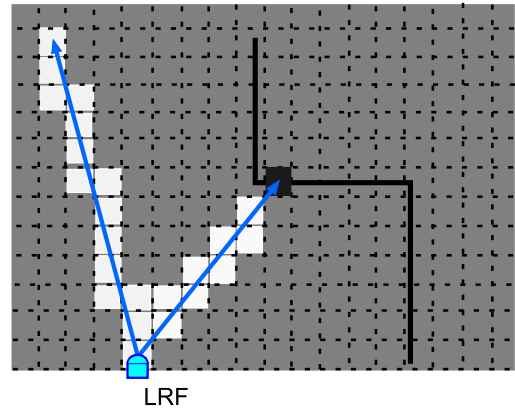


Fig.16 LRF のビームモデル

$$p_{k,i,j} = p_{k-1,i,j} + \frac{\rho_{k,i,j}}{1 - \rho_{k,i,j}} + \frac{p_0}{1 - p_0} \quad (23)$$

本研究では、 $p_0 = 0.5$ とすることでの右辺第 3 項を 0 とし、セルの更新における初期値の影響を取り除く。各セルの占有度は $\rho_{i,j}$ が 0.5 より小さければ占有度が減少し、0.5 より大きくなれば占有度が増加する。

センサモデルは LRF の測定点及び、LRF と測定点とを結ぶ線分上のセルにのみ適用する。したがって、それ以外のセルについては占有度の更新処理を適用しない。さらに、占有度に対して「空」、「未知」、「占有」の 3 種類のラベルをつけて区別する。具体的には占有度 0.25 以下で「空」、0.25 ~ 0.75 のとき「未知」、0.75 以上のとき「占有」とラベル付けする。各しきい値については経験的に設定した。また、各セルの大きさは一般に精度と処理時間とのトレードオフの関係にある。本研究では経験的にセルの大きさを $4\text{cm} \times 4\text{cm}$ の正方形とした。Fig.14 のデータに対して占有格子地図を生成した結果を Fig.17 に示す。図において、占有度は色の濃淡で表される。黒い部分ほど占有されている確率が高い。図より、確かに、廊下の壁などの障害物がある部分については黒く、走行可能な場所は白くなっていることがわかる。

4. 実験

4.1 特徴点抽出アルゴリズムの評価

2.2.1 で述べた特徴点抽出アルゴリズムの各しきい値を決定するため、様々な環境における特徴点抽出実験を行った。LRF の設定は視野角度 180° 、角



Fig.17 占有格子地図生成結果例

度分解能 0.5° ，最大測定距離 8m とし，床に置いて計測した．実験は鳥取大学電気電子工学科棟4Fの3箇所を計測し，得られたデータに対してしきい値を様々に変えて特徴点抽出アルゴリズムを適用した．評価の指標には，適合率 (P)，再現率 (R) 及び F 値を用いる．適合率は過検出に対する指標を表し，再現率は欠落に対する指標を表す． F 値はそれらの調和平均で表され， F 値が高いほど精度が高い．三つの値は以下の式で表される．

$$P = \frac{\text{抽出された正しいデータ数}}{\text{抽出された全データ数}} \quad (24)$$

$$R = \frac{\text{抽出された正しいデータ数}}{\text{抽出すべき正しいデータ数}} \quad (25)$$

$$F = \frac{2PR}{P + R} \quad (26)$$

4.1.1 角検出アルゴリズム

初めに，角検出アルゴリズムの評価を行った．廊下の角検出において，しきい値 a を $60\text{mm} \sim 100\text{mm}$ の範囲とし 10mm 間隔で計測した．また， α_{th} を $15^\circ \sim 40^\circ$ まで 5° 間隔に変更し，適合率と再現率，及び F 値を求め，最適なしきい値を決定した． F 値をグラフにしたものを Fig.18 に示す．グラフより，しきい値 α_{th} が大きくなれば F 値が増加している傾向がわかる．また，点間隔 a については増加するに従い，若干ではあるが減少傾向にあることがわかった．実験より，点間隔 $a = 60\text{mm}$ ，しきい値 $\alpha_{th} = 35^\circ$ のとき最大値 0.822 を得た．

4.1.2 不連続点検出アルゴリズムの評価

不連続点検出において，しきい値 d_{th} を $100\text{mm} \sim 250\text{mm}$ までの範囲とし 10mm 間隔で変化させ，適合率と再現率， F 値を求め，最適な d_{th} を決定した．なお，この実験では述べた不連続点検出の改良を行っていない．Fig.19 に F 値をグラフにしたも

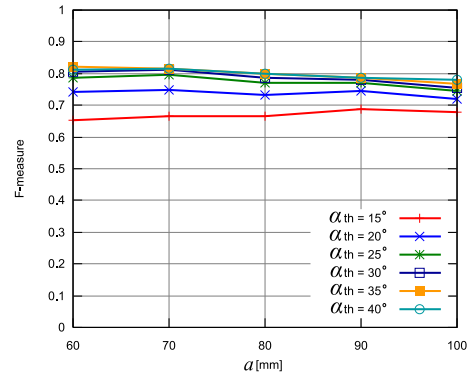


Fig.18 廊下の角検出結果

のを示す．図より d_{th} の増加にしたがって F 値が増加していることがわかる．ここで， $d_{th} = 100\text{mm}$ のとき再現率が 1.00 となっている．そこで，本研究における不連続点アルゴリズムは再現率を重視し， $d_{th} = 100\text{mm}$ を最適とし，適合率の改善を試みる．

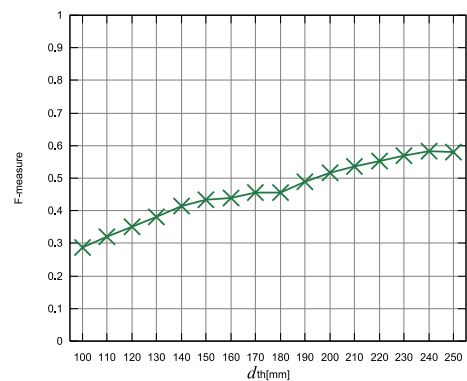


Fig.19 不連続点の検出結果

4.2 EKF-SLAM の評価実験

EKF-SLAM の性能を評価するため走行実験を行った．Fig.20 に示す環境において車いすをジョイスティックにより走行させ，EKF-SLAM を適用して自己位置推定，占有格子地図の生成を試みた．コース C は直進約 13m のコース，コース D は左旋回を含めた約 15m のコースである．車いすの走行速度は 1km/h である．両コースとも 5 回ずつ走行させ，自己位置推定結果と実際の位置とのずれを比較する．比較対象として，EKF-SLAM を適用した場合と，エンコーダ信号から得られるオドメトリのみを用いた場合とを比較した．ここで，グローバル座標は車いすの初期位置を原点，姿勢と

平行に y 軸，右手方向に x 軸とする．したがって，コース C ではグローバル座標が Fig.20 中の破線の xy 座標系となり，コース D においては中の点線の $x'y'$ 座標系となる．

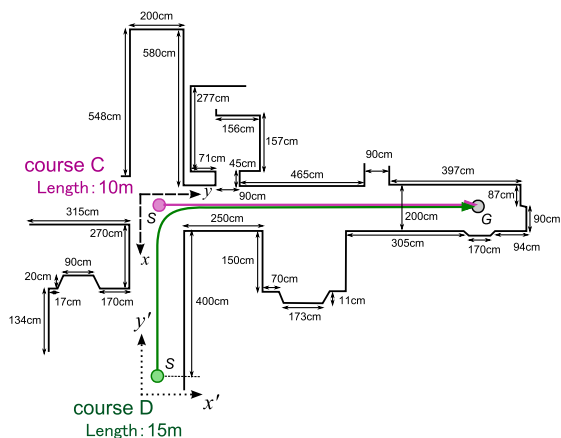


Fig.20 走行実験環境

Table2, Table3 にそれぞれのコースでの比較結果を示す．また，Fig.21, Fig.22 に走行時に作成した占有格子地図を示す．コース C においてはオドメトリのみと比べて推定結果が改善しているが差はあまり見られない．これはコース C は直進のみのコースであり，単純な動きであるために差が表れなかったものと思われる．3 回目の姿勢角推定結果の誤差が他と比べて高い．このとき，生成した占有格子地図は GOAL 地点付近の地図が歪んでいることがわかる．これは GOAL 地点付近では得られる特徴点の数が少なく，特徴点間の対応付け処理時に対応関係を誤ったためと考えられる．

コース D は右旋回動作が含まれたコースである．動作が増え，走行距離も長くなっているため，オドメトリのみでの推定結果は真値とのずれが大きくなっている．しかし，EKF-SLAM の結果については，ずれがコース C と同程度であることがわかる．また占有格子地図も作成できていることが確認できた．

5. まとめ

本章では，LRF のデータに EKF-SLAM を適用することで車いすの自己位置及び周囲環境を推定し，占有格子地図として表現する手法を検討，実装した．本研究では LRF のデータから環境中の幾何形状に注目して，廊下の角を抽出し，これをランドマークとすることで，EKF-SLAM を適用する．さ

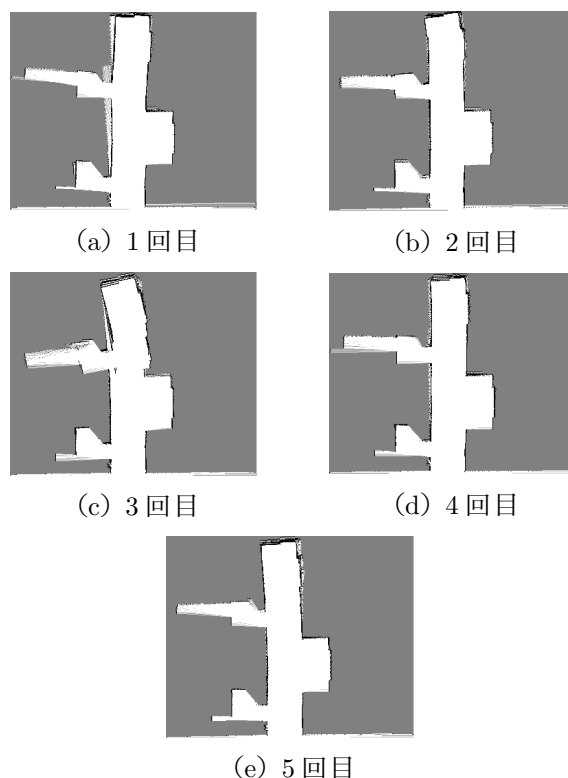


Fig.21 占有格子地図生成結果 (コース C)

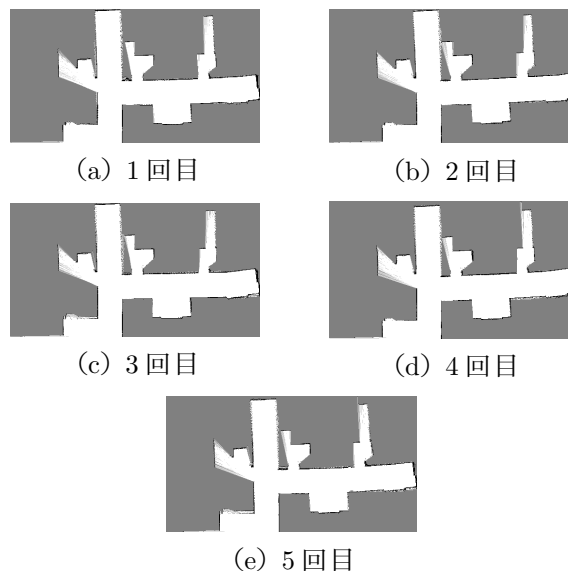


Fig.22 占有格子地図生成結果 (コース D)

Table 2 自己位置推定比較結果 (コースC)

回数	実際の位置, 姿勢			EKF-SLAM			Odometry		
	x [cm]	y [cm]	θ [°]	x_{error} [cm]	y_{error} [cm]	θ_{error} [°]	x_{error} [cm]	y_{error} [cm]	θ_{error} [°]
1	-23.5	1046	96	1.86	-12.65	0.52	2.92	-24.28	-2.57
2	-6	1030	92	-9.04	-10.80	5.31	10.87	-19.68	-2.47
3	-15.5	1007.5	87	-6.38	-15.10	16.27	0.23	-22.29	2.53
4	-13.5	1013	89	13.90	-16.78	2.24	4.20	-23.40	0.06
5	-14.5	1028	90	3.29	-16.26	2.31	13.43	-21.29	0.78
6	-1	1050.5	85	-8.46	-17.71	2.10	10.19	-23.74	-2.96

Table 3 自己位置推定比較結果 (コースD)

回数	実際の位置, 姿勢			EKF-SLAM			Odometry		
	x [cm]	y [cm]	θ [°]	x_{error} [cm]	y_{error} [cm]	θ_{error} [°]	x_{error} [cm]	y_{error} [cm]	θ_{error} [°]
1	1036.5	508	-1	-2.78	18.89	5.74	-10.11	-79.87	-5.87
2	1005.5	484	-3	-3.82	26.14	8.78	-6.67	-48.48	-2.16
3	998	502.5	3	-5.55	13.13	4.63	-10.19	-28.80	-3.16
4	968	515	1	-6.65	31.95	5.60	-6.31	-44.94	-3.50
5	981	504.5	-1	-6.40	28.27	4.69	-6.57	-31.22	-2.13

らに, 得られた車いすの自己位置から LRF のデータを占有格子地図上に反映する.

走行実験を通して, EKF-SLAM の有効性が確認でき, また, 占有格子地図についても正しく周囲環境を表現できることを確認した. 本手法により, システムは占有格子地図を通して周囲環境を正確に把握することが可能となる. したがって, 占有格子地図を用いて, 障害物回避など, 車いすの走行制御を行うことができると考えられる.

本手法では, LRF のスキャンデータから特徴点として廊下の角を抽出している. しかし, 平たい壁が続く廊下などを走行する場合, 特徴点が抽出できないため, 自己位置や周囲環境の推定精度が悪くなることが考えられる. したがって, 今後の課題として抽出する特徴点の検討などがあげられる.

6. 謝辞

この論文は当研究室の齊藤剛史助教 (現九工大准教授), 尾崎知幸技術専門職員との共同研究を筆者がまとめたものである. ここに心より感謝致します. 具体的な実験ならびにデータ処理に関しては当研究室の院生村井彰君の修士論文に負うところが多くここに礼申しあげます. 最後に本研究の一部は科学研究費基盤研究 (C) (21500515) の助成を受けたことについて礼申しあげます.

参考文献

- 1) 橋場参生, 中島康博, 音声操作型電動車椅子の開発と評価, 信学技報, WIT2002-66, pp.29-34, (2002).
- 2) 高強, 西原主計, 音声制御の車椅子, 信学技報, SSS2003-1, pp.1-4, (2003).
- 3) 小宮加容子, 中島康博, 橋場参生, 景川耕宇, 黒須 顕二, 狭い空間における音声指令による電動車いす走行テスト, 日本機械学会論文集 (C 編), Vol.69, No.688, pp.210-217, (2003).
- 4) G. Pires, U. Nunes, A wheelchair steered through voice commands and assisted by a reactive fuzzy-logic controller, Journal of Intelligent and Robotic Systems, Vol.34, No.3, pp.301-314, (2002).
- 5) L.M.Bergasa, M.Mazo, A.Gardel, R.Barea, L.Boquete, Commands generation by face movements applied to the guidance of a wheelchair for handicapped people, International Conference on Pattern Recognition, Vol.4, No.12, pp.4660-4663, (2000).
- 6) 後藤 健志, レーザレンジファインダを用いた電動車椅子の操縦補助システム, 奈良先端科学技術大学院大学 情報科学研究科 情報システム学専攻, 修士論文, (2002).

- 7) 石和 淳子, 富沢 哲雄, 大矢 晃久, 油田 信一, 測域センサを用いた移動ロボットの狭空間走行, 第6回計測自動制御学会システムインテグレーション部門講演会, Vol.2D3-1”, pp.587-588, (2005).
- 8) Ohnishi, N and Imiya, A, Corridor navigation and obstacle avoidance using visual potential field, In Canadian Conference on Computer and Robot Vision, pp.131-138, (2007).
- 9) 齊藤 豊, 安 弘, 吉野 貴光, 柏本 大輔, 益田 智, 勝又 幹夫, 岡本 孝一, 狭路走行を含む未知環境での車輪型移動ロボットの障害物回避法, ロボティクス・メカトロニクス講演会論文集, Vol.2000,pp.101, (2000).
- 10) Denis Wolf and Gaurav S. Sukhatme, Online Simultaneous Localization and Mapping in Dynamic Environments, Proc. of IEEE International Conference on Robotics and Automation, (2004).
- 11) 三澤 正志, 吉田 智章, 油田 信一, 自律的帰還機能を持つ荷物運搬用電動台車の開発, 日本ロボット学会誌, Vol.25, No.8, pp.55-62, (2007).
- 12) Young-Ho Choi and Tae-Kyeong Lee and Se-Young Oh, A line feature based SLAM with low grade range sensors using geometric constraints and active exploration for mobile robot, Autonomous Robots, Vol.24, pp.13-27, (2008).

(受理 平成22年10月29日)