

移動ロボットのための
自由視野と半自律制御を用いた遠隔操作手法

筑波大学
図書館情報メディア研究科
2020年3月
降籟 直人

目次

第 1 章	序論	1
1.1	本研究の背景	1
1.2	本研究の目的	2
1.3	本研究の構成	2
第 2 章	関連研究	3
2.1	遠隔操作における映像提示	3
2.2	遠隔操作における半自律制御	5
2.3	本研究の提案	5
第 3 章	システム構成	6
3.1	ハードウェア構成	6
3.1.1	使用機器	7
3.1.2	移動ロボット	7
3.1.3	カメラ	7
3.1.4	Laser Range Finder(LRF)	8
3.1.5	ゲームパッド	9
3.2	ソフトウェア構成	9
3.2.1	Robot Operating System(ROS)	9
3.2.2	その他のライブラリ	10
3.3	システム全体の構成	10
3.3.1	視野画像生成	11
3.3.2	自動障害物回避	11
3.3.3	SLAM	11
3.3.4	costmap 生成	11
3.3.5	操作者への画像表示	12
第 4 章	水平 360 度自由視野に基づくロボット制御	13
4.1	水平 360 度自由視野を用いた移動ロボットの制御	13
4.1.1	水平 360 度画像生成	13
4.1.2	視野方向へ進む移動ロボットの制御	15
4.1.3	視野方向の補正	15
第 5 章	水平 360 度自由視野に基づく半自律制御	17
5.1	半自律制御	17
5.2	Dynamic Window Approach(DWA)	17
5.2.1	概要	17
5.2.2	Dynamic Window の生成	19
5.2.3	軌跡の生成	19

5.2.4	軌跡の評価	20
第6章	評価実験	25
6.1	実験概要	25
6.2	結果	27
6.2.1	完走時間と衝突回数	27
6.2.2	アンケート	30
第7章	考察	38
第8章	結論	40
	参考文献	42

目 次

1.1	遠隔操作の概要図	1
2.1	操作者への提示画像 ([6] より引用)	3
2.2	操作者への提示画像 (3人称視点) ([3] より引用)	4
2.3	操作者への提示画像 (自由視点) ([3] より引用)	4
2.4	操作者への提示画像 ([3] より引用)	4
3.1	本研究で使用したロボット	6
3.2	ロボット上面図	6
3.3	ロボット側面図	6
3.4	PIONEER-3DX	8
3.5	PIONEER-3DX 主要諸元	8
3.6	SJ5000	8
3.7	UTM-30LX	8
3.8	UTM-30LX 主要諸元	8
3.9	URG-04LX-UG01	9
3.10	URG-04LX-UG01 主要諸元	9
3.11	F310	9
3.12	システム概要	10
3.13	作成したコースと SLAM によって作成した環境地図	11
3.14	costmap	12
3.15	操作者への提示画像	12
4.1	カメラ配置	13
4.2	水平 360 度画像生成の流れ	14
4.3	水平 360 度画像	15
4.4	自由視野を利用したロボット制御	15
4.5	移動ロボットとカメラシステムの座標軸	16
5.1	大局的経路計画	17
5.2	DWA の処理の流れ	18
5.3	Dynamic Window	19
5.4	生成された移動ロボットの軌跡	20
5.5	heading	21
5.6	(v_r, ω_r) と (v, ω)	22
5.7	$(v_r, \omega_r) = (-1, -1)$ の時の \cos 類似度	23
5.8	costmap	24
5.9	コスト値の付加方法	24

6.1	実験に使用したコースの見取り図	26
6.2	実験に使用したコースの写真1	26
6.3	実験に使用したコースの写真2	27
6.4	「遠隔操作手法によるコースの完走時間の平均」に関する評価の結果	28
6.5	「遠隔操作手法による衝突回数の平均」に関する評価の結果	29
6.6	直感的（感覚的）に操作しやすいと感じた	31
6.7	たくさんの練習が必要だと感じた	32
6.8	多くの人簡単に操作を行うことができると感じた	33
6.9	ロボットの操作としてふさわしいと感じた	34
6.10	ロボットの向きや動き方が気になると感じた	35
6.11	自動障害物回避はあったほうが良いと感じた	36
6.12	動きが滑らかだったと感じた	36
6.13	狭い空間でも快適に操作できた	36
7.1	自由視野+ポテンシャル法を用いた半自律制御手法の軌跡	39
7.2	提案手法の軌跡	39

第1章 序論

1.1 本研究の背景

移動ロボットを離れた場所から遠隔操作することで、人間が存在することが危険な災害現場の情報の取得や、行くことが困難な地域の人々と交流することが可能になる。そのような利点から遠隔操作移動ロボットの研究開発が行われてきた [1]。移動ロボットを遠隔操作する際、操作者は移動ロボットの周囲の環境を把握した上で、移動ロボットへ指令値を送る必要がある。しかし、操作者自身が遠隔地にある移動ロボットの周囲環境を把握することは困難なため、移動ロボットに搭載したカメラや Laser Range Finder (LRF) などのセンサ類を用いて、環境情報を操作者に提示する必要がある。

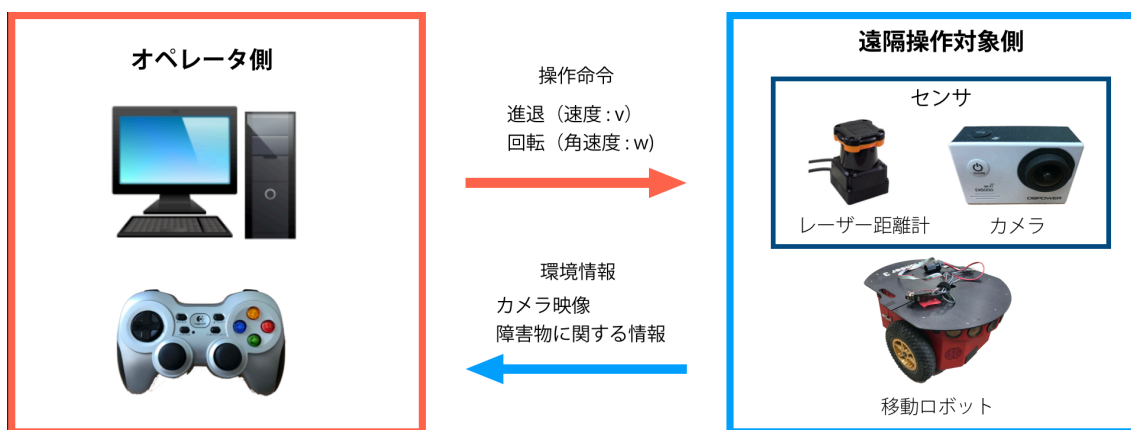


図 1.1: 遠隔操作の概要図

操作者が移動ロボット周辺の環境を把握しやすくするような情報提示手法やそれに基づく移動ロボットの制御手法が研究されてきた。人間が環境を認識する際、視覚から得る情報が多いため、移動ロボット周囲の視覚情報の取得方法について、広角カメラを用いた手法、複数のカメラを組み合わせた手法が提案されてきた。操作者が得る情報が増えると、移動ロボットの進行方向と操作者が注意を向けている方向が一致しない状況が起こり、障害物と衝突する危険性が増す。その課題に対して、移動ロボットを俯瞰で捉える視点を提示する手法 [2][3] や視野方向とロボットの移動方向を一致させる手法が提案された [4]。

遠隔操作ロボットの制御手法は自律制御と手動制御の2つに分けることができる。自律制御は、予め用意された環境地図から目標地点を指定することにより大局的な経路が生成され、加えてロボットに搭載したセンサ類の情報を元に局所的な経路を生成することで障害物の回避を行いつつ、目標地点まで制御することが可能である。自律制御は操作者の負担が小さいが、センサ類が対応できない状況が生じた場合、目標地点まで到達できない可能性がある。また人間共生環境において、事故が生じた場合の責任の所在は未だ議論の途

中である [5]. 手動制御においては, 操作者が移動ロボット周囲の環境を認識し, 制御するため, 柔軟に状況に対応することが可能である. しかし, 操作者が移動ロボット周囲の環境を正確に把握しつつ, 的確に指令値を送ることは難しく, 負担の大きいタスクである.

1.2 本研究の目的

本研究の目的は, 水平 360 度自由視野に基づく手動制御手法に, 自律制御手法を組み合わせることにより, 操作者の負担を減らした半自律遠隔操作の制御方法を実装することである. 水平 360 度自由視野に基づく手動制御手法により, 操作者は移動ロボットの姿勢を意識することなく, 視野方向とロボットの移動方向が一致した状態で遠隔操作が可能になるので, 表示画像に中止することが可能になる. また自律制御手法として, Dynamic Window Approach を用いて半自律制御を実現することにより, 操作者の指令値を元にした障害物回避を実現した.

1.3 本研究の構成

本論文の構成は以下のようにになっている. 第 2 章では遠隔操作ロボットについての関連研究について述べ, 第 3 章では本研究で実装したシステムの全体概要やハードウェア構成, ソフトウェア構成について述べる. 第 4 章では水平 360 度自由視野に基づく移動ロボットの制御方法について述べた後, 第 5 章において前述の制御方法に自動障害物回避を加えた半自律制御手法の説明を行う. 第 6 章では提案手法についての実験についての説明と結果を述べる. 第 7 章で考察を述べる. 第 8 章では結論と今後の課題について述べる.

第2章 関連研究

2.1 遠隔操作における映像提示

操作者の移動ロボットの周辺環境の状況認識に視覚情報は大きく影響を与えるため，カメラを移動ロボットに搭載して，周辺の環境情報を操作者に提示する手法はこれまで多く研究されてきた．最も単純な単眼カメラを用いた遠隔操作の手法では，視野が狭く得られる情報が少ない問題（keyhole問題）[6]が生じる．視野を広げるために，広角カメラを用いた研究[7]や複数のカメラを用いた遠隔操作ロボット[1][6]が開発された．しかし，前者は表示画像に歪みが生じ，操作者が酔いやすくなったり，移動している物体の速度が実際よりも速く感じてしまうという問題があった．後者は操作者の注意が一つのカメラからの映像に向けられてしまい，他のカメラからの映像に注意を向けることが難しくなる cognitive tunneling 問題がある[8]．



図 2.1: 操作者への提示画像 ([6] より引用)

そこで，カメラの動きとロボットの動きを分離する手法が提案された．この手法は，両者の向きが一致していない時，ロボットの移動方向の状況認識ができず，障害物との衝突の可能性がある．この問題に対して，3次元の仮想視点を生成しロボットの動きとカメラの動きの両方を俯瞰して捉える手法[2][3]と，ヘッドマウントディスプレイを用いて，操作者が向いた方向とカメラの動きを同期させる手法が提案された．

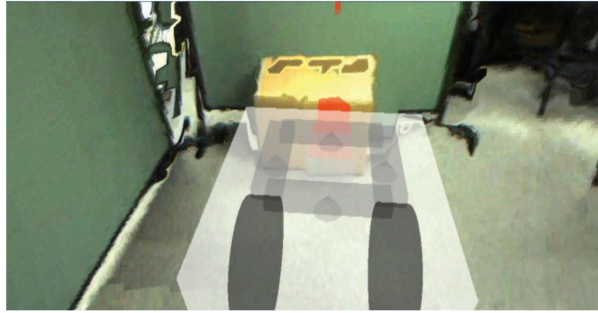


図 2.2: 操作者への提示画像（3人称視点）（[3]より引用）



図 2.3: 操作者への提示画像（自由視点）（[3]より引用）

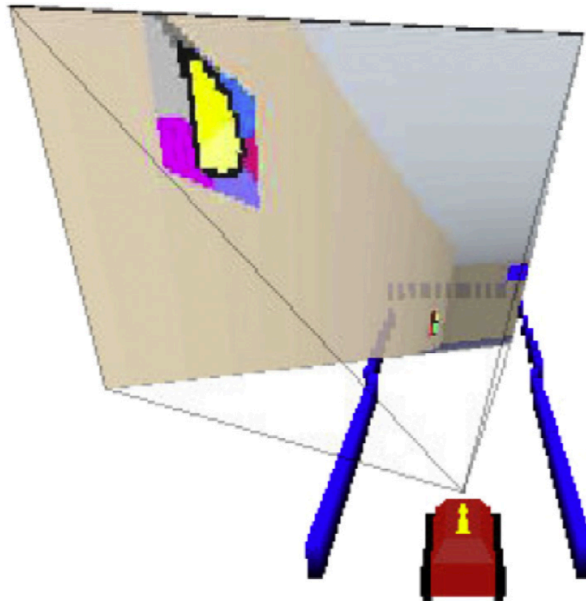


図 2.4: 操作者への提示画像（[3]より引用）

前者の手法は3人称視点を生成するために深度カメラをしようするため操作者へ提示する映像の画質が落ちてしまい、その処理によりタイムラグが生じるという問題点が存在する。後者の手法はヘッドマウントディスプレイによる酔いが生じるという問題や操作者が用意すべき機器が多いという問題が存在する。カメラの動きとロボットの動きを分離させ

た上で、カメラの向きとロボットの動きを一致させた手法が提案された [4]。この手法では、水平 360 度自由視野に基づく制御により、操作者はロボットの姿勢を意識することなく、視野方向を動かすだけで移動ロボットの制御が可能になり、操作性が向上した。しかし、操作者は移動ロボットの大きさや姿勢を把握できないので、障害物に衝突する恐れがあるため、半自律制御を行う必要がある。

2.2 遠隔操作における半自律制御

遠隔操作移動ロボットの制御において、操作者の入力による手動制御に加えて、移動ロボットに搭載したセンサ類を利用して自律制御を行うことは操作者の負担を減らすことが可能である [9][10]。ゲームパッドによる速度の制御入力にポテンシャル法 [11] を用いた自律制御を組み合わせた手法 [4] が提案されている。しかし、この手法では手動制御と半自律制御の間に衝突回数に関して有意差が認められなかった。ポテンシャル法が障害物の位置を元に計算された斥力ベクトルと操作者の入力によって計算されたベクトルを合成することで半自律制御を実現しており、移動ロボットの動力学を考慮していない制御を行っているからと考えられる。

2.3 本研究の提案

本研究では、水平 360 度自由視野に基づく制御手法を大館の手法 [4] を元の実装する。この手法では 4 台のカメラを円周上に配置し、それらから得られる画像を合成することで水平 360 度画像を生成し、操作者の入力によって切り取る範囲を決定することで自由視野を実現する。操作者が決定した視野方向が移動ロボットの進行方向となるように制御を行うことで、操作者はロボットの大きさや姿勢を意識せずに操作することが可能になるが、障害物に衝突しやすくなる。そのため、移動ロボットの動力学を考慮した自律制御手法である Dynamic Window Approach(DWA)[12] を元に、操作者の入力を考慮した処理を新たに加えた手法に改良する。

第3章 システム構成

3.1 ハードウェア構成

本研究で使用したハードウェアについて説明する。カメラ4台，レーザレンジファインダ（LRF）2台，ノートPCを搭載した移動ロボットの外観を図3.1～図3.3に示す。

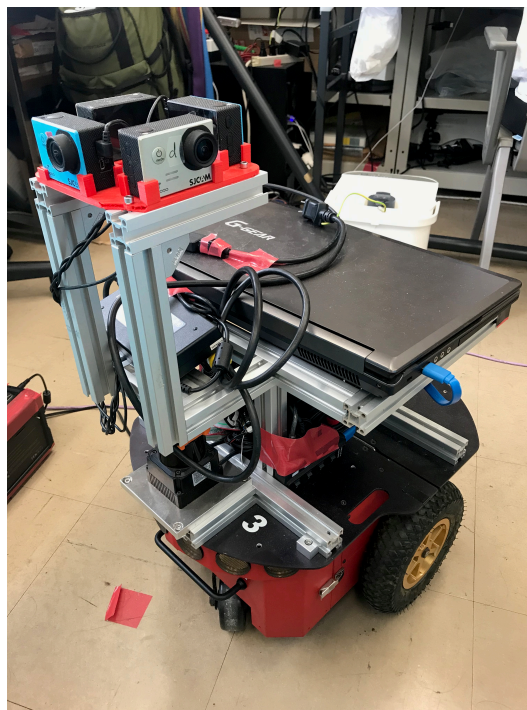


図 3.1: 本研究で使用したロボット



図 3.2: ロボット上面図



図 3.3: ロボット側面図

3.1.1 使用機器

本研究で使用した機器を表 3.1～表 3.3 に示す。本研究において、ノート PC は移動ロボットに搭載し、移動ロボット上のセンサ類の情報をサーバ PC に送信する。またサーバ PC はノート PC からのデータを受信し、画像処理や移動ロボットの制御計算を行う。

機器名	メーカー	モデル	台数
移動ロボット	MobileRobots	PIONEER-3DX	1
LRF	北陽電気株式会社	UTM-30LX	1
LRF	北陽電気株式会社	URG-04LX-UG01	1
カメラ	SJCAM	SJ5000	4
ゲームパッド	Logicool	F310	1

表 3.1: 使用機器

パーツ	仕様
OS	Ubuntu16.04LTS
CPU	Intel Corei7-4700MQ 2.40GHzx8
メモリ	8GB
GPU	GeForce GTX 765M 2GB

表 3.2: ロボットに搭載したノート PC

パーツ	仕様
OS	Ubuntu16.04LTS
CPU	Intel Corei9-9900K @ 3.60GHzx16
メモリ	16GB
GPU	GeForce GTX 1650

表 3.3: サーバ PC

3.1.2 移動ロボット

移動ロボットは MobileRobots 製の PIONEER-3DX を使用した。この移動ロボットは対向二輪移動ロボットであり、非ホロノミックの特性を持つ。移動ロボットに与える制御入力として、並進速度と角速度を用いた。外観を図 3.4、主要諸元を図 3.5 に示す。

3.1.3 カメラ

遠隔地の映像を撮影するために用いたカメラを図 3.6 に示す。画角は 170[deg] である。本研究ではこのカメラを 4 台使用し、水平 360 度画像を生成した。

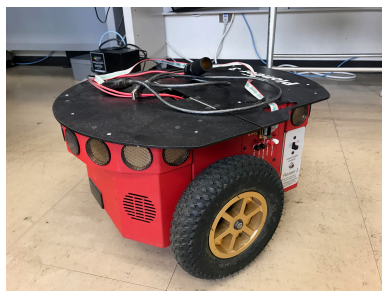


図 3.4: PIONEER-3DX

品名	測域センサ
型式	PIONEER-3DX
全長	0.44[m]
幅	0.38[m]
高さ	0.22[m]
最大並進速度	1.2[m/s]
最大回転速度	300[deg/s]

図 3.5: PIONEER-3DX 主要諸元



図 3.6: SJ5000

3.1.4 Laser Range Finder(LRF)

ロボットの周囲の障害物情報を costmap として認識するため、また移動ロボットの自己位置推定を行うために、移動ロボットの前方と後方に1つずつ LRF を設置し、それぞれの LRF から得られた点群データを合成することで 360 度の点群データを取得した。使用した 2 つの LRF の外観と主要諸元をそれぞれ図 3.7～図 3.10 に示す。

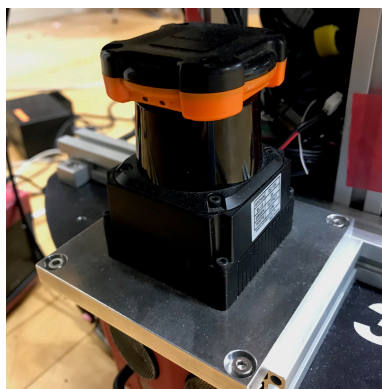


図 3.7: UTM-30LX

品名	測域センサ
型式	UTM-30LX
検出距離	0.1 30[m]
走査角度	270[deg]
角度分解能	0.25[deg]
重量	210[g]

図 3.8: UTM-30LX 主要諸元

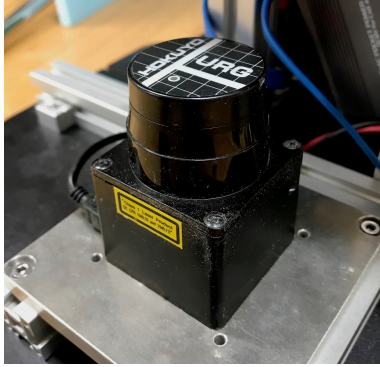


図 3.9: URG-04LX-UG01

品名	測域センサ
型式	URG-04LX-UG01
検出距離	0.02 5.6[m]
走査角度	240[deg]
角度分解能	0.36[deg]
重量	160[g]

図 3.10: URG-04LX-UG01 主要諸元

3.1.5 ゲームパッド

本研究では、移動ロボットを遠隔操作する際の操作者の入力デバイスとして、LogicoolのF310を使用した。図 3.11 にその外観を示す。



図 3.11: F310

3.2 ソフトウェア構成

本研究で実装したシステムのソフトウェア構成について述べる。

3.2.1 Robot Operating System(ROS)

Robot Operating System(ROS)[13]とは、ロボット用のオープンソースソフトウェアプラットフォームである。ROSはロボットアプリケーション作成を支援するライブラリとツールを提供しており、具体例として、デバイスドライバやライブラリ、視覚化ツール、メッセージ通信が提供されている。ROSでは、行われる一連の処理をノードという単位で実行している。複数のノードを同時に実行し、HTTPベースのXMLRPCプロトコルを用いて相互に通信することで分散処理を可能にしている。ノードは独立して実行できるため、再利用性が高く、オープンソースとして公開されたノードを自分のシステムへ組み込むことが容易である。

3.2.2 その他のライブラリ

本研究で使用したライブラリを表 3.4 に示す。

使用目的	ライブラリ	バージョン
画像処理	OpenCV(Open Source Computer Vision Library)	3.4.7
ロボット制御	Aria(MobileRobots Advanced Robotics Interface for Applications)	2.9.1

表 3.4: 使用ライブラリ一覧

OpenCV は水平 360 度画像生成のために使用した。Aria はロボット制御のために ROS を通して使用した。

3.3 システム全体の構成

本研究で実装したシステムの概要を図 3.12 に示す。システムはロボット、サーバ、操作者の 3 つに分けることができ、ロボット側では、ロボットの内界センサと LRF, カメラなどのセンサデータをロボット上の PC から送信し、ロボットへの指令値を受信する役割を持つ。サーバでは、ロボット上の PC から送信された画像データを処理して、操作者に送信する役割やロボット上の LRF のデータや操作者の入力情報を元に最適なロボットへの指令値を計算する役割を持つ。操作者側では、サーバから送信された画像データを受信し、ディスプレイに表示し、ゲームパッドを通じて操作者の入力を受け取り、サーバに送信する役割を持つ。図 3.12 内の処理についての概略を以下に記述する。

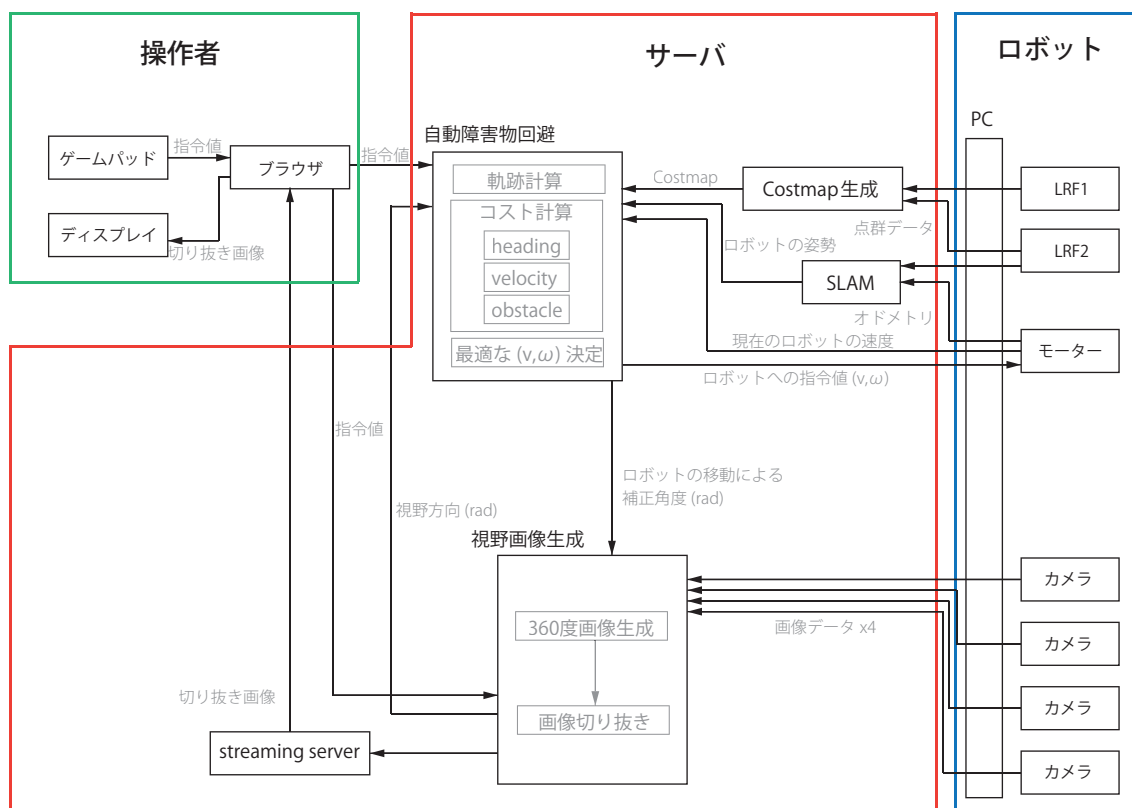


図 3.12: システム概要

3.3.1 視野画像生成

4台のカメラを用いて、水平360度画像を生成する。操作者はゲームパッドを用いて自由に視野方向を操作することができる。第4章で説明を行う。

3.3.2 自動障害物回避

第4章で実装した移動ロボットの制御方法により、操作者はロボットの姿勢や大きさを意識することなく遠隔操作を行うことが可能になったが、同時に視野方向以外に存在する障害物に衝突しやすくなった。そこで、障害物回避を制御に組み込むことで操作者の負担を減らした遠隔操作を実現した。第5章で説明を行う。

3.3.3 SLAM

自動障害物回避を行う際に、移動ロボットの自己位置を把握する必要がある。本研究では、SLAM(Simultaneous Localization And Mapping)を用いて自己位置推定を行った。SLAMとは、LRFなどの外界センサーから得る周囲の環境情報とロボットの内界センサから得たオドメトリ情報をもとに、ロボット周辺の環境地図作成とロボットの姿勢を同時に推定する技術のことである。本研究では、Brain Gerkeyらにより開発されたgmapping[14]を用いて、SLAMを実現した。



図 3.13: 作成したコースと SLAM によって作成した環境地図

3.3.4 costmap 生成

自動障害物回避を行う際に、周囲の障害物情報を得る必要がある。本研究では、LRFから得た点群データから costmap を生成することで障害物情報を取得した。第5章で説明を行う。

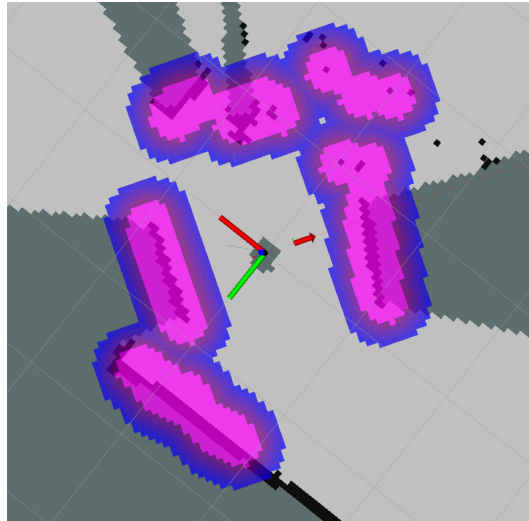


図 3.14: costmap

3.3.5 操作者への画像表示

本研究では，操作者は web ブラウザに表示されたロボットから送信された映像を見ながら，ゲームパッドで遠隔操作を行う．図 3.15 は，web ブラウザに表示されている画面である．ゲームパッドで入力されたデータは GamepadAPI[15] を利用することで web ブラウザが受信する．web ブラウザと ROS システム間の通信は WebSocket 通信を利用した．



図 3.15: 操作者への提示画像

第4章 水平360度自由視野に基づくロボット制御

4.1 水平360度自由視野を用いた移動ロボットの制御

本研究では，操作者に提供する視覚情報を増やし，移動ロボットの操作性を向上させるために，4台のカメラを用いた自由視野と操作者が選択した視野方向に移動ロボットが動く制御方法 [4] を実装した．

4台のカメラを設置し，隣り合うカメラから得られる画像をスティッチングすることで水平360度の自由視野を実現した．操作者はゲームパッドを用いて，360度の画像データの任意の方向を選択することができる．操作者が視野方向を決定したあと，ロボットに速度指令値を送信することで，ロボットが視野方向に移動する．ロボットの移動により，操作者が指定した視野方向が変化していくので，ロボットの移動量から補正角度を計算し，ロボットが移動しても同じ方向の画像を表示し続ける処理を行った．

360度の画像から任意の方向に進むことの出来るシステムにより，操作者はロボットの大きさや姿勢を意識することなく遠隔操作することが可能になる．

4.1.1 水平360度画像生成

水平360度画像を生成し，操作者が任意の方向の映像を取得するために，大舘 [4] の手法を用いた．この手法は図4.1のように4台のカメラを90[deg]間隔に配置し，隣り合うカメラから得られる画像をスティッチングすることで水平360度画像を生成する．全体の処理の流れと実行結果をそれぞれ図4.2と図4.3に示す．

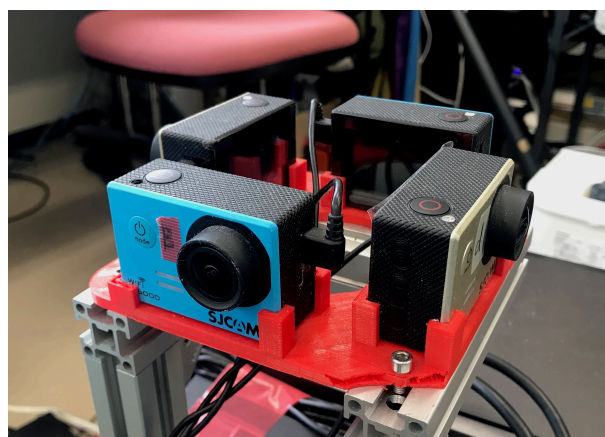


図 4.1: カメラ配置

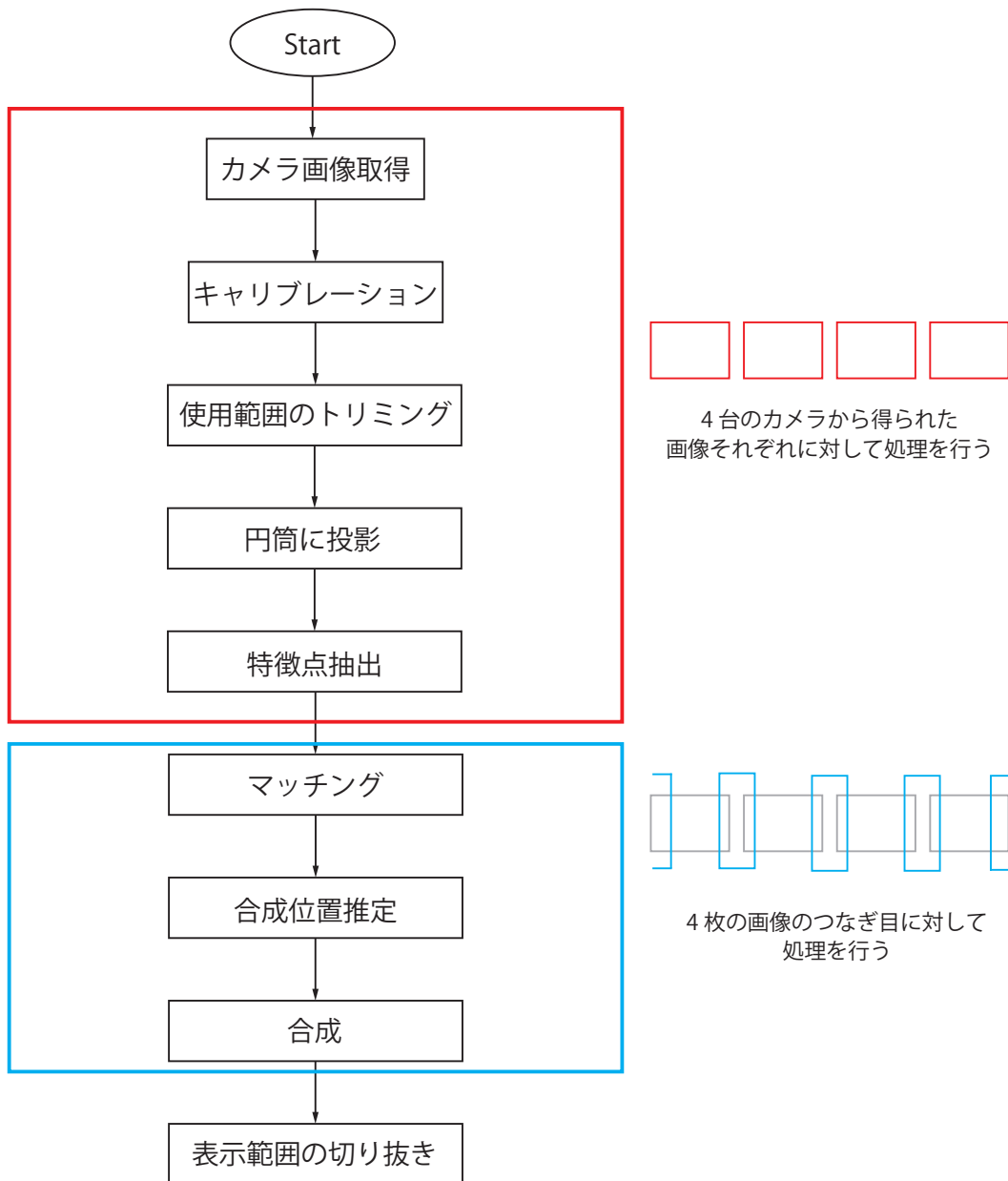


図 4.2: 水平 360 度画像生成の流れ



図 4.3: 水平 360 度画像

4.1.2 視野方向へ進む移動ロボットの制御

360 度の自由視野から操作者に表示した視界方向をロボットの進行方向とすることで、操作者は広い視野を得られ、ロボットの動きを意識することなく移動ロボットを操作することができる。本研究では、[4] の手法を用いた。この手法ではカメラシステムの位置を移動ロボットの回転中心から離れた位置にすることで非ホロミックな移動ロボットでも、カメラシステムの中心の点は全方位移動を行うことができる。

次に、カメラシステム座標系 Σ_c 上での操作者の入力と移動ロボットに与える制御入力の関係について述べる。移動ロボットは並進速度 v_r と角速度 ω_r を入力することで制御可能である。図 4.4 において、カメラシステムの中心と移動ロボットの回転中心の距離を d_c 、自由視野における操作者の視野方向を θ_c 、視野方向に進む速度を v_c とすると、 (v_r, ω_r) は式 (4.1) で求められる。

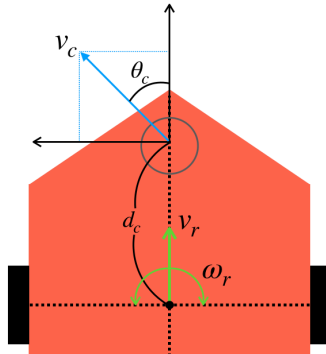


図 4.4: 自由視野を利用したロボット制御

$$\begin{pmatrix} v_r \\ \omega_r \end{pmatrix} = \begin{pmatrix} v_c \cos \theta_c \\ \frac{v_c \sin \theta_c}{d_c} \end{pmatrix} \quad (4.1)$$

4.1.3 視野方向の補正

4.1.2 で述べた制御手法により、操作者はロボットの姿勢を考慮しないことが可能になるが、視野方向にロボットの移動を制御する時、ロボットの回転によってカメラシステム座標系 Σ_c も回転することを考慮する必要がある。ワールド座標系 Σ_w 上のロボットの姿勢を ${}^w\theta_r$ 、ロボット座標系 Σ_r 上の視野方向を ${}^r\theta_c$ とすると、 Σ_w 上の視野方向 ${}^w\theta_c$ は式 (4.2) で求められる (図 4.5)。

$${}^w\theta_c = {}^w\theta_r + {}^r\theta_c \quad (4.2)$$

視野方向に直進する制御を行うためには、 ${}^w\theta_c$ を一定にする必要がある。 ${}^w\theta_r$ は SLAM によって推定された姿勢なので、 ${}^r\theta_c$ を補正する。時刻 t での ${}^r\theta_c$ を ${}^r\theta_c[t]$ 、時刻 $t+1$ での ${}^r\theta_c$ を ${}^r\theta_c[t+1]$ とすると、補正值 α を用いて式 (4.3) が成り立つ。ただし、 α は時刻 t での ${}^w\theta_r$ を ${}^w\theta_r[t]$ 、時刻 $t+1$ での ${}^w\theta_r$ を ${}^w\theta_r[t+1]$ とすると、式 (4.4) で求められる。

$${}^r\theta_c[t+1] = {}^r\theta_c[t] + \alpha \quad (4.3)$$

$$\alpha = {}^w\theta_r[t] - {}^w\theta_r[t+1] \quad (4.4)$$

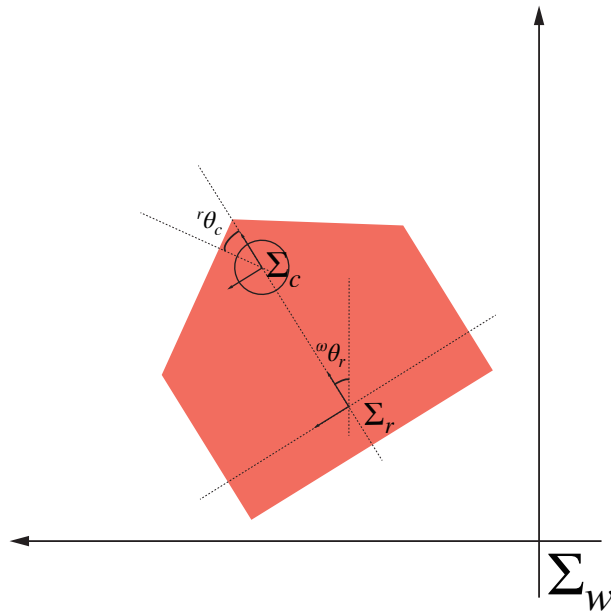


図 4.5: 移動ロボットとカメラシステムの座標軸

第5章 水平360度自由視野に基づく半自律制御

5.1 半自律制御

第4章で述べた360度の視野と視野方向に進む遠隔操作によって、操作者はロボットの姿勢や動きを意識することなく操作することが可能になる。しかし、操作者はロボットの動きを把握できないため障害物との衝突を避けることが困難である。そこで、操作者の入力に Dynamic Window Approach[12] を組み合わせることで半自律遠隔操作を実現する。

5.2 Dynamic Window Approach(DWA)

移動ロボットを自律制御する際、大局的経路計画を実行し、ロボットの現在座標から目標座標までのロボットの動力学を考慮しない経路を生成し、ロボットに最も近い経路上の座標をサブゴールに設定する。サブゴールに対してロボットの動力学を考慮した局所的経路計画を行い、目標座標までサブゴールを更新し続けることで、目標座標まで自律制御を行う。

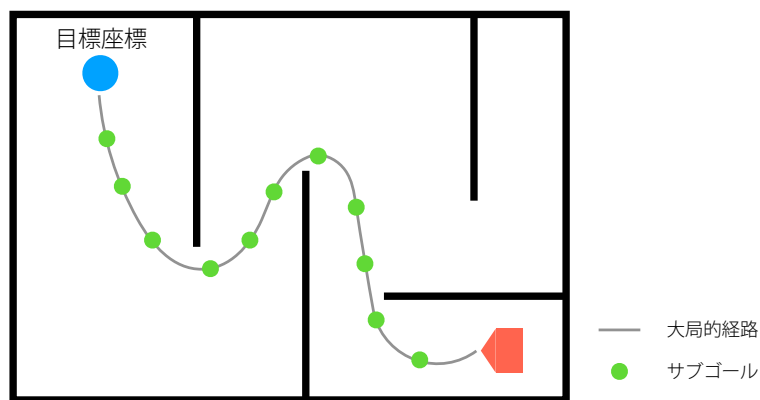


図 5.1: 大局的経路計画

本研究では、操作者の入力と自律制御を組み合わせるために、操作者の入力を利用して局所的経路計画を行うシステムを実装した。局所的経路計画の手法として、非ホロノミックなロボットにも適応でき、ロボットの動力学も考慮した Dynamic Window Approach (DWA) を利用した。

5.2.1 概要

本研究では、操作者の手動制御と自動障害物回避を組み合わせることで、操作者の負担を減らした遠隔操作を実現する。ロボットの現在座標と操作者の入力から目標座標を計算し、DWA を用いて、障害物回避を行ったロボットの動力学を考慮した目標座標までの経路

生成とロボットへの制御値の計算を行う。Dynamic Window Approachの全体の処理の流れを図5.2に示す。各処理の説明を以下に述べる。従来のDWAはロボットの車軸中心を基準に計算を行っている。しかし、本研究ではロボットの車軸中心から離れた位置にカメラシステムを設置することで操作者が全方位移動を実行可能にしている。カメラシステムを中心としたDWAの説明も重ねて行う。

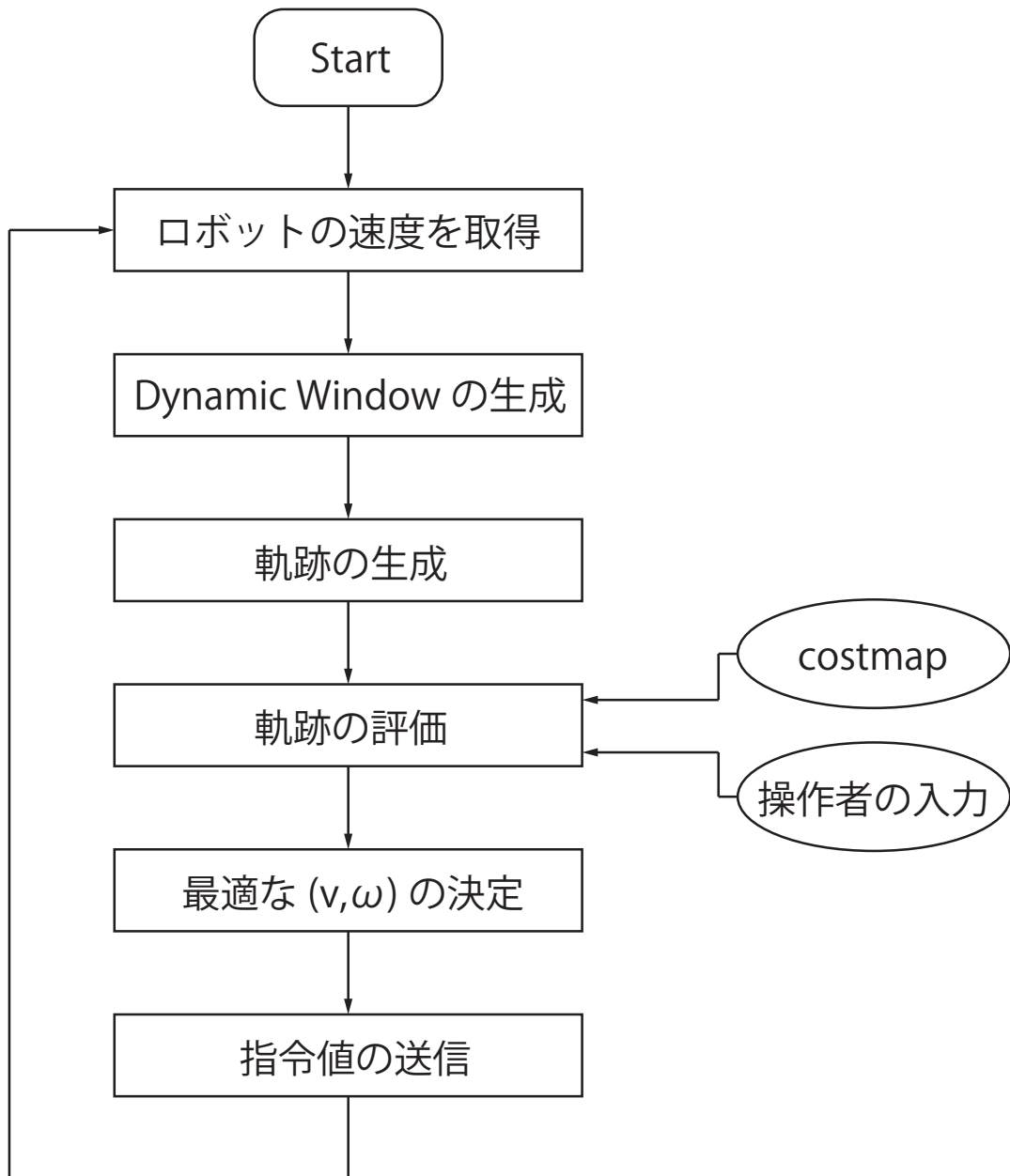


図 5.2: DWA の処理の流れ

5.2.2 Dynamic Window の生成

図 5.3 において，Dynamic Window は，ある時刻でのロボットの速度から処理が 1 ステップ進んだ後にロボットが到達しうる速度の範囲である．本研究で使用したロボットは，並進速度 [m/s]，角速度 [rad/s] を制御入力として受ける．

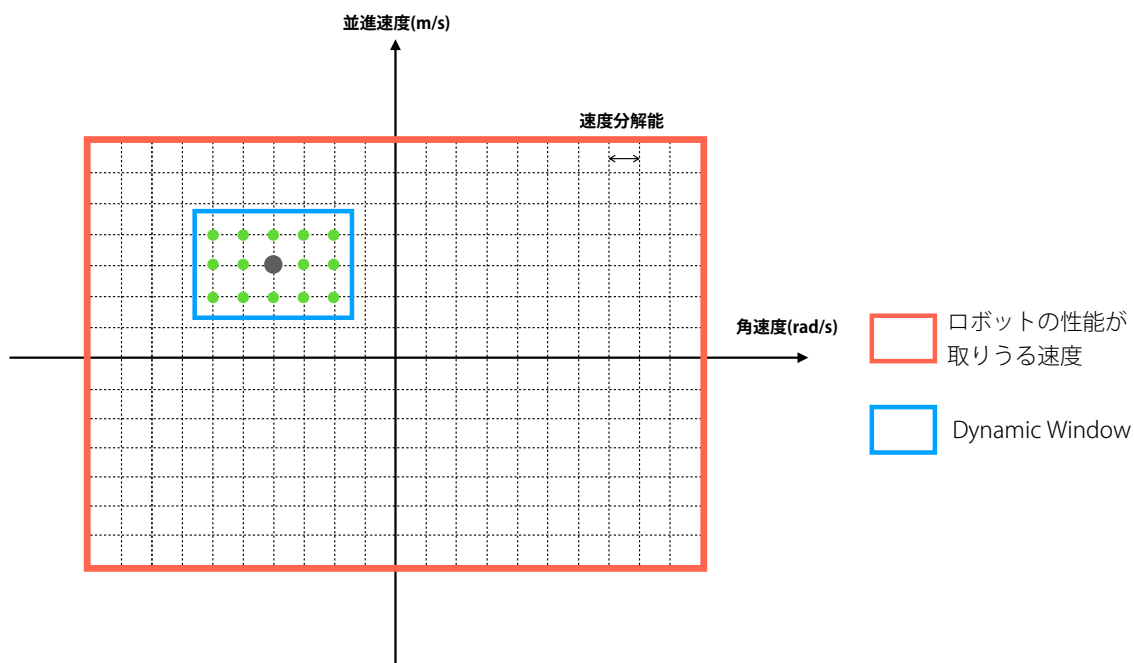


図 5.3: Dynamic Window

図 5.3 において，赤枠は移動ロボットが取りうる速度の最大値と最小値の範囲であり，青枠（Dynamic Window）は次のステップに移動ロボットが取りうる速度の範囲である．緑色の点は Dynamic Window 内の値を速度分解能によって離散化したものである．DWA は，Dynamic Window 内の速度の値（緑色の点）で微小時間，移動ロボットを制御したときの軌跡を計算し，評価を行う．DWA は，次ステップで移動ロボットが取りうる速度に対してのみ軌跡の計算を行うので，計算コストが抑えられ，移動ロボットの動力学を考慮した制御方法となる．

5.2.3 軌跡の生成

DWA では，Dynamic Window 内の速度で微小時間移動ロボットを制御したときの軌跡を計算する．本項では，時刻 t_0 と時刻 t_n における移動ロボットの位置と姿勢の変化について述べる．時刻 t におけるワールド座標系の移動ロボットの x 座標， y 座標，姿勢 $(x(t), y(t), \theta(t))$ は，時刻 t における並進速度，角速度をそれぞれ $v(t), \omega(t)$ とすると

$$x(t_n) = x(t_0) + \int_{t_0}^{t_n} v(t) \cdot \cos \theta(t) dt \quad (5.1)$$

$$y(t_n) = y(t_0) + \int_{t_0}^{t_n} v(t) \cdot \sin \theta(t) dt \quad (5.2)$$

(5.1)(5.2) で表せる. $v(t)$ は時刻 t_0 における移動ロボットの速度 $v(t_0)$ と並進加速度 $\dot{v}(\hat{t}) (\hat{t} \in [t_0, t])$ によって求められる. 同様に, $\theta(t)$ は時刻 t_0 における移動ロボットの姿勢 $\theta(t_0)$ と角加速度 $\dot{\omega}(\hat{t}) (\hat{t} \in [t_0, t])$ によって求められる. 以上より, (5.3)(5.4) を得る.

$$x(t_n) = x(t_0) + \int_{t_0}^{t_n} \left(v(t_0) + \int_{t_0}^t \dot{v}(\hat{t}) d\hat{t} \right) \cdot \cos \left(\theta(t_0) + \int_{t_0}^t \left(\omega(t_0) + \int_{t_0}^{\hat{t}} \dot{\omega}(\tilde{t}) d\tilde{t} \right) d\hat{t} \right) dt \quad (5.3)$$

$$y(t_n) = y(t_0) + \int_{t_0}^{t_n} \left(v(t_0) + \int_{t_0}^t \dot{v}(\hat{t}) d\hat{t} \right) \cdot \sin \left(\theta(t_0) + \int_{t_0}^t \left(\omega(t_0) + \int_{t_0}^{\hat{t}} \dot{\omega}(\tilde{t}) d\tilde{t} \right) d\hat{t} \right) dt \quad (5.4)$$

以上より, (5.3)(5.4) から, 移動ロボットの軌跡を求められる. 実装上, 1 ステップ間においては加速度, 速度は一定値を取ると仮定し, (5.3)(5.4) を離散化した (5.5)(5.6) を用いた. 計算の結果, 生成される軌跡は図 5.4 のように, 円弧を描く.

$$x[t+1] = x[t] + v[t] \cos \theta[t] \quad (5.5)$$

$$y[t+1] = y[t] + v[t] \sin \theta[t] \quad (5.6)$$

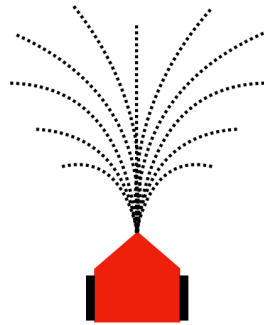


図 5.4: 生成された移動ロボットの軌跡

5.2.4 軌跡の評価

5.2.3 のように生成した軌跡は移動ロボットがこれから通ると想定される座標の集合である. 各軌跡とそれに対応する制御入力 (v, ω) に対して, (5.7) に示したコスト関数を用いて計算を行い, 最もコスト値が小さい制御入力を移動ロボットに指令値として与える.

$$G(v, \omega) = \alpha \times \text{heading}(v, \omega) + \beta \times \text{velocity}(v, \omega) + \gamma \times \text{obstacle}(v, \omega) \quad (5.7)$$

(5.7)において、 $\text{heading}(v, \omega)$, $\text{velocity}(v, \omega)$, $\text{obstacle}(v, \omega)$ の各項はそれぞれ、0から1の間の値を取り0に近いほど評価が高い。 α , β , γ はそれぞれの項のパラメータであり、大きい値をとるほどその項の影響が大きくなる。以下、コスト関数のそれぞれの項について説明を行う。

(a) heading

本研究では、 heading 項を改良することによって、操作者の入力を考慮したDWAを実現している。従来手法では、生成した軌跡の最終点において、移動ロボットがどれだけサブゴールに向いているかを評価する。しかし、半自律制御において操作者の入力はサブゴールの位置情報ではなく、移動ロボットの指令値である並進速度 v と角速度 ω である。よって、提案手法では操作者の入力 (v, ω) を元にコスト値の計算を行っている。以下で従来手法と提案手法の説明を行う。

● 従来手法 サブゴール情報からコスト値を計算

図5.5において、制御入力 (v, ω) によって生成された軌跡の最終点からその1ステップ前の点を結んだベクトル $\mathbf{d}_{\text{simulated}}$ と、軌跡の最終点からサブゴールまでのベクトル $\mathbf{d}_{\text{ideal}}$ のCOS類似度 $\text{COS}(\mathbf{d}_{\text{simulated}}, \mathbf{d}_{\text{ideal}})$ を計算する (5.8)。

$$\text{COS}(\mathbf{d}_{\text{simulated}}, \mathbf{d}_{\text{ideal}}) = \frac{\mathbf{d}_{\text{simulated}} \cdot \mathbf{d}_{\text{ideal}}}{|\mathbf{d}_{\text{simulated}}| |\mathbf{d}_{\text{ideal}}|} \quad (5.8)$$

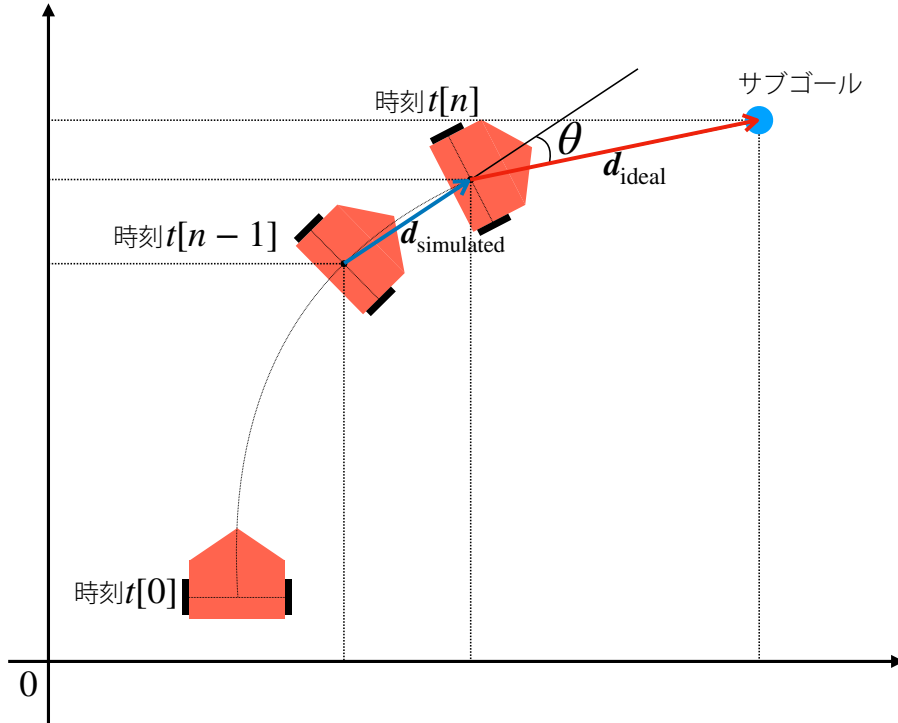


図 5.5: heading

COS類似度は移動ロボットがサブゴールに向かっている時、1になり、真逆に向い

ている時-1になるので，0から1の範囲に正規化する必要がある．よって求めるコスト値 $cost_{heading}$ は式 (5.9) で求められる．

$$cost_{heading} = \frac{-\cos(\mathbf{d}_{simulated}, \mathbf{d}_{ideal}) + 1}{2} \quad (5.9)$$

● 提案手法 操作者の入力 (v, ω) からコスト値を計算

上記の従来手法を操作者の入力と組み合わせて用いると，操作者の入力によってサブゴールの設定を行い，それに対して heading 項の計算を行うことになる．しかし，本研究において，操作者は4.1.2項で示したような速度の入力を移動ロボットに対して行う．そこで，本研究では， v と ω の2次元平面上において，4.1.2項で求めた $\mathbf{c}_{input} = (v_r, \omega_r)$ に対して，Dynamic Window 内の制御入力 $\mathbf{c}_{dw} = (v, \omega)$ との \cos 類似度 (5.10) をコスト値とすることで，操作者の理想の入力ベクトル (v_r, ω_r) との成す角が0度に近づくほど1に近づき，180度に近づくほど-1に近づく． $(v_r, \omega_r) = (-1, -1)$ のときの \cos 類似度の値を表したのが図 5.7 である．最終的に求める $cost_{heading}$ は式 (5.11) で求められる．

$$\cos(\mathbf{c}_{input}, \mathbf{c}_{dw}) = \frac{\mathbf{c}_{input} \cdot \mathbf{c}_{dw}}{|\mathbf{c}_{input}| |\mathbf{c}_{dw}|} \quad (5.10)$$

$$cost_{heading} = \frac{-\cos(\mathbf{c}_{input}, \mathbf{c}_{dw}) + 1}{2} \quad (5.11)$$

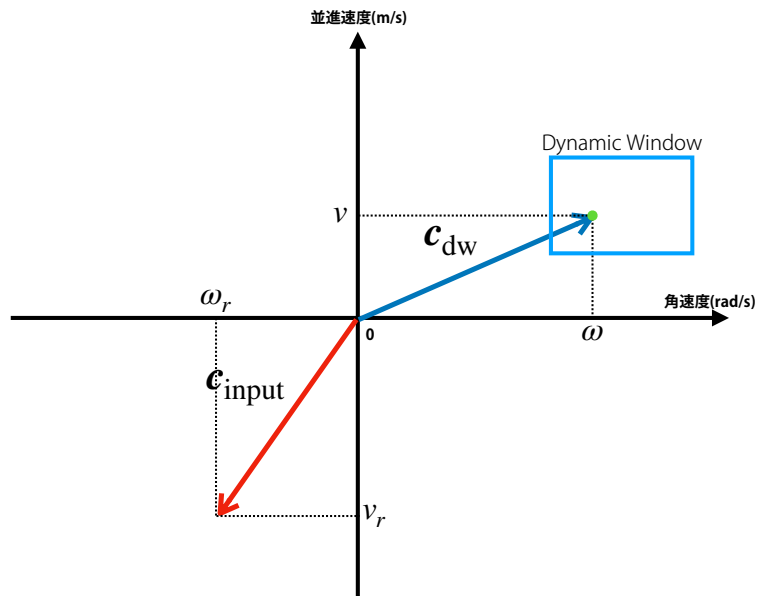


図 5.6: (v_r, ω_r) と (v, ω)

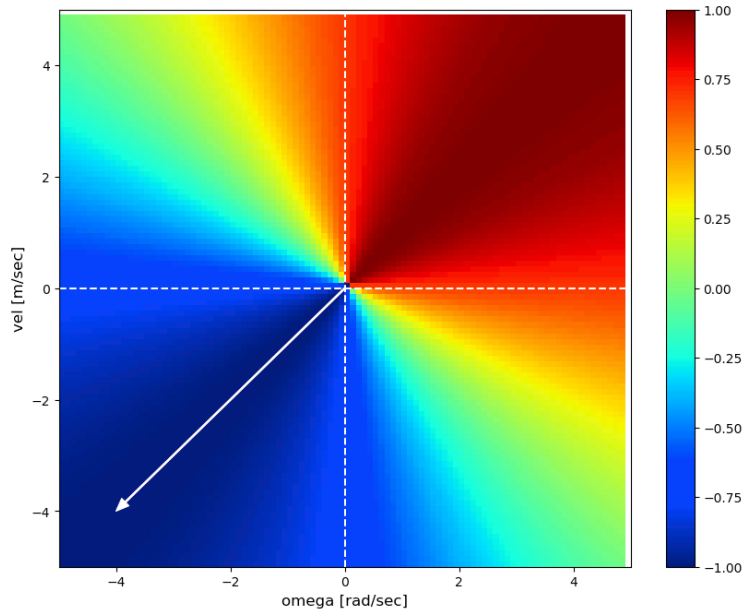


図 5.7: $(v_r, \omega_r) = (-1, -1)$ の時の \cos 類似度

このように操作者による制御入力に対して \cos 類似度を計算することで，操作者の制御入力に近い (v, ω) ほど計算されるコスト値は小さくなる．操作者の入力は式 (4.1) のようにカメラシステムを中心に計算を行っているため，カメラシステムの中心に対して計算された DWA といえる．

(b) velocity

式 (5.7) の $velocity(v, \omega)$ の項では，どれだけ移動ロボットの並進速度が大きいかどうかを評価する．よって， v の値が大きいほど，小さいコスト値になるように計算する． $0 \leq velocity(v, \omega) \leq 1$ となるように (5.12) を用いた．(5.12) において， v_{max} は移動ロボットが取りうる最大の並進速度である．

$$velocity(v, \omega) = \frac{v_{max} - v}{v_{max}} \quad (5.12)$$

(c) obstacle

式 (5.7) の $obstacle(v, \omega)$ の項では， (v, ω) を入力として生成された軌跡の最終点の座標が障害物へどれくらい接近しているかをコストで表している．そのために本研究では，costmap を利用した．以下に costmap 生成について記述する．

- costmap

costmap とは，LRF によって移動ロボット周辺に存在する障害物の位置を取得し，その位置に対応したグリッドにコスト値を割り当てた占有格子地図 (occupancy grid map) である．障害物に近いほどコスト値が高くなり，最大で 100 最小で 0 を取る．コスト値が 100 の時，障害物と接していることを意味する．

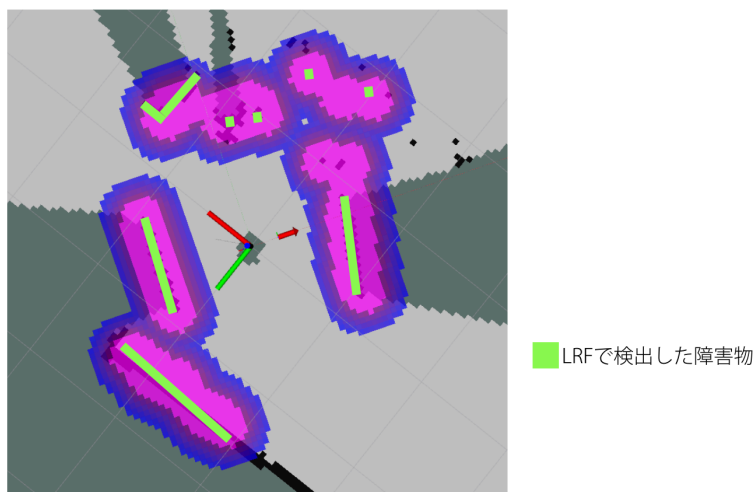


図 5.8: costmap

LRF で検出した障害物の位置からの距離に対して，図 5.9 のようにコスト値を付加する．

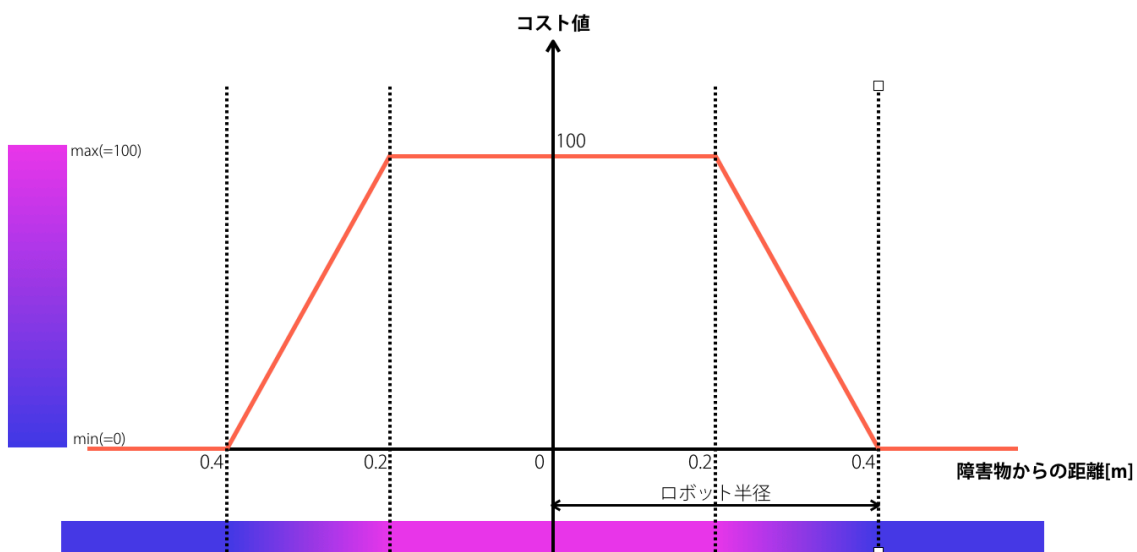


図 5.9: コスト値の付加方法

costmap においてピンク色に近いほどそのグリッドが障害物に近いことを表しており，ピンク色のグリッドはそのグリッドに移動ロボットが存在する時障害物に移動ロボットが衝突するとみなすことができる．

移動ロボットの軌跡の評価に costmap を使用する場合，軌跡の最終点の座標に対応する costmap のグリッドの値をコスト値とする．本研究で使用している移動ロボットのは，車軸中心の点を中心として，円形に近似できるので，costmap の計算に車軸中心の点の軌跡を使用した．

第6章 評価実験

6.1 実験概要

被験者に移動ロボットを遠隔操作してもらい、複数の手法を比較することで提案手法の評価を行った。表 6.1 に示すように被験者に提案手法を含めた 3 手法で移動ロボットを遠隔操作して、同じコースを走行してもらった。その時の完走時間と衝突回数を計測した。

比較した 3 つの手法について述べる。1 つ目の A 手法は、自由視野に基づいた制御手法で手動制御で操作してもらった。2 つ目の B 手法は、自由視野に基づいた制御手法に自動障害物回避を加えた手法で、自動障害物回避にポテンシャル法を用いている。提案手法である 3 つ目の C 手法は、2 つ目の手法と同じく自由視野に基づいた制御+自動障害物回避の手法であり、自動障害物回避に Dynamic Window Approach を用いている。

実験を行う際、被験者が実験の進行とともにコースや移動ロボットの操作に慣れが生じると考えたため、被験者ごとに実験を行う手法の順番を変えた。それぞれの手法について、操作後にアンケートに回答してもらい、「とても感じた」「感じた」「やや感じた」「どちらでもない」「あまり感じなかった」「感じなかった」「全く感じなかった」の 7 段階のリッカート尺度で評価を行った。表 6.2 に A, B, C 手法に共通のアンケート、表 6.3 に B, C 手法で行ったアンケートを示す。

	手法	障害物回避
A	自由視野	無し
B	自由視野+自動障害物回避	ポテンシャル法
C	自由視野+自動障害物回避	DWA

表 6.1: 手法一覧

番号	内容
1	直感的（感覚的）に操作しやすいと感じた
2	たくさんの練習が必要だと感じた
3	多くの人が簡単に操作を行うことができると感じた
4	ロボットの操作方法としてふさわしいと感じた
5	ロボットの向きや動き方が気になると感じた

表 6.2: A, B, C, 3 手法共通のアンケート項目

番号	内容
1	自動障害物回避はあったほうが良いと感じた
2	動きが滑らかだったと感じた
3	狭い空間でも快適に操作できたと感じた

表 6.3: B, C 手法共通のアンケート項目

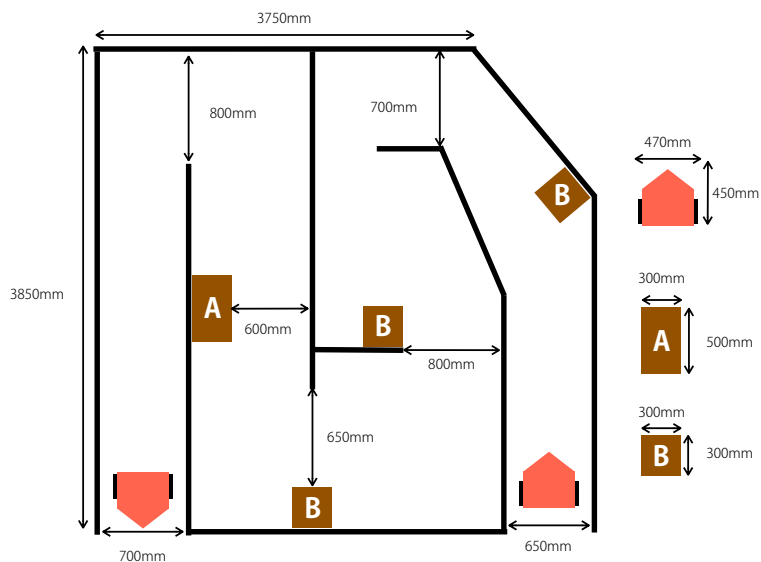


図 6.1: 実験に使用したコースの見取り図



図 6.2: 実験に使用したコースの写真 1

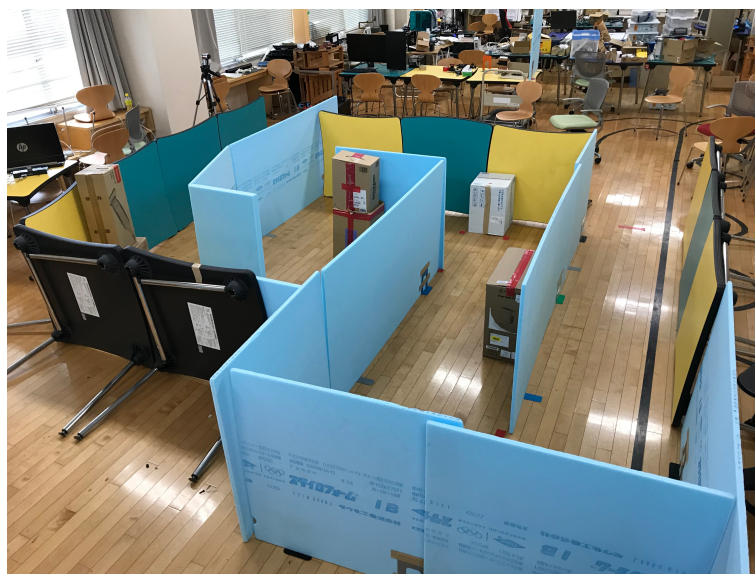


図 6.3: 実験に使用したコースの写真 2

6.2 結果

20代の男女10人（男性8人，女性2人）の被験者に，提案手法を含めた3手法で移動ロボットを遠隔操作してもらい，結果を比較することで評価を行った．コースの完走時間とその間の衝突回数を計測した．また，それぞれの手法で操作したあとにアンケートに回答してもらった．

6.2.1 完走時間と衝突回数

表 6.4 に全被験者の完走時間と衝突回数，表 6.5 に完走時間と衝突回数の合計，平均，分散，標準偏差を示す．

被験者番号	完走時間[sec]			衝突回数		
	A	B	C	A	B	C
1	78	93.59	75.08	5	0	0
2	158.07	92.9	71.89	3	0	1
3	237.98	157.44	105.8	3	0	0
4	156.88	126.27	80.89	6	0	0
5	172.75	109.12	99.5	4	1	0
6	109.91	88.37	87.45	3	0	0
7	119.05	97.67	107.41	9	0	1
8	123.96	113.71	112.32	7	2	0
9	90.09	124.06	84.77	4	2	0
10	138.47	107.37	120.37	7	1	0

表 6.4: 全被験者の完走時間と衝突回数

	完走時間[sec]			衝突回数		
	A	B	C	A	B	C
合計	1385.2	1110.5	945.5	51	6	2
平均	138.5	111.1	94.5	5.1	0.6	0.2
分散	1920.1	390.5	252.4	3.9	0.6	0.2
標準偏差	43.8	19.8	15.9	2.0	0.8	0.4

表 6.5: 完走時間と衝突回数の合計, 平均, 分散, 標準偏差

遠隔操作手法 A, B, C によるコースの完走時間の平均の差を分析するために有意差水準 5% で分散分析を行った。その結果, 異なる手法で有意な差が認められた ($F=3.354, p<.05$)。

評価項目	統計量	A	B	C
完走時間	平均値	138.516	111.05	94.548
	標準偏差	43.819	19.761	15.887
	有意確率	$F=3.354, p<.05(*)$		

表 6.6: 「遠隔操作手法によるコースの完走時間の平均」に関する分散分析の結果

次に, 3つの手法によるコースの完走時間の平均の差を調べるために, 有意差水準 5% で最小有意差法 (LSD 法) による多重比較を行った。その結果, 「手法 A, 手法 C」に有意な差が確認された。図 6.4 に「遠隔操作手法によるコースの完走時間の平均」に関する評価の結果と表 6.7 に LSD 法による多重比較の結果として各手法間の p 値を示す。

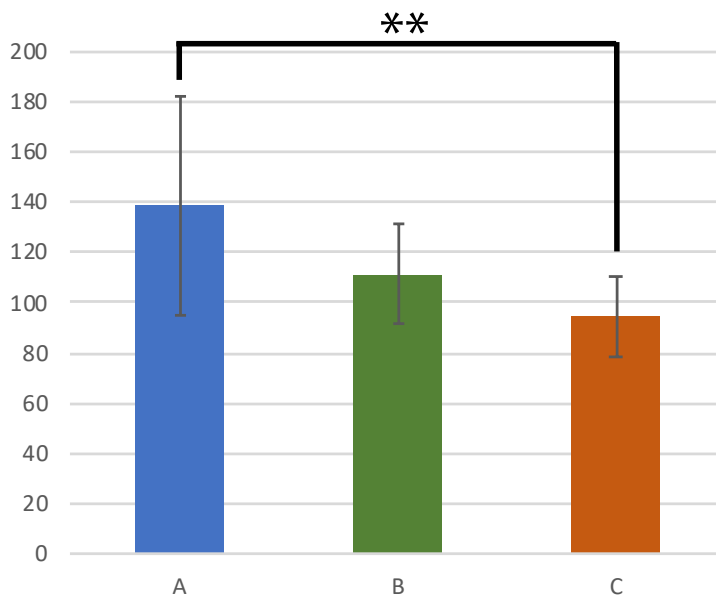


図 6.4: 「遠隔操作手法によるコースの完走時間の平均」に関する評価の結果

有意確率 (多重比較)	A	B	C
A	×	0.056	p<.01(**)
B	0.056	×	0.241
C	p<.01(**)	0.241	×

表 6.7: 「遠隔操作手法によるコースの完走時間の平均」に関する LSD 法による多重比較の結果

遠隔操作手法 A, B, C による衝突回数の平均の差を分析するために有意差水準 5% で分散分析を行った。その結果, 異なる手法で有意な差が認められた (F=3.354, p<.05)

評価項目	統計量	A	B	C
衝突回数	平均値	5.1	0.6	0.2
	標準偏差	1.972	0.8	0.4
	有意確率	F=3.354, p<.05(***)		

表 6.8: 「遠隔操作手法による衝突回数の平均」に関する分散分析の結果

次に, 3つの手法の衝突回数の平均の差を分析するために有意差水準 5% で LSD 法による多重比較を行った。その結果, 「手法 A, 手法 B」「手法 A, 手法 C」に有意な差が確認された。図 6.5 に「遠隔操作手法による衝突回数の平均」に関する評価の結果と表 6.9 に LSD 法による多重比較の結果として各手法間の p 値を示す。

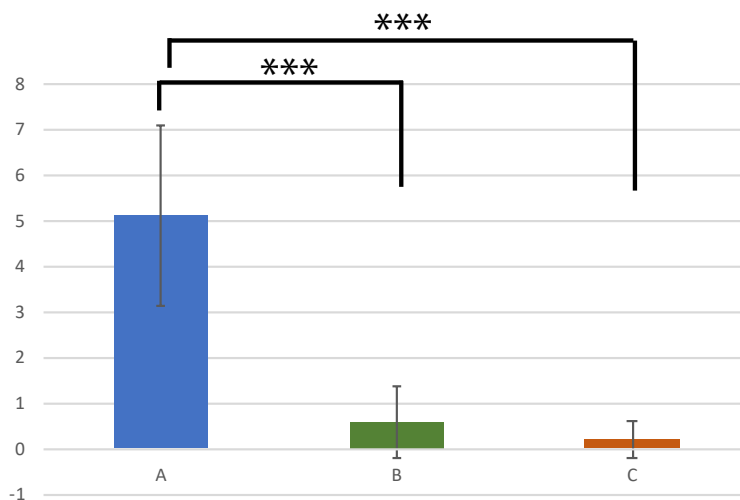


図 6.5: 「遠隔操作手法による衝突回数の平均」に関する評価の結果

有意確率 (多重比較)	A	B	C
A	×	p<.001(***)	p<.001(***)
B	p<.001(***)	×	0.503
C	p<.001(***)	0.503	×

表 6.9: 「遠隔操作手法による衝突回数の平均」に関する LSD 法による多重比較の結果

6.2.2 アンケート

表 6.10 に A, B, C 手法共通, 表 6.11 に B, C 手法共通のアンケート結果の合計, 平均, 分散, 標準偏差を示す.

評価方法	手法	直感的 (感覚的)に操作しやすいと感じた	たくさんの練習が必要だと感じた	多くの人が簡単に操作を行うことができると感じた	ロボットの操作としてふさわしいと感じた	ロボットの向きや動き方が気になると感じた
合計	A	30	56	27	35	55
	B	47	47	48	49	51
	C	60	37	54	56	37
平均	A	3	5.6	2.7	3.5	5.5
	B	4.7	4.7	4.8	4.9	5.1
	C	6	3.7	5.4	5.6	3.7
分散	A	1.6	1.8	1.8	2.9	1.5
	B	1.2	1.0	1.4	0.9	1.9
	C	0.4	1.0	0.6	0.4	1.0
標準偏差	A	1.3	1.4	1.3	1.7	1.2
	B	1.1	1.0	1.2	0.9	1.4
	C	0.6	1.0	0.8	0.7	1.0

表 6.10: A, B, C 手法共通のアンケート結果の合計, 平均, 分散, 標準偏差

評価方法	手法	自動障害物回避はあったほうが良いと感じた	動きが滑らかだったと感じた	狭い空間でも快適に操作できたと感じた
合計	B	61	42	49
	C	64	59	54
平均	B	6.1	4.2	4.9
	C	6.4	5.9	5.4
分散	B	0.9	3.2	1.1
	C	0.4	0.9	0.8
標準偏差	B	0.9	1.8	1.0
	C	0.7	0.9	0.9

表 6.11: B, C 手法共通のアンケート結果の合計, 平均, 分散, 標準偏差

以下, それぞれの項目について分析を行った.

- 直感的（感覚的）に操作しやすいと感じた
遠隔操作手法 A, B, C による「直感的（感覚的）に操作しやすいと感じた」の項目について分析するために有意差水準 5% で分散分析を行った。その結果、異なる手法で有意な差が認められた ($F=3.354, p<.001$)

評価項目	統計量	A	B	C
完走時間	平均値	3	4.7	6
	標準偏差	1.265	1.1	0.632
	有意確率	$F=3.354, p<.001(***)$		

表 6.12: 「直感的（感覚的）に操作しやすいと感じた」の項目に関する分散分析の結果

次に、3つの手法の「直感的（感覚的）に操作しやすいと感じた」の項目についての差を分析するために有意差水準 5% で LSD 法による多重比較を行った。その結果、「手法 A, 手法 B」「手法 B, 手法 C」「手法 C, 手法 A」に有意な差が確認された。図 6.6 と表 6.13 に「直感的（感覚的）に操作しやすいと感じた」の項目に関する評価の結果と LSD 法による多重比較の結果として各手法間の p 値を示す。

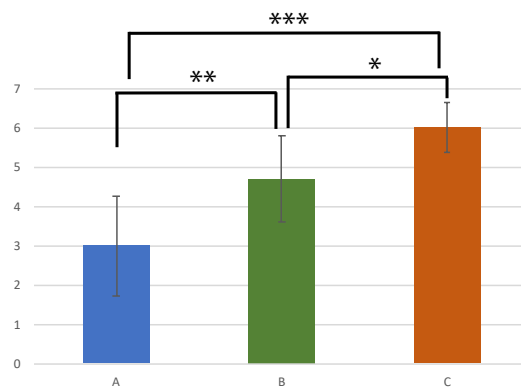


図 6.6: 直感的（感覚的）に操作しやすいと感じた

有意確率（多重比較）	A	B	C
A	×	$p<.01(**)$	$p<.001(***)$
B	$p<.001(***)$	×	$p<.05(*)$
C	$p<.001(***)$	$p<.05(*)$	×

表 6.13: 「直感的（感覚的）に操作しやすいと感じた」に関する LSD 法による多重比較の結果

- たくさんの練習が必要だと感じた
遠隔操作手法 A, B, C による「たくさんの練習が必要だと感じた」の項目について分析するために有意差水準 5% で分散分析を行った。その結果、異なる手法で有意な差が認められた ($F=3.354, p<.01$)

評価項目	統計量	A	B	C
完走時間	平均値	5.6	4.7	3.7
	標準偏差	1.356	1.005	1.005
	有意確率	$F=3.354, p<.01(**)$		

表 6.14: 「たくさんの練習が必要だと感じた」の項目に関する分散分析の結果

次に、3つの手法の「たくさんの練習が必要だと感じた」の項目についての差を分析するために有意差水準 5% で LSD 法による多重比較を行った。その結果、「手法 C, 手法 A」に有意な差が確認された。図 6.7 と表 6.15 に「たくさんの練習が必要だと感じた」の項目に関する評価の結果と LSD 法による多重比較の結果として各手法間の p 値を示す。

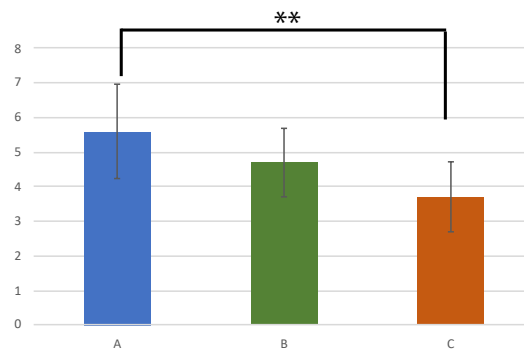


図 6.7: たくさんの練習が必要だと感じた

有意確率 (多重比較)	A	B	C
A	×	0.104	$p<.01(**)$
B	0.104	×	0.072
C	$p<.01(**)$	0.072	×

表 6.15: 「たくさんの練習が必要だと感じた」に関する LSD 法による多重比較の結果

- 多くの人が簡単に操作を行うことができると感じた
遠隔操作手法 A, B, C による「多くの人が簡単に操作を行うことができると感じた」の項目について分析するために有意差水準 5% で分散分析を行った。その結果、異なる手法で有意な差が認められた ($F=3.354, p<.001$)

評価項目	統計量	A	B	C
完走時間	平均値	2.7	4.8	5.4
	標準偏差	1.345	1.166	0.8
	有意確率	F=3.354, p<.001(***)		

表 6.16: 「多くの人が簡単に操作を行うことができると感じた」の項目に関する分散分析の結果

次に、3つの手法の「多くの人が簡単に操作を行うことができると感じた」の項目についての差を分析するために有意差水準 5% で LSD 法による多重比較を行った。その結果、「手法 A, 手法 B」「手法 C, 手法 A」に有意な差が確認された。図 6.8 と表 6.17 に「多くの人が簡単に操作を行うことができると感じた」の項目に関する評価の結果と LSD 法による多重比較の結果として各手法間の p 値を示す。

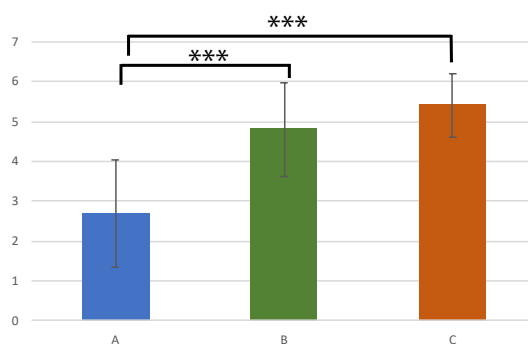


図 6.8: 多くの人が簡単に操作を行うことができると感じた

有意確率 (多重比較)	A	B	C
A	×	p<.001(***)	p<.001(***)
B	p<.001(***)	×	0.269
C	p<.001(***)	0.269	×

表 6.17: 「多くの人が簡単に操作を行うことができると感じた」に関する LSD 法による多重比較の結果

- ロボットの操作としてふさわしいと感じた
遠隔操作手法 A, B, C による「ロボットの操作としてふさわしいと感じた」の項目について分析するために有意差水準 5% で分散分析を行った。その結果, 異なる手法で有意な差が認められた ($F=3.354, p<.01$)

評価項目	統計量	A	B	C
完走時間	平均値	3.5	4.9	5.6
	標準偏差	1.688	0.943	0.663
	有意確率	F=3.354, p<.01(**)		

表 6.18: 「ロボットの操作としてふさわしいと感じた」の項目に関する分散分析の結果

次に, 3つの手法の「ロボットの操作としてふさわしいと感じた」の項目についての差を分析するために有意差水準 5% で LSD 法による多重比較を行った。その結果, 「手法 A, 手法 B」「手法 C, 手法 A」に有意な差が確認された。図 6.9 と表 6.19 に「ロボットの操作としてふさわしいと感じた」の項目に関する評価の結果と LSD 法による多重比較の結果として各手法間の p 値を示す。

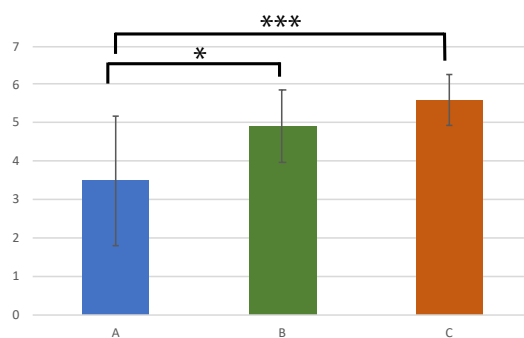


図 6.9: ロボットの操作としてふさわしいと感じた

有意確率 (多重比較)	A	B	C
A	×	p<.05(*)	p<.001(***)
B	p<.05(*)	×	0.219
C	p<.001(***)	0.219	×

表 6.19: 「ロボットの操作としてふさわしいと感じた」に関する LSD 法による多重比較の結果

- ロボットの向きや動き方が気になると感じた
遠隔操作手法 A, B, C による「ロボットの向きや動き方が気になると感じた」の項目について分析するために有意差水準 5% で分散分析を行った。その結果、異なる手法で有意な差が認められた ($F=3.354, p<.01$)

評価項目	統計量	A	B	C
完走時間	平均値	5.5	5.1	3.7
	標準偏差	1.204	1.374	1.005
	有意確率	F=3.354, p<.01(**)		

表 6.20: 「ロボットの向きや動き方が気になると感じた」の項目に関する分散分析の結果

次に、3つの手法の「ロボットの向きや動き方が気になると感じた」の項目についての差を分析するために有意差水準 5% で LSD 法による多重比較を行った。その結果、「手法 B, 手法 C」「手法 C, 手法 A」に有意な差が確認された。図 6.10 と表 6.21 に「ロボットの向きや動き方が気になると感じた」の項目に関する評価の結果と LSD 法による多重比較の結果として各手法間の p 値を示す。

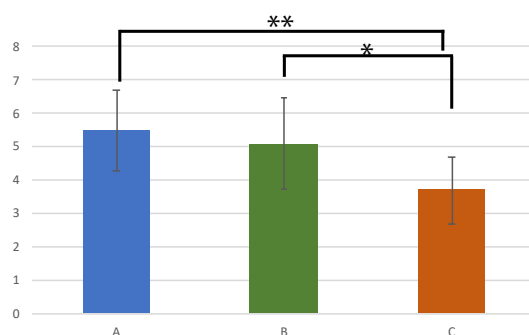


図 6.10: ロボットの向きや動き方が気になると感じた

有意確率 (多重比較)	A	B	C
A	×	p<.0487	p<.01(**)
B	p<.0487	×	p<.05(*)
C	p<.01(**)	p<.05(*)	×

表 6.21: 「ロボットの向きや動き方が気になると感じた」に関する LSD 法による多重比較の結果

ポテンシャル法を用いた半自律遠隔操作手法である B 手法と DWA を用いた半自律遠隔操作手法である C 手法において、アンケートの「自動障害物回避はあったほうが良いと感じた」、「動きが滑らかだったと感じた」、「狭い空間でも快適に操作できたと感じた」という項目について有意差水準 5% のウィルコクソンの順位和検定を行った。その結果、「動きが滑らかだったと感じた」の項目について代表値に有意差があると示唆された。図 6.11 図 6.12 図 6.13 と表 6.22 にそれらの項目に関する評価の結果および 7 段階評価の平均値及び標準偏差を示す。

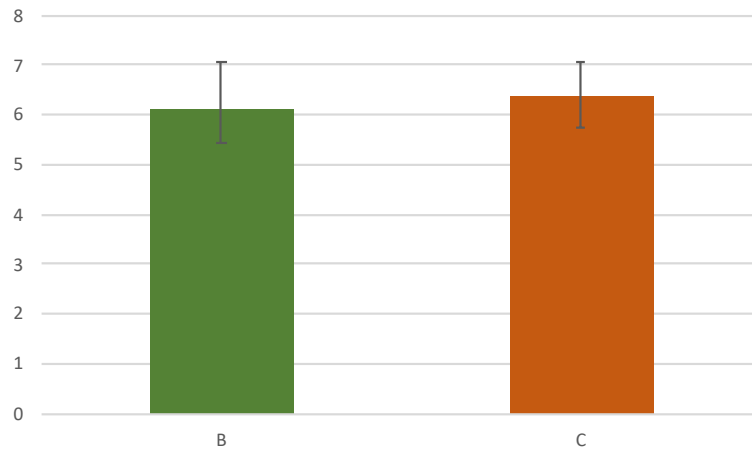


図 6.11: 自動障害物回避はあったほうが良いと感じた

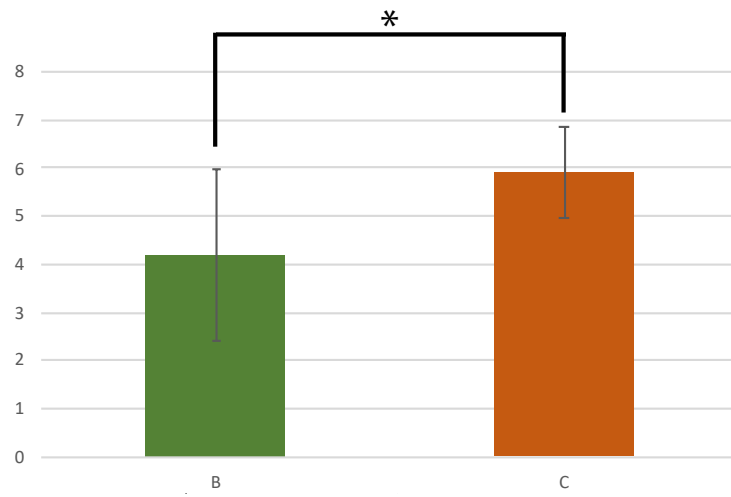


図 6.12: 動きが滑らかだったと感じた

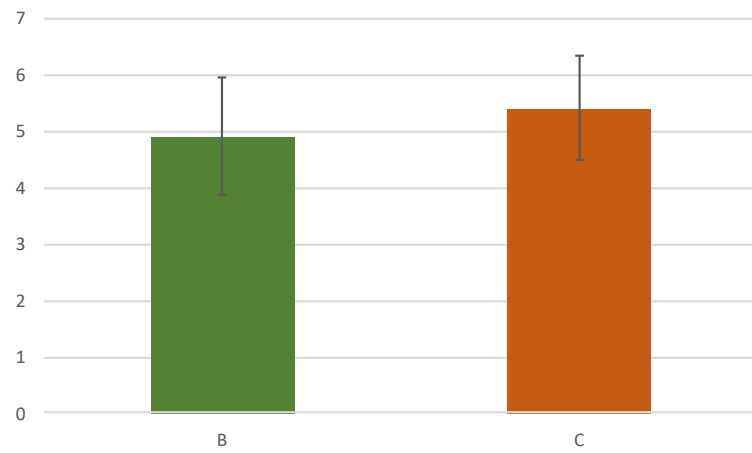


図 6.13: 狭い空間でも快適に操作できた

評価項目	p 値
自動障害物回避があったほうが良いと感じた	p=0.629
動きが滑らかだったと感じた	p=0.034, p<.05(*)
狭い空間でも快適に操作できたと感じた	p=0.365

表 6.22: B 手法と C 手法共通のアンケート項目に関する 7 段階評価のウィルコクソンの順位和検定の結果

第7章 考察

完走時間について、A, B, C手法で比較すると、図6.4より、平均値はA, B, Cの順で小さくなっている。平均値の差を調べるために、多重比較を行った結果、「手法A, 手法C」に有意な差を確認できたが、「手法A, 手法B」「手法B, 手法C」には有意な差を確認できなかった。「手法A, 手法C」については、DWAを利用した半自律制御が操作者の負担を減らした結果、コースの途中で停止する回数が減ったことが理由だと考えられる。一方、その他の手法間で有意差を確認できなかったのは、操作者の遠隔操作への慣れや性格により、完走時間に大きな差が生じてしまうことが原因と考えられる。

次に、衝突回数について3手法を比較すると、図6.5に示すように、平均値の差について多重比較を行った結果、「手法A, 手法B」「手法A, 手法C」に有意な差を確認できた。これは、手法Bで用いたポテンシャル法、手法Cで用いたDWAともに障害物回避を行っていたことを示している。「手法B, 手法C」に有意な差を確認できなかったのは、障害物回避の性能について両手法には違いがないことを意味している。完走時間と衝突回数の結果から、自動障害物回避を行わない手法Aよりも、自動障害物回避を行う手法Bは衝突回数で、同じく自動障害物回避を行う手法Cは完走時間と衝突回数の両方で、優れているといえる。

手法Bと手法Cについて比較すると、図6.12に示すようにアンケートの「動きが滑らかだったと感じた」の項目で有意差が確認できた。これは、ポテンシャル法とDWAの障害物回避を行う際の制御の違いであるといえる。ポテンシャル法は、操作者の入力から求めるベクトルと障害物からの斥力ベクトルを合成することで、障害物回避を実現している。操作者の入力があった時点での移動ロボット周囲の環境に対して計算を行うので、移動ロボットへの制御入力が大きく変化する特徴を持つ。

一方、DWAは現在の移動ロボットの速度情報を考慮しつつ、1タイムステップ後の軌跡を予測しながら障害物回避を行うので、ロボットへの制御入力連続的に変化していく特徴を持つため、滑らかな移動ロボットの制御ができると考えられる。図7.1と図7.2はそれぞれ直線の通路をB手法とC手法で遠隔操作したときの移動ロボットの軌跡である。前者は蛇行するように進んでいるのに対し、後者は直進している様子がわかる。このように、手法Bより手法Cの方が滑らかな動きをするように制御されているにも関わらず、完走時間に有意差が確認できなかった要因として、手法Cにおいて、障害物に直進していった場合、DWAの軌跡評価の結果として操作者の意図した移動ロボットへの制御入力と大きく違う制御入力が行われることがある。この問題に対しては、視野方向に対応して、Dynamic Windowを更に改良することで視野方向と大きく違う方向へ進む制御入力を除外することができると考える。また、アンケートの回答で、操作者への提示画像の中心に目印となるマークを描画することで、操作者がロボットの進行方向を理解しやすくなるなど、水平360度視野画像にも改良する必要がある。

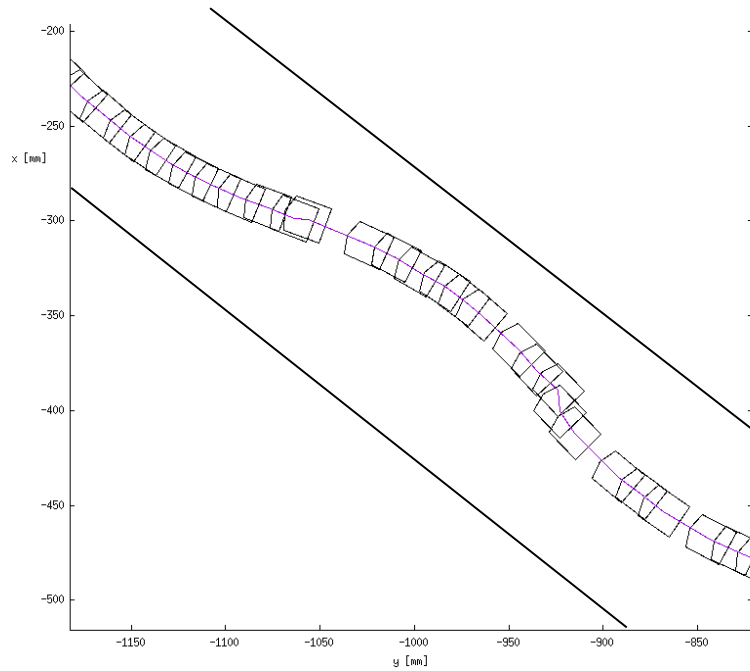


図 7.1: 自由視野+ポテンシャル法を用いた半自律制御手法の軌跡

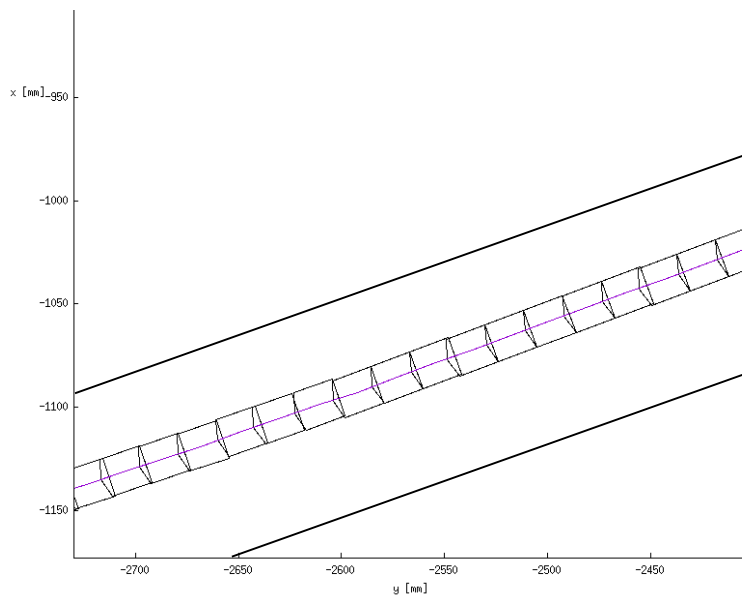


図 7.2: 提案手法の軌跡

第8章 結論

本論文では、移動ロボットのための自由視野と Dynamic Window Approach (DWA) を用いた半自律遠隔操作手法を提案した。水平 360 度視野画像とその画像内で操作者が選択した視野方向に移動ロボットが前進するシステムを実装した。このシステムにより操作者は移動ロボットの姿勢や大きさを意識することなく遠隔操作することが可能になった。しかし、その影響で操作者が気づかないうちに障害物に衝突してしまう恐れがある。そこで、操作者の入力に DWA を用いた障害物回避を組み合わせた半自律制御を行うことでその問題を解決した。本研究では、操作者の入力を考慮した DWA を実装した。

実装したシステムを評価するために、障害物回避を行わない自由視野に基づく手動制御手法と先行研究で提案されたポテンシャル法を用いた障害物回避を行う自由視野に基づく半自律制御手法と本論文で提案した手法の 3 つの手法で被験者に移動ロボットを操作してもらう実験を行った。その際、完走時間と衝突回数を計測し、実験終了後にアンケートに回答してもらった。提案手法は実験で比較した手動制御手法よりも完走時間と衝突回数で優れているという結果が得られた。また、ポテンシャル法を用いた半自律制御手法と比べて滑らかな制御を行えているという結果を得ることができ、提案手法の有用性を確認することができた。しかし、障害物に直進する操作をした場合、DWA によるコストの計算の結果、操作者の意図した向きと大きく違う方向に移動ロボットが進む問題が生じた。これにより、操作者の意図した動きにならず、完走時間にも影響が及んだ。

今後の課題として、この問題に対しては、移動ロボットが常に視野方向に進むように視野方向に対応した Dynamic Window を変更する処理を加えることで解決することができると考えられる。また、操作者に提示している画像の中心に目印を描画し、操作者がロボットの進行方向を理解しやすくするなど、水平 360 度視野画像の改良も必要であると考えられる。

謝辞

研究生活を過ごす中で、未熟な私に多大な時間を割いて研究に限らず様々なことをご指導して下さった三河正彦准教授に心より感謝申し上げます。また、合同ゼミでご指導頂いた藤澤誠助教に深く感謝いたします。そして、合同ゼミやその他の場面で研究に限らず話をして下さった藤澤研究室の皆様、研究生活で最も多く一緒に過ごした三河研究室の皆様、支えて下さった家族、友人の方々に感謝いたします。

参考文献

- [1] Keiji Nagatani, Seiga Kiribayashi, Yoshito Okada, Satoshi Tadokoro, Takeshi Nishimura, Tomoaki Yoshida, Eiji Koyanagi, and Yasushi Hada. Redesign of rescue mobile robot quince. In *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*, pages 13–18. IEEE, 2011.
- [2] Maki Sugimoto, Georges Kagotani, Hideaki Nii, Naoji Shiroma, Fumitoshi Matsuno, and Masahiko Inami. Time follower’s vision: a teleoperation interface with past images. *IEEE Computer Graphics and Applications*, 25(1):54–63, 2005.
- [3] Fumio Okura, Yuko Ueda, Tomokazu Sato, and Naokazu Yokoya. Free-viewpoint mobile robot teleoperation interface using view-dependent geometry and texture. *ITE Transactions on Media Technology and Applications*, 2(1):82–93, 2014.
- [4] Yuta Odate, Masahiko Mikawa, Wasuke Hiiragi, and Makoto Fujisawa. Wide view image stitching method with little distortion using multi cameras and application to mobile robot teleoperation. *SICE Annual Conference 2016*, 2016.
- [5] 国土交通省. 自動運転における損害賠償責任に関する研究会, 3 2018.
- [6] Martin Voshell, David D Woods, and Flip Phillips. Overcoming the keyhole in human-robot coordination: simulation and evaluation. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 49, pages 442–446. Sage Publications Sage CA: Los Angeles, CA, 2005.
- [7] Ami Eliav, Talia Lavie, Yisrael Parmet, Helman Stern, and Yael Edan. Advanced methods for displays and remote control of robots. *Applied ergonomics*, 42(6):820–829, 2011.
- [8] Lisa C Thomas and Christopher D Wickens. Effects of display frames of reference on spatial judgments and change detection. Technical report, ILLINOIS UNIV AT URBANA-CHAMPAIGN SAVOYAVIATION RESEARCH LAB, 2000.
- [9] Nicolas Small, Kevin Lee, and Graham Mann. An assigned responsibility system for robotic teleoperation control. *International journal of intelligent robotics and applications*, 2(1):81–97, 2018.
- [10] Leila Takayama, Eitan Marder-Eppstein, Helen Harris, and Jenay M Beer. Assisted driving of a mobile remote presence system: System design and controlled user evaluation. In *2011 IEEE international conference on robotics and automation*, pages 1883–1889. IEEE, 2011.
- [11] Jen-Hui Chuang and Narendra Ahuja. An analytically tractable potential field model of free space and its application in obstacle avoidance. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 28(5):729–736, 1998.

- [12] Dieter Fox, Wolfram Burgard, and Sebastian Thrun. The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1):23–33, 1997.
- [13] Ros.org. <http://wiki.ros.org>.
- [14] Brian Gerkey. gmapping documentation. <http://wiki.ros.org/gmapping>.
- [15] Steve Agoston, James Hollyer, and Matt Reynolds. Gamepadapi documentation. <https://w3c.github.io/gamepad/>.