博士論文

画像に基づくファジィ制御器を用いた 移動ロボットの制御

平成 25 年 3 月

加藤 達也

岡山大学大学院 自然科学研究科

論文概要

本論文は移動ロボットのための画像に基づく視覚サーボをファジィ制御器によっ て行う手法を提案するものである、視覚サーボはカメラ画像を基にロボットを制 御する手法であり「位置に基づく視覚サーボ」と「画像に基づく視覚サーボ」に |大別される . 位置に基づく視覚サーボはカメラ画像の特徴量と事前知識として持っ ている環境や物体の三次元モデルとを照合することでロボットの位置を推定し,そ の位置を制御する手法である、この手法ではロボットの位置を直接制御するため、 これまで提案されてきた制御理論を容易に適用することができるという利点があ るが,位置推定を行うための計算量が多く,環境や物体の三次元モデルが要求さ れる.また,これは位置を基に制御するという従来の移動ロボットの制御手法と 同じような制御手法であると思われる.一方,画像に基づく視覚サーボはカメラ 画像の特徴量からロボットの制御入力を直接決定する手法である.位置に基づく 視覚サーボと比べて位置推定のプロセスが必要ないため,計算量が少ないという 利点がある.また,この手法は,我々人間が自分自身の正確な位置を気にするこ となく行動していることから、人間的な処理を行っている制御手法であると思わ れる、しかしながら、画像特徴量とロボットの動作を結びつける特殊な制御器を 設計することが必要であり,実機による実現例は少なく,十分な検証が行われて いるとは言えない、そこで、本研究では画像に基づく視覚サーボを取り扱うこと とし、移動ロボットを制御対象として実機による検証までを行うことを研究目的 とする.

画像に基づく視覚サーボに関する先行研究には相互行列 (Interaction matrix) を用いたモデルベースの手法が多く提案されている.相互行列とは画像座標上の点の運動とロボットの制御入力との関係をモデル化したものであるが,その点の奥行きに依存しているため,その点の三次元位置を推定する必要がある.しかしながら,これでは画像に基づく視覚サーボの利点である計算量の少なさを損なってしまう恐れがある.そこで,画像に基づく視覚サーボがより人間的な制御手法であると考えると,モデルベース制御ではなく経験に基づくファジィ制御や人間の脳を模したニューラルネットワーク制御などのモデルフリーの制御手法が適していると思われる.このことより本論文では,ファジィ制御器を用いた手法を提案し,移動ロボットのシミュレーションと実機実験を通して,提案手法の有用性を実証する.

本論文で取り組む問題は移動ロボット制御の1つである軌道追従制御であり,地面に描かれた軌道をカメラを搭載した移動ロボットが追従することを目標とした.ファジィ制御器は経験的に獲得した制御規則を扱うことができるため,制御対象を乗用車のような操舵型の4輪移動ロボットとし,人間が行う車の運転を参考に制御器を設計することを行った.設計する制御器はカメラ画像に映った目標軌道の特徴量からロボットの制御入力を決定しなければならない.まずは,直線を目標軌道とし,カメラ画像に映った目標軌道の傾きと切片を求めて,これらの値を

基にロボットの制御入力を決定するファジィ制御器を設計した.カメラ画像から これらの値を抽出する方法として直線ハフ変換を用いることとした.シミュレー ションと実機実験の両方で評価を行い、提案手法によって直線軌道追従を達成す ることができた、次に、より複雑な任意軌道を追従する問題に対して、手作業に よる設計ではなく GA を用いたファジィ制御器の設計手法を提案した.この場合 は、画像に映った目標軌道全体を包括した特徴量を抽出するために、最小二乗法 を用いて求められた近似直線の傾きと切片を基に制御入力を決定する制御器を設 計することとした.この手法では,ファジィ制御器のメンバシップ関数の形状を決 定するパラメータと制御規則の後件部の定数値をコード化して, GA を用いてそれ らの最適値を探索することで制御器を設計する、また、GA による設計時のシミュ レーションにノイズを付加することで制御器が評価用の目標軌道に特化すること を防止し,ロバスト性を向上させる工夫を行った.提案手法はシミュレーション と実機実験によって評価し、付加するノイズの大きさによってロボットの挙動が 変化し、適当な大きさのノイズを付加することで実機でも目標軌道を追従するこ とが可能であることを確認した.さらに,画像に基づくファジィ制御器の応用例 として、画像に基づく障害物回避の手法を提案した、この手法では、取得したカ メラ画像からエッジ検出によって障害物を検出し、このエッジを避けるように生 成した目標軌道を画像に基づくファジィ制御器によって追従することで障害物回 避を行う.実機実験によって提案した障害物回避手法の有用性を示した.

本論文は6章で構成されており、各章の内容は以下のとおりである、第1章では 研究背景について述べ、視覚サーボの概要について位置に基づく視覚サーボと画 像に基づく視覚サーボの2つの手法の違いについてその詳細を述べる.また,本 研究の目的と本論文の構成について述べる.第2章では,先行研究であるモデル ベースの画像に基づく視覚サーボについて述べ、シミュレーションを行い、その 特徴について述べる、第3章では直線軌道追従を問題設定として、提案手法であ る画像に基づくファジィ制御器の設計を行う.また,設計した制御器についてシ ミュレーションと実機実験の両方を行い、その性能を評価する、第4章では任意 軌道追従を問題設定として、より複雑なファジィ制御器をGAによって設計する手 法を提案する.また,評価関数の重み係数とロボットの挙動の関係や設計時のシ ミュレーションに付加するノイズとロボットの挙動の関係について、シミュレー ションと実機実験による評価を行う.第5章では画像に基づくファジィ制御器の応 用例として、画像に基づく障害物回避手法を提案する、カメラ画像から障害物を 検出し画像上にその障害物を回避する軌道を生成し,前述のファジィ制御器を用 いてその軌道を追従することで障害物回避を行う.第6章では各章で述べた内容 を総括し,本論文をまとめる.

目次

第1章	序論	1
1.1	研究背景....................................	1
1.2	視覚サーボ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	3
	1.2.1 位置に基づく視覚サーボ	5
	1.2.2 画像に基づく視覚サーボ	
1.3	ファジィ制御器を用いたロボット制御	8
1.4	研究目的....................................	8
1.5	本論文の構成	
第2章	相互行列を用いたモデルベースの画像に基づく視覚サーボ	11
2.1	問題設定	11
2.2	相互行列の導出	13
2.3	リヤプノフ直接法を用いた制御器の設計	15
2.4	シミュレーション	16
2.5	まとめ	20
第3章	画像に基づくファジィ制御器を用いた直線軌道追従	21
3.1	問題設定	21
3.2	ハフ変換を用いた直線軌道の抽出	24
3.3	人間が行う自動車の運転に基づいたファジィ制御器	27
	3.3.1 ファジィ制御器の設計	27
	3.3.2 シミュレーション	32
3.4	状態変数の変化量を考慮したファジィ制御器の設計	37
	3.4.1 ファジィ制御器の設計	37
	3.4.2 シミュレーションと実機実験	39
3.5	まとめ	45
第4章	任意軌道追従のためのファジィ制御器の GA による設計	46
4.1	問題設定	46
4.2	任意軌道追従のためのファジィ制御器	49
4.3	GA を用いたファジィ制御器の設計	51
4.4	実験 1	51

	4.4.1 GA によるファジィ制御器の設計	52
	4.4.2 シミュレーション上のロボットの動作	53
	4.4.3 ノイズの種類とロボットの動作	56
4.5	実験 2	59
	4.5.1 重み係数とロボットの挙動	60
	4.5.2 付加するノイズの大きさとロボットの挙動	63
4.6	まとめ	69
第5章	画像に基づく移動ロボットの障害物回避	7 0
5.1	問題設定	70
5.2	障害物回避軌道の生成	71
	5.2.1 ポテンシャルの計算	72
	5.2.2 走行可能領域の判別	72
	5.2.3 目標軌道の生成	73
5.3	障害物回避のためのファジィ制御器	73
	5.3.1 ファジィ制御器の設計	74
	5.3.2 ファジィ制御器の性能評価実験	78
5.4	実機実験	81
5.5	まとめ	85
第6章	結論	86
	謝辞	88
付録A	リヤプノフの安定性理論	89
	参考文献	90
	発表論文	97

図目次

1.1	Examples of visual servoing for robot control	3
1.2	Examples of extracted image features from a captured image	4
1.3	Typical block diagram of a position-based visual servoing	5
1.4	Typical block diagram of an image-based visual servoing	7
2.1	Difinition of coordinates	12
2.2	Robot model	12
2.3	Camera model	13
2.4	Coordinate of the image plane	13
2.5	Simulation results for the model based control method	18
2.5	Simulation results for the model based control method	19
3.1	World coordinate and image coordinate	22
3.2	Model of a mobile robot	22
3.3	Block diagram of the image-based control	23
3.4	Concept of the Hough transform for extracting a line	25
3.5	Line representation using parameters ρ and θ	26
3.6	Parameter space for ρ and θ	26
3.7	Membership functions for the controller based on human driving tech-	
	niques	29
3.8	Simulation results for the controller based on human driving techniques	34
3.8	Simulation results for the controller based on human driving techniques	35
3.8	Simulation results for the controller based on human driving techniques	36
3.9	Membership functions for the controller considering $\Delta \alpha$ and $\Delta \beta$	38
3.10	Simulation result for the controller considering $\Delta \alpha$ and $\Delta \beta$	40
3.11	Experimental robot	41
3.12	Image processing	42
3.13	Appearance of the robot (0.5 [s] interval)	42
3.14	Experimental result for the real robot	43
3.15	Robot advancing along the target line	44
3.16	Robot advancing toward the target line	44
4.1	Definition of coordinates	47

4.2	Block diagram of image-based tracking control for an arbitrary trajectory	48
4.3	Membership functions for arbitrary trajectory tracking	50
4.4	Chromosomeencoding membership function parameters and consequent	
	constants	51
4.5	History of the evaluation function values	53
4.6	Simulation result of using the designed fuzzy controllers with additional	
	motion noise σ_{m}	54
4.7	Simulation result of using the designed fuzzy controllers with additional	
	observation noise σ_0	55
4.8	Experimental result of using the designed fuzzy controllers with addi-	
	tional motion noise $\sigma_{\rm m}$	57
4.9	Experimental result of using the designed fuzzy controllers with addi-	
	tional observation noise σ_0	58
4.10	Simulation results when using the fuzzy controllers A, B and C	61
	Simulation results when using the fuzzy controllers A, B and C	62
	Simulation results when using the fuzzy controllers D, E and F	65
	Simulation results when using the fuzzy controllers D, E and F	66
	Experimental results when using the fuzzy controllers D, E and F, where	
	the robot motion was picked up every 1.0 [s] interval	67
4 13	State variables α and β in experiments when using the fuzzy controllers	07
1.13	D, E and F	68
	D, D and I	00
5.1	Overview of the present image-based obstacle avoidance	71
5.2	Block diagram of the present image-based obstacle avoidance	72
5.3	Illustration of generating a trajectory for the obstacle avoidance	73
5.4	Membership functions for obstacle avoidance	76
5.5	Simulation result of the trajectory of the robot	79
5.6	Translational velocity of the robot on the simulation	79
5.7	Actual experimental results for testing the fuzzy controller	80
5.8	Experimental environment (unit in cm)	82
5.9	Robot movement and image processing in the case when the initial po-	
	sition was assigned to B	83
5.10	Experimental result in the case when the initial position was assigned to A	83
	Experimental result in the case when the initial position was assigned to C	

表目次

3.1	Fuzzy rules for $\dot{\phi}$ in the controller based on human driving techniques .	30
3.2	Fuzzy rules for \dot{s} in the controller based on human driving techniques	31
3.3	Robot parameters	32
3.4	Fuzzy rules for ϕ in the controller considering $\Delta \alpha$ and $\Delta \beta$	39
3.5	Specifications of the experimental robot	41
5.1	Fuzzy rules for the steering angle ϕ	77
5.2	Fuzzy rules for the translational velocity s	77

第1章 序論

本章では,研究背景として移動ロボット,視覚サーボ,ファジィ制御に関する背景を述べ,それぞれの分野の関連研究を紹介し本論文の位置付けおよび目的について述べる.

1.1 研究背景

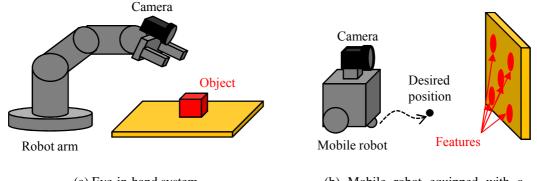
移動ロボットに関する研究は世界中で盛んに行われており、大きく分けて陸上 を動き回る陸上ロボット,水中や海中に潜水する水中ロボット,空中を飛ぶ飛行 ロボットの3つのタイプに分類される.陸上ロボットはその3つの中でも最も事 例が多く, 車輪型(1,2)やクローラ型(3,4), 脚式(5,6), 人型(7,8)など, 様々なものが 研究されている. 陸上ロボットを用いた研究の1つの目的はロボットのナビゲー ションを行うことであり,自己位置推定に基づいてロボットの位置制御を行うも の (9) や , 自己位置推定と同時に地図作成を行う SLAM(Simultaneous Localization and Mapping) に関するもの (10, 11, 12) が盛んに研究されている. 水中ロボットは機体 の防水を行う必要がるため、陸上ロボットと比べて実機製作の事例は少ないもの の,スクリューを用いたもの^(13,14)や水中の生物を模倣したもの^(15,16)など,さま ざまなタイプのものが提案されている、飛行ロボットについては、小型かつ高出 力のモータが比較的安価に手に入るようになったため,近年実機の開発が盛んに 行われている.特に,従来より研究されていたヘリコプターのような2つのロータ のもの (17) だけでなく,より運動性の高い4つのロータを有する X4 フライヤー (18) やクアッドロータ (19,20) などと呼ばれる飛行ロボットが注目されている. さらに, これらの移動ロボットを単体で用いるのではなく、複数用いる研究も行われてい る、複数のロボットを協調させることで単体では行うことのできない複雑な作業 を行うことが可能となり、また、広範囲を探索する場合は、より短時間で効率よ く探索を行うことができる、複数のロボットによる協調に関する研究事例として、 山田と斎藤⁽²¹⁾は1台では押し動かすことのできない箱を3台のロボットの協調に より押し動かすことに成功した. Huntsberger ら (22) は崖のある山岳地帯の調査を 目的として,2台のアンカーロボットと1台の降下ロボットが協力して崖を安全に 降りる手法を提案し、実機による実証を行った. Burgard ら (23) は屋内の地図作成 を最大5台の移動ロボットを用いて行い,使用するロボットの台数の増加に伴い地 図作成完了までに要する時間が短縮されることを示した.また,複数のロボット

を用いる場合にはロボットの移動を効率よく行うために編隊移動が行われる.移動ロボットの編隊移動に関する研究として,Kwon と Chwa $^{(24)}$ はベクトルフィールドを用いることで個々の経路計画を行うことなく分散的に編隊移動を行う手法を提案した.Savkin と Teimoori $^{(25)}$ は通信範囲が制限されているという条件下でそれぞれのロボットの向きを分散的に統一させる手法を提案した.Kato ら $^{(26)}$ は個々のロボットが予め指定された $^{(26)}$ は何々のロボットが予め指定された $^{(26)}$ はできるさまざまな隊形を形成する IET アルゴリズムを提案した.

上述のとおり、移動ロボットの関する研究は幅広い分野で盛んに行われているが、実際に導入されているのは工場や倉庫などに限られており、家庭やオフィスなどにはまだロボットが普及しているとは言い難い、工場や倉庫などはロボットや作業内容に合わせて設計することが可能であり、外乱の少ない環境を構築することができる、一方で、家庭やオフィスは人間主体の環境であり、その上、人間の活動のために予期せぬ外乱がそこには溢れている、したがって、後者のような環境にロボットを導入するためには、その環境に合わせたロボットを設計する必要があり、特に外乱に対するロバスト性が要求される、これらの要求を実現することは容易ではないため、家庭やオフィスにロボットが普及していないものと思われる、

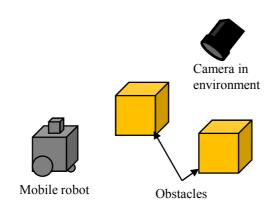
典型的な従来の移動ロボットの制御手法は,何らかの方法でロボットの自己位置を推定し,その位置を基にロボットを制御するものであった.自己位置推定の方法として車輪の回転数を基にしたオドメトリ (27) や加速度センサと角速度センサを用いた慣性航法 (28) などが挙げられる.これらの手法を人間で例えると,どちらの方向に何歩歩いたかを数えながら進んでいるようなものと思われるが,我々人間はそのように正確な自己位置を基に行動しているとは考えられない.また,人間は外乱で溢れた動的な環境でも,状況に応じて臨機応変に対応することができる.人間の行動は視覚に深く依存していると言われており,暗闇では明かりを灯すことからも視覚が重要であることがわかる.視覚からは物体の色,形やその物体までの大まかな距離など,さまざまな情報を得ることができ,人間はこれらの情報を基に状況を把握し臨機応変な行動を行っているものと考えられる.そこで,視覚と同様の情報を得ることのできるカメラを用いてロボットを制御することで人間のように臨機応変なロボットの行動を実現できるのではないかと考えられ,カメラを用いた視覚サーボに関する研究が盛んに行われている (29,30,39).

我々人間の思考は言語によって行われていると考えられ,行動も同じように言語による思考で決定されていると考えることができる.また,言語は数字とは異なり,あいまいさを持つものである.このようなあいまいさを持つ言語を扱うことのできる制御手法としてファジィ制御が提案され,主に非線形なシステムの制御に広く利用されている (32).ファジィ制御は 1965 年に Zadeh (33) によって提唱されたファジィ理論に基づいたものであり,1975 年の中頃から制御の分野への応用が研究され始めた (34).ファジィ制御器は「大きい」や「速い」などの言語表現を



(a) Eye-in-hand system

(b) Mobile robot equipped with a camera



(c) Camera in environment

Fig. 1.1: Examples of visual servoing for robot control

扱うことができ,例として自動車の運転を挙げると「もし右カーブに近づいたら,速度を遅くしてハンドルをゆっくり右に切る」というような IF-THEN ルールの形式で記述された制御規則に基づいて制御を行う.したがって,人間が経験的に会得した制御規則をロボットの制御に利用することが可能な手法であり,人間のように臨機応変な行動を行うことを目的とした視覚サーボを実現するためにファジィ制御器を用いることは有用であると考えられる.

1.2 視覚サーボ

視覚サーボはロボットに搭載されたカメラ,あるいは環境に設置されたカメラの画像を基にロボットの動作を制御する手法である。図 1.1 に視覚サーボの 3 つの例を示す。図 1.1(a) はロボットハンドにカメラが搭載されたハンドアイシステムを示しており,視覚サーボの典型的な例である。このシステムでは,カメラに映った対象物の姿勢などを基にロボットハンドの制御を行う。図 1.1(b) は移動ロボット

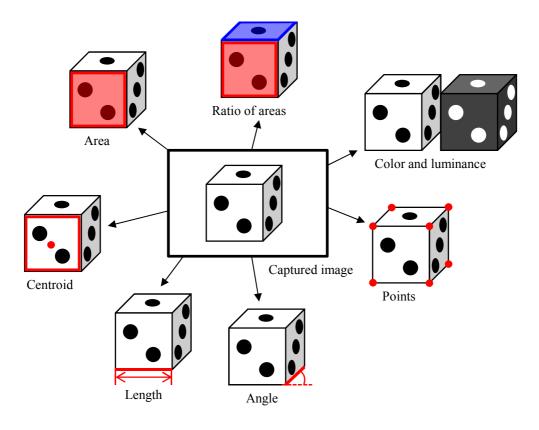


Fig. 1.2: Examples of extracted image features from a captured image

にカメラを搭載し、得られたカメラ画像を基にそのロボットを制御するシステムである.この例では、壁に描かれた人工的な特徴点を基に移動ロボットを目標の位置へと移動させることを行っている.特にこの移動ロボットの視覚サーボは近年盛んに行われており、その要因として Web カメラなどの登場により個人でも安価にイメージセンサを手に入れることが可能になったことが挙げられる.図 1.1(c)は、環境にカメラが設置されている場合の視覚サーボの例を示している.この場合、カメラを基準とした絶対座標上のロボットの位置を容易に取得することができるため、移動ロボットのナビゲーションなどに有用なシステム構成であると考えられる.

人間はカメラ画像を見ただけで瞬間的に映っている人や物を判別することができるが,コンピュータの中では数値が羅列されたデータであるため様々な画像処理を行って情報を抽出しなければならない.画像処理によって得られる画像特徴量の例を図 1.2 に示す.図の中央に取得したカメラ画像があり,その周囲に抽出された特徴量を示しており,色や明るさだけでなく,点,面積,重心,長さ,角度など,様々な情報が画像より抽出できることが分かる.視覚サーボでは,これらの特徴量を基にしてロボットの制御を行う.

視覚サーボに関する最初の文献は 1979 年の Agin (58) の技術報告書だと言われて

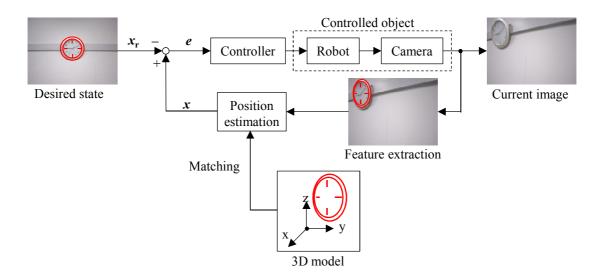


Fig. 1.3: Typical block diagram of a position-based visual servoing

おり、その中ではプロジェクターによってパターン投影を行い、カメラ画像より対象物までの距離などを推定してロボットハンドを制御する手法が述べられている、IEEEの雑誌 Robotics and Automation に掲載されたロボットの視覚サーボに関する最初の論文はWeissら⁽⁵⁹⁾のものであり、そこでは視覚サーボの枠組みを明確化し、「位置に基づく視覚サーボ (Position-based visual servoing)」と「画像に基づく視覚サーボ (Image-based visual servoing)」とに分類して、それぞれの手法の特徴を述べている。この分類は現在でも用いられており、2006年に掲載された Chaumette とHutchinson (35,36) の視覚サーボに関する調査論文もこの分類に基づいている。以下にそれぞれの手法の詳細について述べる。

1.2.1 位置に基づく視覚サーボ

位置に基づく視覚サーボは,カメラ画像の特徴量を基にロボットの位置を推定し,その位置を基にロボットを制御する手法である.図 1.3 に例として典型的な位置に基づく視覚サーボのブロック線図を示す.これは時計の正面にロボットを移動させることを目的とした場合のものである.前提条件として対象物の三次元モデルを持っており,カメラキャリブレーションによって正確なカメラヤコビアンが導出済みであることが必要である.図 1.3 に示すように,目標状態 $x_{\mathbf{r}}=(x_{\mathbf{r}}\ y_{\mathbf{r}}\ z_{\mathbf{r}}\ \phi_{\mathbf{r}}\ \theta_{\mathbf{r}}\ \psi_{\mathbf{r}})^T$ は対象物の三次元モデルを基にしたロボットの位置と姿勢である.現在のロボットの状態 $x=(x\ y\ z\ \phi\ \theta\ \psi)^T$ は制御対象であるロボットに搭載されたカメラの画像より対象物の特徴量を抽出し,その三次元モデルとのマッチングを行うことにより得られる.したがって,この場合の偏差 e は次式のように表わされる.

$$e = x - x_{\rm r} \tag{1.1}$$

つまり、制御器はロボットの状態そのものの誤差を基に入力を決定することとなる.これは、これまで行われてきた移動ロボットの制御手法と同じ問題であり、従来の手法と比べて自己位置を推定するためのセンサがカメラに代わっただけのように感じられる.したがって、従来の制御理論を容易に適用することが可能であり、高精度な位置制御を行うことができる.一方で、自己位置推定を行うための三次元モデルのマッチングには大きな計算負荷がかかるため、処理能力の高い計算器が要求されることや、ロボットの移動速度を制限するなどの工夫が必要である.

位置に基づく視覚サーボに関する研究として, Thuilotら (37) は直交座標型ロボッ トのエンドエフェクタにカメラを搭載し、移動する対象物をカメラの視野内に維 持する方法を示した、そこでは対象物の三次元モデルと取得した画像とを比較す ることでエンドエフェクタの位置を推定している.また,小川ら ⁽³⁸⁾ は,予め3眼 カメラを用いて作成した三次元環境地図を利用して、移動ロボットに搭載した単 眼カメラより得られた画像からロボットの自己位置を推定しナビゲーションを行っ た.Bajracharyaら (39) は火星探査ロボットのためのマニピュレータの位置制御を 行う高精度の位置に基づく視覚サーボを行った. Ha ら (40) はカメラを搭載したマ ニピュレータによって初期位置と目標位置を結んだ直線上を追従するための制御 器を設計し、実機実験による評価を行った、Cherubiniら(41)は非ホロノミック移 **動ロボットによる軌道追従問題について、カメラ画像のいずれかの縁に目標軌道** が交差しているという条件の下で、その交点からカメラの位置を推定し、その位 置を基にロボットを制御する手法を提案し、実機実験による評価を行った. Wang ら(42)は,カメラを搭載したマニピュレータによって,移動する対象物を追従する ために、対象物の三次元位置を推定しその位置を基にマニピュレータを制御する 手法を提案した.

1.2.2 画像に基づく視覚サーボ

画像に基づく視覚サーボの手法は,カメラ画像の特徴量から直接ロボットの制御入力を決定する制御手法である.したがって,位置に基づく視覚サーボの手法と比べて計算コストが少ないという利点があり,また,自己位置を参照することなく画像の情報を基に行動を決定するという点で,より人間的な情報処理を行う制御手法であると思われる.図 1.4 に例として典型的な画像に基づく視覚サーボのブロック線図を示す.これは前節と同様に時計の正面にロボットを移動させることを目的とした場合のものである.画像座標として $\mathbf{x}_{\mathbf{i}}$ - $\mathbf{y}_{\mathbf{i}}$ 座標を定義すると,ここでは,現在の状態 \mathbf{x} は観測対象物である時計の領域の重心 $(x_{\mathbf{i}},y_{\mathbf{i}})$ とその面積 S で表され, $\mathbf{x}_{\mathbf{i}}=(x_{\mathbf{i}}y_{\mathbf{i}}S)^T$ とする.したがって,目標状態も $\mathbf{x}_{\mathbf{ir}}=(x_{\mathbf{ir}}y_{\mathbf{ir}}S_r)^T$ のように画像の特徴量として与えられ,このときの偏差 \mathbf{e} は以下の式となる.

$$e = x_{i} - x_{ir} \tag{1.2}$$

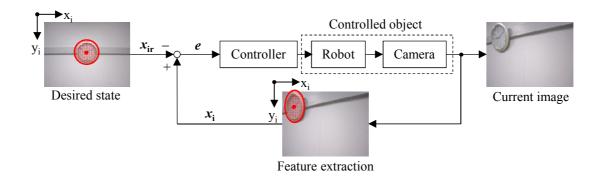


Fig. 1.4: Typical block diagram of an image-based visual servoing

したがって,制御器は画像特徴量の誤差を基にロボットへの制御入力を決定しなければならない.もし,次式のように画像特徴量と制御入力の関係を表した運動学モデルのようなものを導出することができれば,モデルベースの制御を行うことが可能となる.

$$\dot{x}_{i} = Lu \tag{1.3}$$

ただし, $u=(v\ \omega)^T$ はロボットへの制御入力であり,三次元の各軸方向の並進速度をv,各軸周りの回転角速度を ω としている.このL は相互行列 (Interaction matrix) と呼ばれており $^{(35)}$,モデルベースの画像に基づく視覚サーボには必要不可欠なものである.ここで $\dot{x_i}=\dot{e_i}$ であれば,以下のように相互行列の逆行列を用いた制御器によってロボットを制御することが可能であると考えられる.

$$u = -\lambda L^{-1} \dot{e}_{i} \tag{1.4}$$

しかしながら,相互行列の導出は容易ではなく,この例でも対象物の重心 (x_i, y_i) と u との関係はカメラモデルとロボットの動作モデルを用いて導出できたとしても,対象物の面積 S と u との関係の導出は困難であることが分かる.

画像に基づく視覚サーボの手法に関する研究は盛んに行われており (43,44,45), Espiau (46) はロボットのモデル化やカメラキャリブレーションの誤差に対する画像に基づく視覚サーボのロバスト性を実験によって確かめた.また, Espiauら (47) は相互行列を導出し,点,直線,円などを追従対象とした場合を例にシミュレーションを行った.倉舗ら (48) はカメラを搭載した移動ロボットによって曲線追従を行うための制御器を逆最適制御によって設計し,そのロバスト性の高さを示した.この手法では画像平面の奥行きは一定としている.また,Lucaら (49) は注目画素の奥行きを推定するオブサーバを設計し,より正確に相互行列を求めることで対象物を見失うことなくロボットを制御する手法を提案した.このように,相互行列を用いたモデルベースの画像に基づく視覚サーボについて様々な研究が行われているが,注目画素の奥行きに関する仮定や推定が必要であり,仮定した場合には

誤差の原因となり,推定した場合には画像に基づく視覚サーボの利点である処理の軽さを損なうものと思われる.また,画像特徴量とロボットの動作を結びつける特殊な制御器を設計することが必要であり,実機による実現例は少なく,十分な検証が行われているとは言えない.

1.3 ファジィ制御器を用いたロボット制御

ファジィ制御器は制御規則に基づく手法であることから、初めは経験的に培われ てきた制御規則が確立しているプロセス制御に利用されていた(50).その後,ロボッ トの制御へと応用されるようになり、様々な手法が提案された、例として、Sugeno と Nishida (51) は超音波を用いて位置と姿勢を計測することのできる実験用の模型 自動車を製作し,経験的に制定した制御規則によってロボットを壁に沿って前進さ せることを行った. Tsai ら (52) は3輪の操舵型移動ロボットを制御対象として,目 標軌道とロボットの現在位置との誤差を基にした制御規則を作成し、これに基づい てロボットに軌道追従を行わせる手法を提案した.一方で,ファジィ制御器の特徴 として、制御規則は状態数とそれぞれの状態が取り得るラベルの数との組み合わ せだけ考えられるため、扱う状態数の増加に伴い制御規則の数が非常に多くなる という問題点がある、そこで、制御規則の自動的な生成を行う研究が行われてい る.例として, Bezine ら (53) は, 2 リンクと 3 リンクマニピュレータを制御対象と して,自動的に制御規則を生成する手法を提案した.特に,3リンクマニピュレー タに対しては,ファジィ制御器を2つに分解することで規則数を少なくする工夫 を施している.Seraji (54) は,路面の状態を基にした制御規則を作成し,凹凸の激 しい領域を避けてなめらかな路面を走行する手法を提案した. Alasty と Sepehri (55) はマニピュレータを用いて柔らかい物体を扱う際に物体に発生する振動を抑制す るためのファジィ制御器を設計した、この手法には扱う物体に関する予備知識が 必要ないという特徴がある.Burnら(56)は未知の物体を扱うマニピュレータにつ いて受動的に力制御を行うファジィ制御器を設計しシミュレーションと実機実験 による評価を行った.

ファジィ制御器を用いる利点として,制御対象のモデルが不明である場合でも,試行錯誤的に制御器を設計することが可能という点が挙げられる.本論文で取り組む画像に基づく視覚サーボでは,画像特徴量とロボットの動作の関係のモデル化は容易ではないため,ファジィ制御器を用いることは有用であると考えられる.

1.4 研究目的

前述のように,これまで提案されてきた画像に基づく視覚サーボの手法は相互 行列を用いたモデルベースのものが主流であった.しかし,我々人間は経験的に 相互行列を獲得しているものと考えられ,また,画像に基づく視覚サーボが人間 的な情報処理を行う手法であることから,本来,脳を模倣したニューラルネット ワークや経験に基づく行動を IF-THEN ルールとして表したファジィ制御など,モ デルフリーの制御手法と相性がいいものと思われる.

モデルフリーの制御手法を用いた画像に基づく視覚サーボの手法に関する研究として, Chenら (57) は倒立振子を制御対象とし,画像上で対象が存在する現在の領域と目標状態の領域との差を基に制御入力を決定するファジィ制御器を提案した.一方で,移動ロボットを制御対象としたモデルフリーの画像に基づく視覚サーボの手法は例が少なく,これに取り組むことで人間のようにさまざまな環境に対応できるロボットの実現に近づくことができるものと思われる.

本研究では、移動ロボットを制御対象として、ファジィ制御器を用いた画像に基づく視覚サーボを取り扱うこととし、実機による検証までを行うことを研究目的とする.ファジィ制御器は経験的に獲得した制御規則を利用できることから、人間が行う車の運転を参考に制御器を設計するために、制御対象として使用する移動ロボットには乗用車のような操舵型の4輪移動ロボットを用いることとする.問題設定として移動ロボット制御の1つである軌道追従制御に取り組み、地面に描かれた目標軌道をカメラを搭載した移動ロボットが追従することを制御の目標とする.まずは、直線軌道を追従目標とした場合について経験に基づくファジィ制御器を設計し、次に任意軌道を追従目標とした場合についてGAを用いたファジィ制御器の設計を行う.さらに、提案した画像に基づくファジィ制御器の応用例として、画像に基づく障害物回避を行う手法を展開する.

1.5 本論文の構成

本論文は6章で構成されており,各章の内容は以下のとおりである.第1章では移動ロボットや視覚サーボ,ファジィ制御に関する研究背景について述べ,視覚サーボの概要として,位置に基づく視覚サーボと画像に基づく視覚サーボの2つの手法の違いについて述べる.また,ファジィ制御器を用いたロボットの制御の関連研究を紹介し,本研究の目的を述べる.

第2章では,先行研究であるモデルベースの画像に基づく視覚サーボを紹介する.ここでは直線軌道追従について取り組む.この手法では,制御器を設計するために相互行列を導出する必要があるが,それはロボットに搭載されているカメラが鉛直下向きであるという条件の下に導出される.導出された相互行列を基にリヤプノフ直接法を用いて制御器を設計し,シミュレーションによる評価を行う.シミュレーションの結果より,設計した制御器によって目標の直線軌道を追従可能であることを示す.一方で,カメラの向きが変化した場合は回転行列を用いて座標変換を行う必要があるため,注目画素の奥行きを推定しなければならないという問題点がある.

第3章では直線軌道追従を問題設定として,ファジィ制御器を用いた画像に基づく視覚サーボの手法を提案する.この手法では,相互行列を用いることなく制

御器を設計することが可能である.まず,制御対象を自動車のような操舵型の4輪移動ロボットとして,人間が自動車の運転を行う際のハンドル操作と加減速を基にした制御規則を作成し,シミュレーションによって速度制御を行うことの有用性を示す.次に,画像に映った目標軌道の傾きと切片の値とその変化量を状態変数とするファジィ制御器を設計し,シミュレーションと実機実験の両方を行う.両方の結果より,わずかに残留誤差が残るものの,目標軌道に沿ってロボットが直進することが可能であることを示す.

第4章では任意軌道追従を問題設定とし、より複雑なファジィ制御器を GA によって設計する手法を提案する。GA を用いた制御器の設計では、評価用の目標軌道に特化した制御器が設計される傾向にあるが、ここでは、設計時のシミュレーションにノイズを付加することによって制御器のロバスト性を向上させることを試みる。このようにして設計した制御器を用いたロボットの挙動をシミュレーションと実機実験で比較し、適切な種類と大きさのノイズを付加することで、実機の場合でも目標軌道を追従することが可能であることを示す。

第5章では画像に基づくファジィ制御器の応用例として,画像に基づく障害物回避手法を提案する.これは,カメラ画像より障害物検出を行い,その画像上にその障害物を回避する軌道を生成し,その軌道を画像に基づくファジィ制御器によって追従することで障害物回避を行う手法である.障害物検出はエッジ検出を用いて行い,目標軌道は検出されたエッジに基づくポテンシャルフィールドを用いて生成する.実機実験によって検証を行い,提案手法によって障害物回避を行うことが可能であることを示す.

第6章では各章で述べた内容を総括し,本論文のまとめを行う.

第2章 相互行列を用いたモデルベー スの画像に基づく視覚サーボ

本研究ではファジィ制御器を用いた画像に基づく視覚サーボの手法を提案するが,本章ではモデルベースの手法との違いを示すために,相互行列を用いたモデルベースの画像に基づく視覚サーボの手法について述べる.ここでは,倉舗ら (48) の提案したリヤプノフ直接法に基づく制御器の設計手法について紹介する.

2.1 問題設定

ここでは,図 2.1 のように直線の目標軌道は地面に描かれており,移動ロボットにはカメラが取り付けられているとする.この移動ロボットがカメラ画像を基に目標軌道を追従することを目標とする.世界座標は追従目標である直線に沿った方向に x 軸,これに対して直角に交わるように y 軸をとり,ロボットの状態は位置 (x,y) と,x 軸を基準としたロボットの向き θ を用いて (x,y,θ) で表すこととする.ここで使用するロボットは制御器の導出を簡単化するために,図 2.2 に示す 2 輪独立駆動型の移動ロボットとする.並進速度を s,回転角速度を ω とすると,このロボットの運動学モデルは以下のように表わされる (60).

カメラモデルにはピンホールモデルを用いることとし,カメラ座標を F_c とし,画像座標をu-v 座標として図 2.3 のように定義すると, F_c 内のある点 (x_c,y_c,z_c) は次式によって画像座標の点 (u,v) に投影される $^{(61)}$.

$$l\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{c} \\ y_{c} \\ z_{c} \end{bmatrix}$$
$$= \mathbf{PM}$$
 (2.2)

ただし,画像座標はカメラ座標の x_c-y_c 平面と平行であり,カメラ座標の原点から z_c 軸方向にlだけ離れた距離にあるものとする.また,fはカメラの焦点距離であ

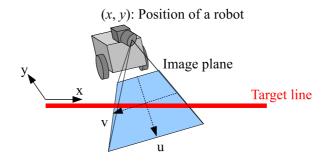


Fig. 2.1: Difinition of coordinates

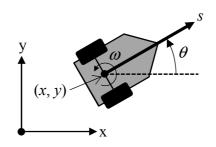


Fig. 2.2: Robot model

り,u-v 座標の原点はカメラ座標の z_c 軸上にあるものとする.カメラが鉛直下向きを向いている場合は,世界座標の x-y 平面とカメラ座標の x_c - y_c 平面が平行であるため,画像特徴量とロボットの動作を結びつける相互行列の導出が容易になる.もし,カメラの向きが傾いていたとしても,次式のようにカメラの回転角ロール ϕ ,ピッチ θ ,ヨー ψ を用いた回転行列 R によって,鉛直下向きのカメラ座標を求めることで,上記の仮定に基づいて設計した制御器を適用することが可能となる.

$$\begin{bmatrix} x_{c} \\ y_{c} \\ z_{c} \end{bmatrix} = lRP^{-1} \begin{bmatrix} u_{a} \\ v_{a} \\ 1 \end{bmatrix}$$
 (2.3)

ただし, (u_a, v_a) はカメラの向きが鉛直下向きでない場合の,画像座標上のある 1 点の座標値であり,回転行列 R は次式のとおりである.

$$\mathbf{R} = \begin{bmatrix} \cos\phi\cos\theta & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \sin\phi\cos\theta & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi \\ -\sin\theta & \cos\theta\sin\psi & \cos\theta\cos\psi \end{bmatrix}$$
(2.4)

したがって,これ以降,カメラは鉛直下向きを向いているという仮定の下で議論 を進める.

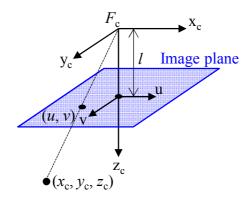


Fig. 2.3: Camera model

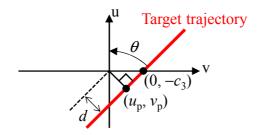


Fig. 2.4: Coordinate of the image plane

2.2 相互行列の導出

画像座標は図 2.4 に示すように , 画像中心を原点とする u-v 座標とする . 仮定よりカメラは鉛直下向きを向いているため , u-v 座標の原点は世界座標におけるロボットの位置 (x,y) と一致する . また , 図 2.4 は画像に映った目標軌道を示しており , その軌道と u 軸とのなす角 θ はロボットの姿勢 θ と一致する .

追従目標である直線は, u-v 座標系において一般的に以下の方程式で表わされる.

$$a_1 u + a_2 v + a_3 = 0 (2.5)$$

ここで , 直線が v 軸と平行でないと仮定すると , $a_2 \neq 0$ であるため , 式 (2.5) を a_2 で除算し , 以下の式が得られる .

$$c_1 u + v + c_3 = 0, \quad \left(c_1 = \frac{a_1}{a_2}, \quad c_3 = \frac{a_3}{a_2}\right)$$
 (2.6)

 c_1 は直線の傾きに関する係数であり, c_3 は図 2.4 に示すように直線と v 軸との交点の v 座標値の符号を反転したものである.なお, c_1 については以下の式が成り立つ.

$$c_1 = -\tan\theta \tag{2.7}$$

ただし,直線が v 軸と平行でないという仮定より $-\pi/2 < \theta < \pi/2$ である.この直線を追従させるためには,直線とロボットとの距離を 0 にしなければならない.世界座標において,それはロボット位置 (x,y) の y 座標値そのものであり,世界座標の y 座標値と画像上の直線より得られる c_3 の関係は幾何学的に以下のように求められる.まず,画像座標上の直線と原点の距離 d を求めると次式のようになる.

$$d = -c_3 \cos \theta \tag{2.8}$$

ここで,カメラが取り付けられている高さを h とすると l=h となり,d と y 座標値の関係は d=yf/h であるため,これを代入し式変形すると, c_3 は以下のように表わされる.

$$c_3 = -\frac{1}{\cos \theta} \cdot \frac{yf}{h} \tag{2.9}$$

これより,画像平面に射影された直線の方程式の係数である c_1 , c_3 をそれぞれ 0 にすることで,世界座標ではロボットの位置の y 座標値 y=0 かつロボットの向き $\theta=0$ が実現されることがわかる.したがって,これ以降, c_1 , c_3 を時間 t に関する変数として $c_1(t)$, $c_3(t)$ とみなし,これらを状態変数として扱うこととする.

次に , 相互行列を導出するために , 式 (2.7) と式 (2.9) をそれぞれ時間微分すると以下のようになる .

$$\dot{c}_{1} = -\frac{1}{\cos^{2}\theta}\dot{\theta}$$

$$\dot{c}_{3} = -\frac{1}{\cos^{2}\theta}\sin\theta \cdot \dot{\theta} \cdot \frac{yf}{h} - \frac{1}{\cos\theta} \cdot \frac{\dot{y}f}{h}$$
(2.10)

ここで , ロボットモデルより $\dot{\theta}=\omega$, $\dot{y}=s\sin\theta$ であり , 式 (2.7) と式 (2.9) の関係を用いて式変形を行うと , この問題設定における相互行列を以下のように導出することができる .

$$\begin{bmatrix} \dot{c}_1 \\ \dot{c}_3 \end{bmatrix} = \begin{bmatrix} 0 & -(c_1^2 + 1) \\ \frac{f}{h}c_1 & -c_1c_3 \end{bmatrix} \begin{bmatrix} s \\ \omega \end{bmatrix}$$
 (2.11)

上式と相互行列の定義式である式 (1.3) を見比べると,確かに同じ形であることがわかる.

一方, c_1 , c_3 は,画像上の直線上にある任意の 2 点をそれぞれ (v_1, u_1) および (v_2, u_2) とすると,次式より求めることができる.

$$\begin{bmatrix} c_1 \\ c_3 \end{bmatrix} = \frac{1}{u_2 - u_1} \begin{bmatrix} v_1 - v_2 \\ -u_2 v_1 + u_1 v_2 \end{bmatrix}$$
 (2.12)

ただし,直線が v 軸と平行ではないという仮定より $u_2-u_1\neq 0$ である.画像処理によって直線を抽出しその中の 2 点を選択することは難しいことではないため,実際にカメラ画像より c_1,c_3 を抽出することは可能であると考えられる.

2.3 リヤプノフ直接法を用いた制御器の設計

本節では,非線形システムの制御器設計に広く利用されているリヤプノフ直接法を用いた制御器の設計を行う (付録 A を参照).この方法では,状態量とみなした c_1,c_3 に関するリヤプノフ関数を見つけることができれば,それぞれの値を共に0 とすることが可能となる(つまり,レギュレータ問題に対する)制御器を設計できる.

まず, c_1,c_3 に関するリヤプノフ関数の候補として次式に示す $V(c_1,c_3)$ ついて考える.

$$V = \frac{1}{2} \left(k_1 c_1^2 + k_3 c_3^2 \right) \tag{2.13}$$

ただし, k_1 および k_3 は正のゲイン定数である.式 (2.13) の形を見ると, V>0 となることは明らかである.これを時間微分し,出てきた $\dot{c_1}$, $\dot{c_3}$ に相互行列 (式 (2.11)) の関係を代入すると,次式のようになる.

$$\frac{dV}{dt} = \frac{fs}{h}k_3c_1c_3 - c_1\{k_1(c_1^2 + 1) + k_3c_3^2\}\omega$$
 (2.14)

ここで, \dot{V} <0となるように,入力 ω を以下のようにとる.

$$\omega = \left\{ k_1(c_1^2 + 1) + k_3 c_3^2 \right\}^{-1} \left(\frac{fs}{h} k_3 c_3 + k_2 c_1 \right)$$
 (2.15)

ただし, k_2 は正のゲイン定数である.式 (2.15) を式 (2.14) に代入すると,次式のようになる.

$$\frac{dV}{dt} = -k_2 c_1^2 \le 0 {(2.16)}$$

ここで, Barbalat's lemma (62) より, 以下が示される.

$$\lim_{t \to \infty} \dot{V} = \lim_{t \to \infty} \left(-k_2 c_1^2 \right)$$

$$= 0 \tag{2.17}$$

$$\lim_{t \to \infty} c_1 = 0 \tag{2.18}$$

また,以下の関係が導かれる.

$$\lim_{t \to \infty} \dot{c_1} = -\lim_{t \to \infty} \left(k_1 + k_3 c_3^2 \right)^{-1} \frac{fs}{h} k_3 c_3$$

$$= 0 \tag{2.19}$$

したがって, $s \neq 0$ という仮定より, $c_3 \rightarrow 0$ となり,目標状態への収束が保証される.

2.4 シミュレーション

設計した制御器の性能評価を行うためにシミュレーションを行った.

条件

制御対象のロボットは 2 輪独立駆動型の移動ロボットとし,その動作モデルには式 (2.1) に示すものと同じものを用いた.ただし,サンプリング時間を $\Delta t=0.001$ [s] とし,式 (2.1) は 4 次のルンゲ・クッタ法で解いた.ロボットの初期姿勢は $(c_1,c_3)=(1.0,1.0)$ となるように, $(x,y,\theta)=(0$ [m],-0.71 [m],-0.78 [rad])とした.追従目標である直線は世界座標上で y=0 とした.また,カメラの焦点距離は f=1.0 [m],カメラは高さ h=1.0 [m] の位置に鉛直下向きに取り付けられているものとして,視野は考慮せずにカメラモデル(式 (2.2))に基づいて,目標軌道上の任意の 2 点を画像座標に射影したものを用いて (c_1,c_3) を求めることとした.制御器のゲインは $(k_1,k_2,k_3)=(0.1,3.0,10.0)$ とそれぞれ設定した.式 (2.15) では ω を決定するためには,予め前進速度 s の値を決めておかなければならないが,今回は軌道追従問題ということでロボットの並進速度は s=0.5 [m/s] で一定とした.設定された初期姿勢からロボットが目標軌道に近づき,y=0 および $\theta=0$ を維持することで軌道追従を達成したものとみなすこととした.

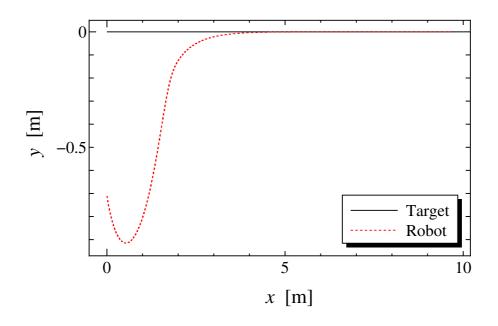
結果

シミュレーションの結果を図 2.5 に示す.図 2.5(a) は目標軌道とロボットの移動軌跡を示したものであり,ロボットが初期位置から少し目標軌道から離れるように移動しているが,その後,目標軌道に向かって移動し軌道上を進んでいることが分かる.図 2.5(b) はロボットの位置の y 座標値と向き θ の時間変化を示したものであり,両方の値がそれぞれ 0 に収束しているため,軌道追従を達成できたとみなすことができる.図 2.5(c) は画像上の目標軌道のパラメータの時間変化を示したものであり,それぞれの値が 0 へと収束していることが分かる.図 2.5(d) は制御入力の時間変化を示したものであり,前進速度 s は設定どおりに一定であり,旋回速度 s の変化によってロボットの動きが制御されていることが分かる.

考察

シミュレーションの結果より,画像に映った目標軌道のパラメータ (c_1,c_3) を状態量とみなし,それぞれ0にするように設計した制御器を用いて,ロボットが世界座標上でy=0および $\theta=0$ を維持し軌道追従を行うことが可能であると示された.一方で,相互行列はカメラが鉛直下向きに取り付けられているという仮定の下で導出されたものであり,カメラの向きが変化した場合は回転行列を用いて

座標変換を行う必要がある。座標変換を行うためにはlの値を推定する必要があるが,これは式(2.2)のカメラモデルより $l=z_c$ であり結局,カメラ座標における奥行き z_c を推定する必要があることがわかる。しかしながら,これでは画像に基づく視覚サーボの特徴である,画像特徴量から直接制御入力を決定するという簡便さが失われてしまう。一方で,相互行列は上記のような仮定なしに導出することは容易ではなく,モデルベースの画像に基づく視覚サーボを行う上での問題点であると言える.



(a) Trajectory of the robot

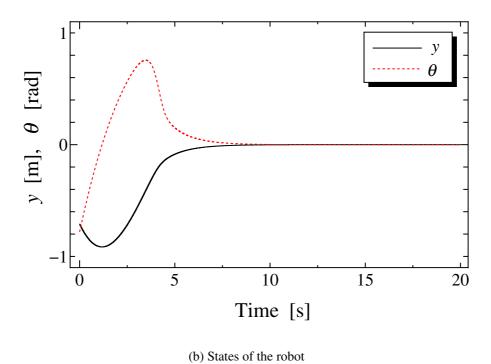
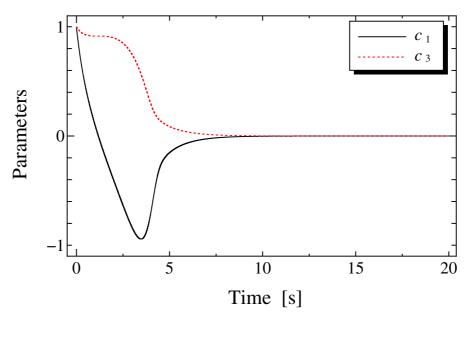
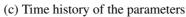


Fig. 2.5: Simulation results for the model based control method





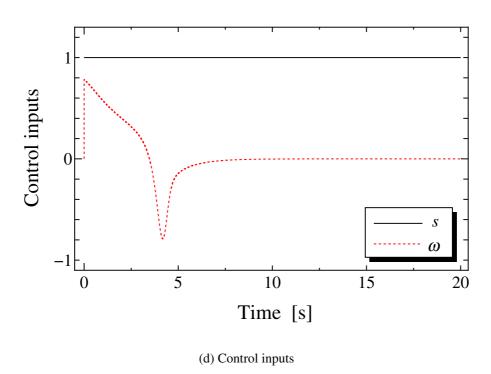


Fig. 2.5: Simulation results for the model based control method

2.5 まとめ

本章では、倉舗らの提案したリヤプノフ直接法を用いた画像に基づく制御器の設計手法について述べ、相互行列を用いたモデルベースの画像に基づく視覚サーボの特徴について述べた。また、シミュレーションによって設計した制御器の性能を評価し、画像に映った目標軌道のパラメータを基にロボットを制御することで、確かに目標軌道を追従することが可能であることを示した。一方、相互行列を導出するためにはカメラが鉛直下向きであるという厳しい仮定を設定する必要があり、カメラの向きが変化した場合は回転行列を用いて座標変換を行う必要があった。しかしながら、この座標変換を行うためには注目画素の奥行きを推定する必要があるため、計算量が少ないという画像に基づく視覚サーボの利点を損なってしまう恐れがある。

第3章 画像に基づくファジィ制御器 を用いた直線軌道追従

前章では相互行列を用いたモデルベースの画像に基づく直線軌道追従の手法について述べた.相互行列の導出には厳しい条件設定が必要であり,また,この手法を実機に適用するためには注目画素の奥行きを推定する必要があるため,画像に基づく視覚サーボの利点を損なう恐れがあることを指摘した.そこで,本章では相互行列を用いることなく画像に基づく視覚サーボを行うために,ファジィ制御器を用いた手法を提案する.はじめに問題設定について述べ,人間が行う自動車の運転を基にしたファジィ制御器の設計を行い,シミュレーションによる評価を行う.次に,状態変数の変化量を考慮したファジィ制御器を設計し,シミュレーションと実機実験の両方により評価する.

3.1 問題設定

ここでは,問題設定として移動ロボットによる直線軌道追従を行うこととする.図 3.1 は環境と座標の定義を示している.この図に示すように目標軌道は地面に描かれており,移動ロボットは搭載されたカメラを用いてその直線を観測する.世界座標として目標軌道に沿った方向を x 軸とする x-y 座標を定義する.したがって,目標軌道は世界座標上で y=0 の直線として表すことができる.画像座標は u-v 座標で表現することとし,図 3.1 のように軸を設ける.画像座標上の直線の式は傾き α と切片 β を用いて $v=\alpha u+\beta$ と表現する.なお, β の値については画像の横幅 W によって正規化した値 $2\beta/W$ を用いることとする.

人間が行う自動車の運転方法を基にファジィ制御器の制御規則を作成するために,制御対象として図 3.2 に示すように操舵型の 4 輪移動ロボットを用いることとする.ロボットの状態は世界座標上の位置 (x,y) と,ロボットの進行方向と x 軸とがなす角 θ を用いて, (x,y,θ) で表現するものとし,制御入力は前進速度 x と前輪の操舵角 x とする.また,前輪と後輪の車軸間距離を x とする.

目標軌道を追従することはロボットの状態が y=0 かつ $\theta=0$ を満たすことであり,この場合,カメラ画像には目標軌道が画面の中央に縦に映っているものと直感的に推測できる.このとき,画像座標上では $\alpha=0$ かつ $\beta=0$ であるため,直線のパラメータ α と β が共に 0 へと収束するようにロボットを移動させることで y=0

かつ $\theta=0$ となり,ロボットに目標軌道を追従させることができると考えられる.したがって, α と β を共に0 とする制御器を設計しなければならない.図 3.3 は提案する画像に基づく直線軌道追従制御手法のブロック線図を示している.この図から,制御器はカメラ画像より抽出された直線のパラメータを基にロボットへの入力を決定しており,自己位置を参照することなくロボットを制御していることがわかる.カメラ画像に映った目標軌道のパラメータを抽出するためにハフ変換を用いる.

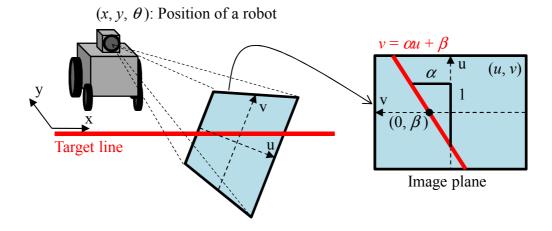


Fig. 3.1: World coordinate and image coordinate

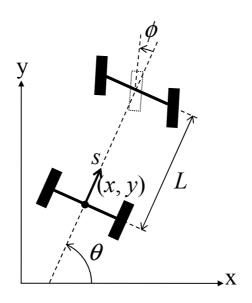


Fig. 3.2: Model of a mobile robot

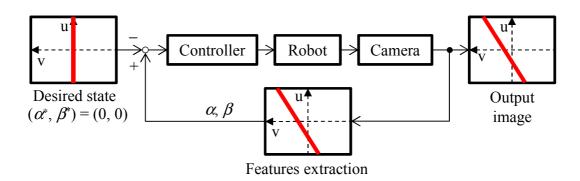


Fig. 3.3: Block diagram of the image-based control

3.2 ハフ変換を用いた直線軌道の抽出

ハフ変換 $^{(63)}$ は画像から直線を始めとする任意の形態を抽出する手法として画像処理の分野で広く利用されている.ここでは,ハフ変換を用いてカメラ画像に映った目標軌道の傾き α と切片 β を求める方法について述べる.

ハフ変換を用いて直線抽出を行う際には,直線のパラメータ空間 $(\alpha$ - β 平面)を利用する.図 3.4 は八フ変換の基本的な概念を示したものである.画像の u-v 座標上に存在する点 p_1 から p_4 は画像に映った目標軌道上の点であるとする.これらの点はパラメータ空間に射影すると,図に示すように直線 m_1 から m_4 として表現される.これらの直線は u-v 座標上で対応する点を通る直線の集合を表している.したがって,点群が同一直線上にあるとき,パラメータ空間ではそれぞれの直線がある 1 点で交わることとなる.パラメータ空間のある 1 点は u-v 座標上のある 1 つの直線を表すため,図 3.4 に示すパラメータ空間の交点 (α_0,β_0) は u-v 座標上で点 p_1 から p_4 を通る直線を表すこととなる.このように,点群をパラメータ空間に直線群として射影し,その直線の交点を求めることで,点群を通る直線の傾きと切片を求めることができる.実際には点群は完全に同一直線上に存在するわけではないので,パラメータ空間をある分解能で区切って離散化した上で交点を求める必要がある.

以上がハフ変換の基本的な原理であるが,パラメータの値域は $-\infty < \alpha < \infty$ および $-\infty < \beta < \infty$ であるため,交点を求めるために用意すべき 2 次元配列の大きさが無限大となってしまう.そこで,直線の方程式を次式のように変更することを考える.

$$\rho = u\cos\theta + v\sin\theta \tag{3.1}$$

ここで, ρ は図 3.5 に示すように原点から直線に下ろした垂線の長さであり,その交点を (u,v) とし, θ はその垂線と u 軸とのなす角である.この方程式を用いて直線を表現することで, θ の値域は $0 \le \theta < \pi$ であり有限となる.また, ρ については画像座標の原点の位置にも依存するが,画像の横幅を W,縦幅を H の無次元の値とすると最大値は画像の対角線の長さ $\sqrt{W^2+H^2}$ であるため,値域は $0 \le \rho < \sqrt{W^2+H^2}$ となり有限である.したがって,パラメータ空間を ρ - θ 平面として八フ変換を行うことで,有限の 2 次元配列を用いて交点を求めることができ,コンピュータによってプログラミングを行うことが可能となる.このパラメータ空間では,u-v 座標上のある 1 点 (u_i,v_i) は次式のように正弦波として射影される.

$$\rho = \sqrt{u_i^2 + v_i^2} \sin(\theta + \theta_i), \quad \theta_i = \tan^{-1} \frac{v_i}{u_i}$$
(3.2)

したがって,図 3.6 に示すようにパラメータ空間で正弦波の交点 (ρ_0, θ_0) を求めることで,u-v 座標上の目標軌道を $\rho_0 = u\cos\theta_0 + v\sin\theta_0$ として抽出することができる.また,上記の異なる直線の方程式のパラメータ間の関係式は以下のとおりで

ある.

$$\alpha = -\frac{1}{\tan \theta}, \quad \beta = \frac{\rho}{\cos \theta}$$
 (3.3)

ハフ変換を用いて目標軌道の ρ と θ を求めた後,上式の関係を用いて α と β を求めることができる.

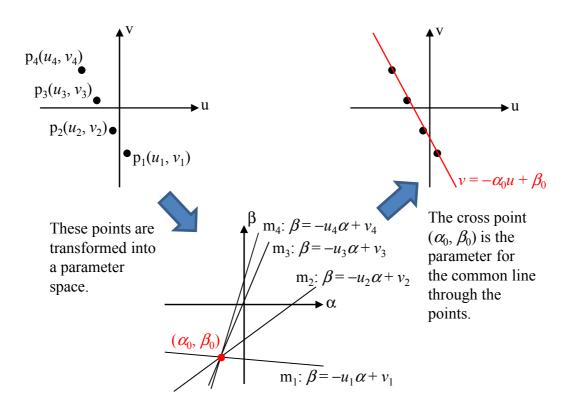


Fig. 3.4: Concept of the Hough transform for extracting a line

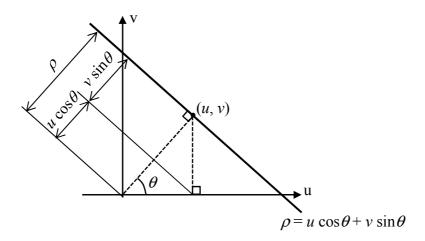


Fig. 3.5: Line representation using parameters ρ and θ

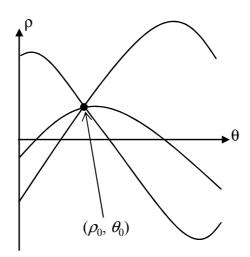


Fig. 3.6: Parameter space for ρ and θ

3.3 人間が行う自動車の運転に基づいたファジィ制御器

2章で述べたモデルベースの制御手法では,ロボットの前進速度sをある一定値として仮定しない限り旋回速度 ω を決定することができなかった.しかし,ファジィ制御器は制御規則に基づいて制御入力を決定する手法であるため,ロボットの前進速度に関する制御規則を作成することができれば,前進速度を時変として扱うことが可能となる.ここでは,人間が行う自動車の運転を参考にファジィ制御器の制御規則を作成することで,前進速度を時変として扱い,速度制御を行うことの有用性を示すこととする.

我々人間が自動車を運転する際には,運転席から見える道路の向き,現在のハンドルの角度および目標速度と現在の速度との差を基に,ハンドルの回転量とアクセルの加減を決定しているものと考えられる.この仮定に基づいて,操舵型の4輪移動ロボットによって直線軌道追従を行うためのファジィ制御器を設計することとする.画像処理によって求められた目標軌道の傾き α と切片 β は運転席から見える道路の向きに例えることができる.また,ハンドルの角度は移動ロボットでは前輪の操舵角 ϕ に置き換えられ,速度については目標速度を $s_{\rm r}$ とすると現在の速度 s と目標速度との差 $s_{\rm e}$ は $s_{\rm e}=s-s_{\rm r}$ として容易に求めることができる.同様に移動ロボットの制御入力は前輪の操舵角速度 ϕ と加速度 s とみなすことができる.したがって,状態変数を $(\alpha,\beta,\phi,s_{\rm e})$ とし,これを基に制御入力 (ϕ,s) を決定するファジィ制御器を設計する.

3.3.1 ファジィ制御器の設計

ファジィ制御器の設計は状態変数をファジィ化・非ファジィ化するためのメンバシップ関数と,ファジィ化された状態変数を基に制御入力を決定するための制御規則をそれぞれ作成することにより行う.図 3.7 に各状態変数のメンバシップ関数を示す.それぞれの状態変数の値はメンバシップ関数を通して適合度を有する言語表現 (N: Negative,ZO: Zero,P: Positive) にファジィ化される.例として β = 1.0 のファジィ化について述べる. β =N, β =ZO, β =Pの適合度を求めるメンバシップ関数をそれぞれ $h_{\beta,N}(\beta)$, $h_{\beta,ZO}(\beta)$, $h_{\beta,P}(\beta)$ と表記する.図 3.7 のメンバシップ 関数より, β = 1.0 は $h_{\beta,N}(1.0)$ = 0.0, $h_{\beta,ZO}(1.0)$ = 0.75, $h_{\beta,P}(1.0)$ = 0.25 のように適合度を持った言語表現にファジィ化される.

ファジィ化された言語表現によって,大まかに場合分けされるロボットと直線との関係をもとに,出力 $\dot{\phi}$, \dot{s} を設定し制御規則を制定する.表 3.1,表 3.2 はそれぞれ $\dot{\phi}$, \dot{s} に対する制御規則である.例として表中の1番のルールは

if
$$\alpha = N, \beta = ZO, \phi = N$$
 then $\dot{\phi} = ZO$ (3.4)

と解釈する.ただし, ϕ は各状態変数が0となるように人間が行う自動車の運転を参考にしながら設定した. \dot{s} はほとんどの場合,目標速度となるように加速度を発

生させているが, β = ZO 付近で α が ZO でない場合は速度を落とし,移動速度に対する操舵角の変化量を大きくすることで旋回性の向上を図る.出力値を求める際に利用する各ルールの適合度には,そのルールに関係するファジィ化された状態変数の適合度の最小値を採用する.その後,各ルールの適合度がその出力値の適合度となるが,同一の出力値が設定されているルールが他にも存在する場合は,そのうち最大の適合度を採用する (min-max 重心法 $^{(34)})$.

出力がファジィ集合の状態では操作量としてロボットへ与えることはできないため,非ファジィ化を行い数値として $\dot{\phi}$ を決定する.非ファジィ化はシングルトンを用いた簡略化推論法を用いることとし,以下のように適合度による重み付き平均を求めることで実数化する.図 3.7 に示す \dot{s} の N,P に対応する出力値は実機の性能を参考にして減速の方をより大きく設定している.

$$\dot{\phi} = \frac{h_{\dot{\phi},N} \times (-1.75) + h_{\dot{\phi},ZO} \times 0 + h_{\dot{\phi},P} \times 1.75}{h_{\dot{\phi},N} + h_{\dot{\phi},ZO} + h_{\dot{\phi},P}}$$
(3.5)

$$\dot{s} = \frac{h_{\dot{s},N} \times (-6.0) + h_{\dot{s},ZO} \times 0 + h_{\dot{s},P} \times 3.0}{h_{\dot{s},N} + h_{\dot{s},ZO} + h_{\dot{s},P}}$$
(3.6)

ただし $\,h_{\dot{\phi},\mathrm{N}}\,$ は $\,\dot{\phi}=$ Nの適合度を表すものとする.

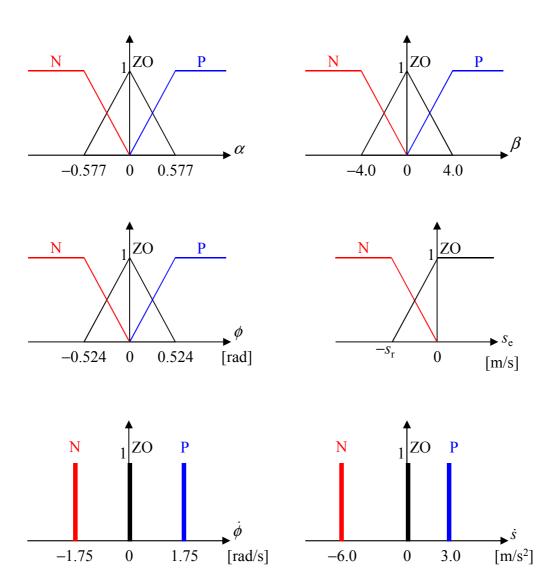


Fig. 3.7: Membership functions for the controller based on human driving techniques

Table 3.1: Fuzzy rules for $\dot{\phi}$ in the controller based on human driving techniques

Rule No.	α	β	ϕ	$\dot{\phi}$
0	N	N	_	N
1	N	ZO	N	ZO
2	N	ZO	ZO	N
3	N	ZO	P	N
4	N	P	N	P
5	N	P	ZO	ZO
6	N	P	P	N
7	ZO	N	N	ZO
8	ZO	N	ZO	N
9	ZO	N	P	N
10	ZO	ZO	N	P
11	ZO	ZO	ZO	ZO
12	ZO	ZO	P	N
13	ZO	P	N	P
14	ZO	P	ZO	P
15	ZO	P	P	ZO
16	P	N	N	P
17	P	N	ZO	ZO
18	P	N	P	N
19	P	ZO	N	P
20	P	ZO	ZO	P
21	P	ZO	P	ZO
22	P	P	_	P

Table 3.2: Fuzzy rules for \dot{s} in the controller based on human driving techniques

Rule No.	α	β	ϕ	s_e	\dot{s}
23	N	N	_	ZO	ZO
24	N	N	_	N	P
25	N	ZO	N	ZO	ZO
26	N	ZO	N	N	P
27	N	ZO	ZO	ZO	N
28	N	ZO	ZO	N	ZO
29	N	ZO	P	ZO	N
30	N	ZO	P	N	ZO
31	N	P	_	ZO	ZO
32	N	P	_	N	P
33	ZO	_	_	ZO	ZO
34	ZO	_	_	N	P
35	P	N	_	ZO	ZO
36	P	N	_	N	P
37	P	ZO	N	ZO	N
38	P	ZO	N	N	ZO
39	P	ZO	ZO	ZO	N
40	P	ZO	ZO	N	ZO
41	P	ZO	P	ZO	ZO
42	P	ZO	P	N	P
43	P	P	_	ZO	ZO
44	P	P	_	N	P

3.3.2 シミュレーション

人間が行う自動車の運転に基づいて設計したファジィ制御器の性能および,速度制御の有用性を評価するためにシミュレーションを行った.

条件

制御対象は図3.2に示す操舵型の4輪移動ロボットとした.このロボットの動作 モデルには次式を用いた.

$$\dot{x} = s \cos \theta
\dot{y} = s \sin \theta
\dot{\theta} = \frac{s \tan \phi}{L}$$
(3.7)

ロボットの性能に関するパラメータは表 3.3 に示す.カメラは高さ h=0.5 [m] の位置に鉛直下向きに取り付けられており,その焦点距離は f=0.02 [m],画角は 2.09 [rad],フレームレートは 30 [fps] とした.ロボットの初期位置は $(x,y,\theta)=(0.0$ [m], 1.0 [m], $\pi/4$ [rad]) とし初速度は $s_0=s_r$ で目標速度と一致しているものとした.画像上の直線の傾きは $-\infty<\alpha<\infty$ で表わされる.実機ではロボットの位置・姿勢によって直線を見失うことが考えられるが,今回のシミュレーションでは常に直線を観測できるものとしたため,画像座標上での直線の切片の範囲は $-\infty<\beta<\infty$ となる.世界座標でロボットが y=0 の状態を維持することで直線 追従を達成したものとみなすこととした.速度制御の有用性は操舵角のみで制御を行う場合と操舵角と速度の両方を用いて制御を行う場合を比較することで評価した.シミュレーションの時間間隔は 0.01 [s] とした.

Table 3.3: Robot parameters

Item	Value	
Steering angle (ϕ)	$-0.873 \sim 0.873$ [rad]	
Steering angular velocity $(\dot{\phi})$	$-1.75 \sim 1.75 \text{ [rad/s]}$	
Distance between front and rear wheel axes (L)	0.5 [m]	
Acceleration (\dot{s})	$-6.0 \sim 3.0 \text{ [m/s}^2\text{]}$	

結果

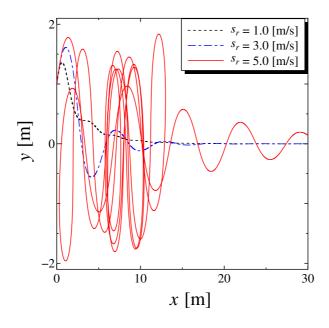
シミュレーション実験の結果を図 3.8(a) から図 3.8(e) に示す.図 3.8(a) と図 3.8(b) はそれぞれ操舵角のみで制御した場合のロボットの移動軌跡とその時の v 座標値

の時間応答を示している.同様に,図 3.8(d),図 3.8(c) は操舵角と速度の両方を用いて制御した場合を示している.図 3.8(b) の移動軌跡より操舵角のみの制御ではロボットの目標速度が速くなるにつれてオーバーシュートが大きくなり,目標速度が 5.0 [m/s] の場合は操舵角の操作が間に合わず,制御不能になっている.一方,操舵角と速度の両方を用いて制御した場合はオーバーシュートが軽減され,目標速度が 5.0 [m/s] の場合でも目標軌道を追従できていることが分かる.図 3.8(e) は速度制御を行う場合のロボットの並進速度の時間変化を示している.いずれの目標速度の場合もシミュレーションの開始直後に減速し,目標軌道に収束する時刻付近に再び目標速度へと到達していることが分かる.

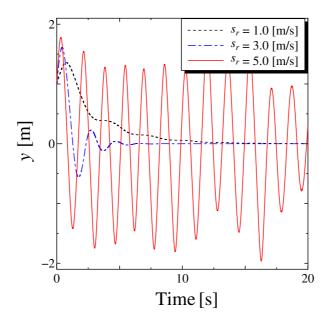
考察

実験結果より、設計したファジィ制御器によって移動ロボットの軌道追従を行うことが可能であることを示した.また、速度制御を行うことでオーバーシュートを軽減し、操舵角による制御のみでは行うことのできなかったロボットの目標速度が 5.0 [m/s] の場合でも、目標軌道を追従できることを可能とした.図 3.8(a)と図 3.8(c)を比べると目標速度がいずれの設定値であっても、速度制御を行うことで同等かより短い距離で目標軌道へ収束していることが分かる.一方、図 3.8(b)と図 3.8(d)を比べると、ロボットの目標速度を 1.0 [m/s] とした場合は、目標軌道へ達するまで約 2 倍の時間を要することが分かる.ロボットの移動軌跡は大きく変化していないため、ロボットの移動速度を必要以上に減速しすぎてしまったことが原因であると考えられる.したがって、目標速度と現在の速度との相対的な関係によって加速度を制御するのではなく、現在の速度そのものも考慮することが必要であると考えられる.

人間が行うカーレースでも減速の技術はラップタイムに影響する重要な要素であり、主にコーナーを走る際に発揮されるものである。今回提案した手法は直線 軌道を追従するものであったが、速度制御は曲線軌道の追従においてさらに効果 を発揮するものと思われる。

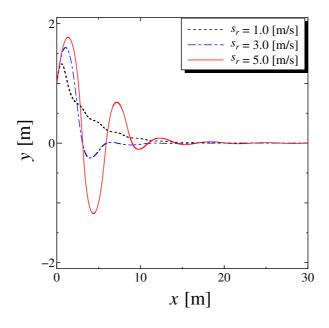


(a) Movement trajectory of the robot without any speed controller

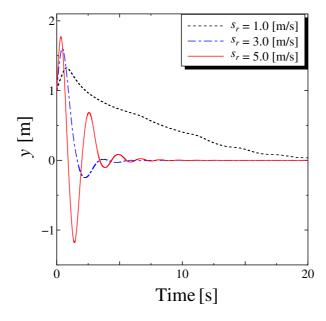


(b) Time response of the y-position of the robot without any speed controller

Fig. 3.8: Simulation results for the controller based on human driving techniques

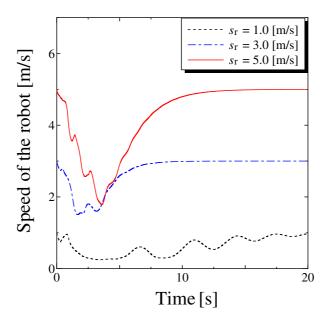


(c) Movement trajectory of the robot with a speed controller



(d) Time response of the y-position of the robot with a speed controller

Fig. 3.8: Simulation results for the controller based on human driving techniques



(e) Time history of the robot speed with a speed controller

Fig. 3.8: Simulation results for the controller based on human driving techniques

3.4 状態変数の変化量を考慮したファジィ制御器の設計

前節にて設計した制御器は主に α と β を基に $\dot{\phi}$ と \dot{s} を決定するものであった.しかしながら,この制御器は現在位置を基に前進加速度を決定するようなものであり,特殊な制御器であると言える.また,実機実験によってこの制御器の評価を行うためには,所望の前進加速度と操舵角の角速度を実現できる実機を製作しなければならないが,これは容易ではない.そこで,本節では実機実験を行うことを目的として,状態変数を $(\alpha,\beta,\Delta\alpha,\Delta\beta)$ とし,制御入力を (s,ϕ) と置き直してファジィ制御器を設計する.なお,速度制御の有用性については前節にて示したので,ここでは速度 s は一定として,操舵角 ϕ のみによって移動ロボットを制御する.

3.4.1 ファジィ制御器の設計

設計する制御器は画像座標上の状態変数 $(\alpha, \beta, \Delta\alpha, \Delta\beta)$ より操舵角 ϕ を決定しなければならない.ファジィ制御器は主に以下の 3 つのステップを通して出力を決定する.

- 1. 状態変数のファジィ化
- 2. 制御規則の適合度の算出
- 3. 出力の非ファジィ化

以下にそれぞれのステップの詳細について述べる.

状態変数のファジィ化

実数を適合度を有する言語表現(ファジィ集合)に変換することをファジィ化と言い、その適合度はメンバシップ関数によって求められる。それぞれの状態変数に対するメンバシップ関数を図 3.9 に示す、状態変数の値はこれらのメンバシップ関数を通して適合度を有する言語表現(N: Negative, ZO: Zero, P: Positive)にファジィ化される。これらのメンバシップ関数はいくつかの予備実験によって試行錯誤的に求めたものであり、カメラやロボットの性能に依存するものである。

制御規則の適合度の算出

表 3.4 はファジィ化された表現によって大まかに場合分けされたロボットと直線 との関係をもとに制定した ϕ のための制御規則である. 例として表中の 1 番のルールは次のように解釈するものとする.

if
$$\alpha = N, \beta = N$$
 then $\phi = N$ (3.8)

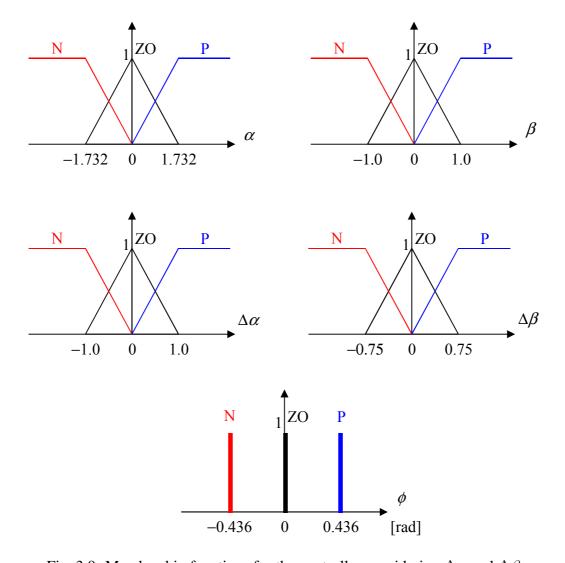


Fig. 3.9: Membership functions for the controller considering $\Delta \alpha$ and $\Delta \beta$

各ルールの適合度には,そのルールに関係する状態変数の適合度の最小値が採用される.各ルールの適合度はその出力の適合度として採用されるが,同一の出力が設定されているルールが他にも存在する場合は,そのうち最大の適合度が採用される (min-max 法 (34)).

出力の非ファジィ化

出力がファジィ集合の状態では操作量としてロボットへ与えることはできないため,非ファジィ化を行い数値にしなければならない.非ファジィ化はシングルトンを用いた簡略化推論法を用いることとし,以下のように適合度による重み付き平均を求めることで実数化することができる.

Table 3.4: Fuzzy rules f	for ϕ in the controller ϕ	considering /	$\Delta \alpha$ and $\Delta \beta$
10010 01.11 00225 100100 1	. σ τ φ τιτι στισ σ σ τισι σ τισι		

No.	α	β	$\Delta \alpha$	$\Delta \beta$	φ
1	N	N	_	_	N
2	N	ZO	_	_	N
3	N	P	_	_	ZO
4	ZO	N	_	_	N
5	ZO	ZO	N	N	N
6	ZO	ZO	N	ZO	N
7	ZO	ZO	N	P	ZO
8	ZO	ZO	ZO	N	N
9	ZO	ZO	ZO	ZO	ZO
10	ZO	ZO	ZO	P	P
11	ZO	ZO	P	N	ZO
12	ZO	ZO	P	ZO	P
13	ZO	ZO	P	P	P
14	ZO	P	_	_	P
15	P	N	_	_	ZO
16	P	ZO	_	_	P
17	P	P	_	_	P

$$\phi = \frac{h_{\phi,N} \times (-0.436) + h_{\phi,ZO} \times 0 + h_{\phi,P} \times 0.436}{h_{\phi,N} + h_{\phi,ZO} + h_{\phi,P}}$$
(3.9)

ただし $h_{\phi,\mathrm{N}}$ は $\phi=\mathrm{N}$ の適合度を表すものとする.

3.4.2 シミュレーションと実機実験

状態変数の変化量を考慮したファジィ制御器の性能を評価するためにシミュレーションと実機実験の両方を行った.

シミュレーションによる評価

制御対象は図 3.2 に示す操舵型の 4 輪移動ロボットとした.このロボットの動作モデルには式 (3.7) を用いることとした.ロボットの操舵角の可動範囲は ± 0.436 [rad] , その角速度の最大値は ± 1.75 [rad/s] , 前輪と後輪の車軸間距離 L は 0.26 [m] とする.カメラは高さ 0.09 [m] の位置に鉛直下向きからチルト方向に 70 [deg] 傾けた向きに

取り付けられており,その焦点距離は $0.02~[\mathrm{m}]$,画角は $2.09~[\mathrm{rad}]$,フレームレートは $30~[\mathrm{fps}]$ とする.画像上の直線の傾きは $-\infty < \alpha < \infty$ で表わされ,切片の範囲は $-\infty < \beta < \infty$ となる.ロボットの初期位置は $(x,y,\theta)=(0.0~[\mathrm{m}],-0.4~[\mathrm{m}],\pi/4~[\mathrm{rad}])$ とし,前進速度は $0.5~[\mathrm{m/s}]$ とする.前進速度は $0.5~[\mathrm{m/s}]$ とする追従目標である直線はy=0とし,y=0の状態を維持することで直線追従を達成したものとみなすこととした.シミュレーションの時間間隔は $0.01~[\mathrm{s}]$ とした.

ロボットの移動軌跡を図 3.10 に示す.最初に目標軌道にさしかかった時にオーバーシュートが発生しているが,振動することなく,追従目標である y=0 に沿って移動していることが分かる.

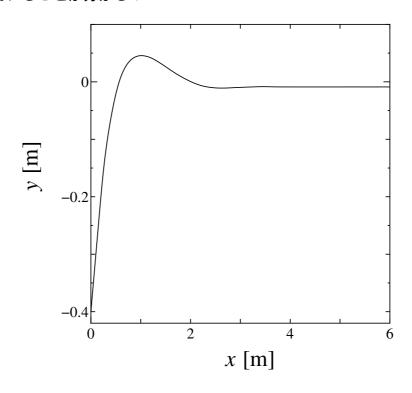


Fig. 3.10: Simulation result for the controller considering $\Delta \alpha$ and $\Delta \beta$

実機実験による評価

使用する移動ロボットを図 3.11 に示す.この移動ロボットは前輪操舵後輪駆動型のロボットであり,市販のラジコンカーをマイクロコントローラによって制御できるように改造したものである.最前部に取り付けられた USB カメラは地面を注視し直線を撮影するものである.ノートパソコンは画像処理を行うもので,図 3.12 のように得られた画像からハフ変換を用いて直線の傾き α と切片 β を抽出する. $\dot{\alpha}$ と $\dot{\beta}$ はそれぞれ 1 つ前の画像より得られた値との差分を用いることとする.ロボッ

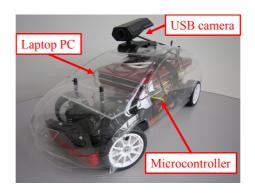


Fig. 3.11: Experimental robot

Table 3.5: Specifications of the experimental robot

Item	Property
CPU	1.33 [GHz]
Memory	2 [MB]
OS	Xubuntu 11.04
Image size	320 × 240 [pixel]
Sampling rate	10 [fps]
Max. advanced speed	0.5 [m/s]
Range of steering angle	± 0.436 [rad]

トの初期位置はシミュレーションと同様に (x, y, θ) = $(0.0 \text{ [m]}, -0.4 \text{ [m]}, \pi/4 \text{ [rad]})$ とし,前進速度も約 0.5 [m/s] とした.

実験結果を図 3.13 と図 3.14 に示す.図 3.13 は軌道追従の様子を表したものである.この写真はビデオカメラで撮影した動画を基に,背景差分を用いて 0.5 [s] 間隔でロボットの部分を抜き出して表示したものである.図 3.14 に示す記録されたロボットの位置より,オーバーシュートも振動もなく,追従目標である y=0 の直線に沿って進んでいるが,y 軸方向に -0.06 [m] 程度の誤差が残っていることが分かる.

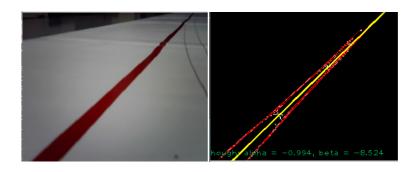


Fig. 3.12: Image processing



Fig. 3.13: Appearance of the robot (0.5 [s] interval)

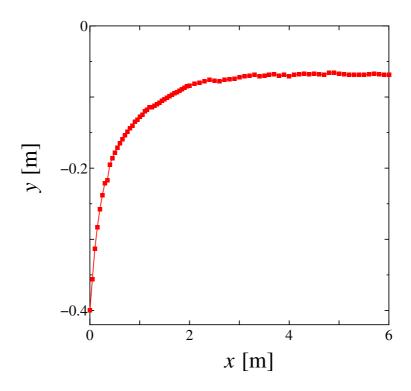


Fig. 3.14: Experimental result for the real robot

考察

実験結果より、設計したファジィ制御器によって移動ロボットの軌道追従を行うことが可能であることを示した.この誤差の原因の一つとして考えられることは、今回の実験に使用した実機がそれぞれの車輪の径の違いや重心の偏りのために直進時でも右側に曲がってしまうという特性を持っているということが挙げられる.また、シミュレーション実験の結果である図 3.10 をよく見ると、この y 方向の誤差はわずかながら残っていることが分かる.したがって、実機の特性だけではなく理論的な原因があると考えられる.図 3.15 は追従目標である直線に沿った向きで y 方向に少し離れた位置にロボットが存在している状態を示している.このとき、画像座標上で追従目標の直線は $\alpha=N$ 、 $\beta=P$ であるため、ロボットは図 3.16 の状態だと信じて直進してしまう.対策として考えられるのは、図 3.15 のような場合と図 3.16 のような場合を区別するためにメンバシップ関数や制御規則を細分化することだが、それに伴いファジィ制御器のチューニングが煩雑になることが懸念される.したがって,GA やニューラルネットワークを組み合わせてチューニングの自動化および最適化を行うことが必要であると思われる.

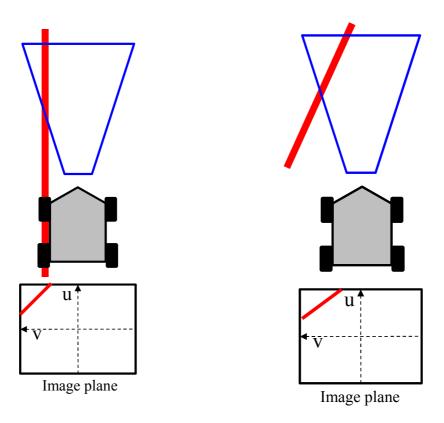


Fig. 3.15: Robot advancing along the target line

Fig. 3.16: Robot advancing toward the target line

3.5 まとめ

本章では、ファジィ制御器を用いた画像に基づく視覚サーボの手法を提案した、 この手法はモデルベースの手法と比べて、相互行列を用いることなく制御器を設 計することが可能であった、したがって、カメラの向きに関する厳密な条件は必 要なく、ロボットの前方を向いているという大まかな仮定で十分であった、また、 ロボットの速度制御を行うファジィ制御器と画像に映った目標軌道のパラメータ の値とその変化量を状態変数とするファジィ制御器を設計した、速度制御を行う ファジィ制御器はシミュレーションによって評価し,その結果より,速度制御を行 うことでオーバーシュートが軽減できることを示した.一方で,元々目標速度が 1.0 [m/s] のように遅い場合は過度な減速によって初期位置から目標軌道に到達す るまでの時間が長くなってしまった.したがって,これを改善するためには状態 変数の選び方や制御規則の変更が必要であると思われる、さらに、画像に映った 目標軌道のパラメータ値とその変化量を状態変数とするファジィ制御器はシミュ レーションと実機実験の両方によって評価し,ロボットの位置の y 座標値にわずか な残留誤差が残ったものの、目標軌道に沿って直進することが可能であることを 示した.この残留誤差を減少させるためには,状態変数のラベルを細分化し,よ りきめ細かい制御規則を作成することが考えられるが、それに伴って煩雑化する 制御器の調整を自動化する手法を合わせて考える必要がある、

第4章 任意軌道追従のためのファ ジィ制御器のGAによる設計

本章では前述の画像に基づくファジィ制御器を拡張して任意軌道追従を行う手法 を提案する、任意軌道追従を行うためのファジィ制御器は直線追従のものと比べて 経験に基づく手作業では調整することが困難である. そこで, ファジィ制御器のメ ンバシップ関数と制御規則を GA を用いて設計を行うこととする . GA を用いた移 動口ボットのファジィ制御器の設計に関する研究として, Homaifar & McCormick (65) や Wong ら (66) の論文が挙げられる.これらは,位置に基づいたファジィ制御器で あり,シミュレーションによる評価に留まっている.本論文は画像に基づくファ ジィ制御器を扱っており, GA によって設計したファジィ制御器を実機に搭載した 際の性能評価を目的としたものである.GA を用いて設計されたファジィ制御器は 評価用の目標軌道に特化し,ロバスト性に欠ける傾向があるので実機へ搭載する ためには工夫が必要である、本論文では GA による設計時のシミュレーションに ノイズを付加することで,設計される制御器のロバスト性を向上させる手法を提 案する.提案手法はシミュレーションと実機実験によって評価し,適切な大きさ のノイズを付加することで制御器のロバスト性が向上し, GA によって設計された 制御器を実機に搭載し目標軌道を追従することが可能となることを示す、以下に、 問題設定,任意軌道追従のための画像に基づくファジィ制御器,GAを用いたファ ジィ制御器の設計,評価実験について述べる.

4.1 問題設定

制御対象は図 4.1 のようにカメラを搭載したロボットであり,取得したカメラ画像を基にロボットが地面に描かれた任意曲線を追従することを制御の目標とする.ロボットは二次元平面上を移動するものとし,世界座標は x-y 座標で表され,ロボットの状態はその座標上の位置と方位角 (x,y,θ) で表わすものとする.カメラ画像の座標は図 4.1 のように u-v 座標で表すものとする.画像座標に射影された目標軌道は最小二乗法 $^{(67)}$ によって直線 $v=\alpha u+\beta$ へと近似され,この直線のパラ

メータである α と β は以下の式により求められる .

$$\alpha = \frac{n \sum_{k=1}^{n} u_k v_k - \sum_{k=1}^{n} u_k \sum_{k=1}^{n} v_k}{n \sum_{k=1}^{n} u_k^2 - \left(\sum_{k=1}^{n} u_k\right)^2}, \qquad \beta = \frac{\sum_{k=1}^{n} u_k^2 \sum_{k=1}^{n} v_k - \sum_{k=1}^{n} u_k v_k \sum_{k=1}^{n} u_k}{n \sum_{k=1}^{n} u_k^2 - \left(\sum_{k=1}^{n} u_k\right)^2}$$
(4.1)

ただし, (u_k,v_k) は画像上の目標軌道を含んでいる k 番目の画素の座標であり,n はその総数を表している.また,この直線のパラメータ α , β の変化量 $\Delta\alpha$, $\Delta\beta$ は,現在のフレームより抽出したパラメータ値 α_t , β_t と 1 つ前のフレームより抽出した値 α_{t-1} , β_{t-1} との差分値とする.なお, β および $\Delta\beta$ に関するメンバシップ関数については,それぞれ画像の横幅 W によって正規化した値 $2\beta/W$ および $2\Delta\beta/W$ を用いることとする.以降,これらのパラメータ α , β , $\Delta\alpha$, $\Delta\beta$ を状態変数として扱うこととする.

制御対象は前章と同様に図 3.2 に示すように前輪操舵型の 4 輪移動ロボットとし,制御入力は前進速度 s と前輪の操舵角 ϕ とする.軌道追従を行うための制御器は画像座標上の状態 α , β , $\Delta\alpha$, $\Delta\beta$ から制御入力 s, ϕ を決定しなければならない.画像に基づく任意軌道追従制御のブロック線図を図 4.2 に示す.このブロック線図より,制御器はロボットの自己位置を参照することなく,画像座標上の情報だけを基に制御入力を決定していることがわかる.

 (x, y, θ) : Position of the robot

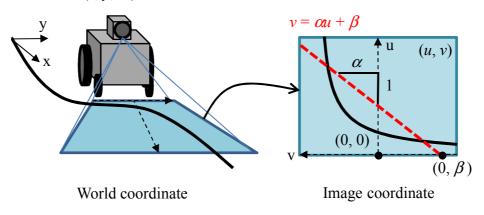


Fig. 4.1: Definition of coordinates

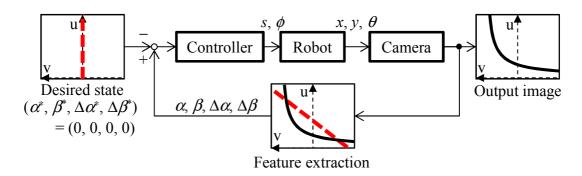


Fig. 4.2: Block diagram of image-based tracking control for an arbitrary trajectory

4.2 任意軌道追従のためのファジィ制御器

状態変数 α , β , $\Delta \alpha$, $\Delta \beta$ は以下に示すようにそれぞれ 3 つのラベルからなるファジィ集合 $(\tilde{\alpha}, \tilde{\beta}, \Delta \tilde{\alpha}, \Delta \tilde{\beta})$ へとファジィ化されることとする .

$$\tilde{\alpha} = \{N, ZO, P\}, \quad \tilde{\beta} = \{N, ZO, P\}, \quad \Delta \tilde{\alpha} = \{N, ZO, P\} \quad \Delta \tilde{\beta} = \{N, ZO, P\}$$
 (4.2)

ただし,N,ZO,P はそれぞれ Negative,Zero,Positive を意味するラベルである.ラベルの数は扱う問題によって調整が必要であるが,この数は予備実験によって決定したものである.設計するメンバシップ関数を図 4.3 に示す.メンバシップ関数はそれぞれのラベルへの適合度を求めるためのものであり, $\tilde{\alpha} = N$ 」の適合度を求めるメンバシップ関数を $\mu_{\tilde{\alpha},N}(\alpha)$ と表すこととすると,図 4.3 の α に関するメンバシップ関数のグラフは $\mu_{\tilde{\alpha},N}(\alpha)$, $\mu_{\tilde{\alpha},ZO}(\alpha)$, $\mu_{\tilde{\alpha},P}(\alpha)$ の 3 つの関数を描いたものである. α 以外のメンバシップ関数についても同様であり,それぞれのメンバシップ関数の値域は [0,1] である.

制御規則は 4 つの状態変数がそれぞれ 3 つのラベルを有することから 3^4 通り考えられ,これらは以下のように IF-THEN ルールとして定められる.

$$R_{0}: \text{IF} \quad \tilde{\alpha} = \text{N}, \ \tilde{\beta} = \text{N}, \ \Delta \tilde{\alpha} = \text{N}, \ \Delta \tilde{\beta} = \text{N} \quad \text{THEN} \quad \tilde{s} = \tilde{s}_{0}, \ \tilde{\phi} = \tilde{\phi}_{0}$$

$$R_{1}: \text{IF} \quad \tilde{\alpha} = \text{N}, \ \tilde{\beta} = \text{N}, \ \Delta \tilde{\alpha} = \text{N}, \ \Delta \tilde{\beta} = \text{ZO} \quad \text{THEN} \quad \tilde{s} = \tilde{s}_{1}, \ \tilde{\phi} = \tilde{\phi}_{1}$$

$$\vdots$$

$$R_{79}: \text{IF} \quad \tilde{\alpha} = \text{P}, \ \tilde{\beta} = \text{P}, \ \Delta \tilde{\alpha} = \text{P}, \ \Delta \tilde{\beta} = \text{ZO} \quad \text{THEN} \quad \tilde{s} = \tilde{s}_{79}, \ \tilde{\phi} = \tilde{\phi}_{79}$$

$$R_{80}: \text{IF} \quad \tilde{\alpha} = \text{P}, \ \tilde{\beta} = \text{P}, \ \Delta \tilde{\alpha} = \text{P}, \ \Delta \tilde{\beta} = \text{P} \quad \text{THEN} \quad \tilde{s} = \tilde{s}_{80}, \ \tilde{\phi} = \tilde{\phi}_{80}$$

ただし,各規則の後件部の制御入力 \tilde{s}_i および $\tilde{\phi}_i$ $(i=0,\dots,80)$ には,以下のように定義されるファジィ集合 \tilde{s} , $\tilde{\phi}$ が有するラベルが入るものとする.

$$\tilde{s} = \{ \text{ZO}, \text{PS}, \text{PB} \}, \quad \tilde{\phi} = \{ \text{NB}, \text{NS}, \text{ZO}, \text{PS}, \text{PB} \}$$
 (4.3)

ここで , \tilde{s} に N のラベルが含まれていない理由は s>0 としているためである . 制御規則の適合度 $h_{\mathrm{R}i}$ ($i=0,\ldots,80$) は前件部のラベルに対応するメンバシップ関数の値の代数積に基づいて以下のように求める .

$$h_{R0} = \mu_{\tilde{\alpha},N}(\alpha) \cdot \mu_{\tilde{\beta},N}(\beta) \cdot \mu_{\Delta\tilde{\alpha},N}(\Delta\alpha) \cdot \mu_{\Delta\tilde{\beta},N}(\Delta\beta)$$

$$h_{R1} = \mu_{\tilde{\alpha},N}(\alpha) \cdot \mu_{\tilde{\beta},N}(\beta) \cdot \mu_{\Delta\tilde{\alpha},N}(\Delta\alpha) \cdot \mu_{\Delta\tilde{\beta},ZO}(\Delta\beta)$$

$$\vdots$$

$$h_{R79} = \mu_{\tilde{\alpha},P}(\alpha) \cdot \mu_{\tilde{\beta},P}(\beta) \cdot \mu_{\Delta\tilde{\alpha},P}(\Delta\alpha) \cdot \mu_{\Delta\tilde{\beta},ZO}(\Delta\beta)$$

$$h_{R80} = \mu_{\tilde{\alpha},P}(\alpha) \cdot \mu_{\tilde{\beta},P}(\beta) \cdot \mu_{\Delta\tilde{\alpha},P}(\Delta\alpha) \cdot \mu_{\Delta\tilde{\beta},P}(\Delta\beta)$$

$$(4.4)$$

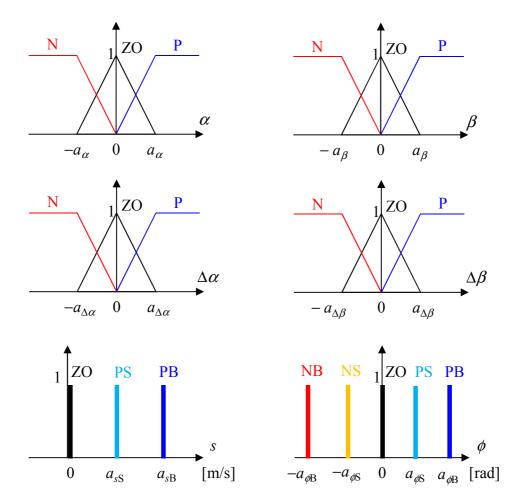


Fig. 4.3: Membership functions for arbitrary trajectory tracking

求めた制御規則の適合度を基にした加重平均によって制御入力 $s,\,\phi$ は以下のように決定する .

$$s = \frac{\sum_{i=0}^{80} h_{Ri} \hat{s}_i}{\sum_{j=0}^{80} h_{Rj}}, \quad \phi = \frac{\sum_{i=0}^{80} h_{Ri} \hat{\phi}_i}{\sum_{j=0}^{80} h_{Rj}}$$

$$(4.5)$$

ただし, \hat{s}_i および $\hat{\phi}_i$ は,それぞれi 番目の制御規則の後件部 \tilde{s}_i および $\tilde{\phi}_i$ のラベルに対応する実数値である.

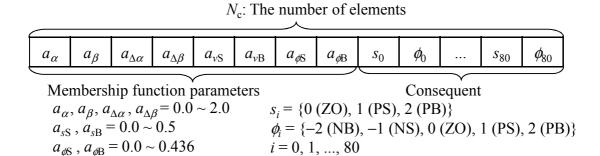


Fig. 4.4: Chromosomeencoding membership function parameters and consequent constants

4.3 GAを用いたファジィ制御器の設計

提案したファジィ制御器の性能はメンバシップ関数の形状や制御規則によって決定される.しかしながら,それらは互いに影響しており,81 通りの規則を手作業で調整することは困難である.そこで,実数値 GA を用いてそれらの値を探索しファジィ制御器の設計を行う.調整が必要な値はメンバシップ関数の形状を表すパラメータ a_{α} , a_{β} , $a_{\Delta\alpha}$, $a_{\alpha\beta}$, $a_{v\rm S}$, $a_{v\rm B}$, $a_{\phi\rm S}$, $a_{\phi\rm B}$ および制御規則の後件部の定数値 \tilde{s}_0 , $\tilde{\phi}_0$, ..., \tilde{s}_{80} , $\tilde{\phi}_{80}$ である.これらをコード化した染色体を図 4.4 に示す.メンバシップ関数の形状を表す変数には,それぞれ図に示す範囲の実数値が設定され,制御規則の後件部にはそれぞれラベルと対応付けされた整数値が設定される.

個体の評価を行うための目標軌道 $y_{r1}=f_1(x)$ は次式で表されるように直線と正弦波からなるものとする.

$$y_{\rm r1} = \begin{cases} 0 & \text{if } 0 \le x < 4\\ 0.5 \sin\left(2\pi \cdot \frac{x}{4}\right) & \text{if } 4 \le x \le 12 \end{cases}$$
 (4.6)

この正弦波部分の振幅や波長は使用する実機の旋回半径を基に設定した値であり, この目標軌道を追従することができれば旋回半径以内の曲線は追従出来るものと 思われる.

4.4 実験1

シミュレーションによって設計したファジィ制御器をそのまま実機に適用した場合,モデル化誤差の影響により所望の動作を実現できない場合が多い.そこで,予めシミュレーションに動作ノイズと計測ノイズを付加することでモデル化誤差の影響を吸収することを試みる.この実験では付加するノイズの種類や大きさと実機の動作の関係について考察する.

4.4.1 GA によるファジィ制御器の設計

シミュレーションに使用するロボットのモデルは実機を基にした前輪操舵型の 4 輪移動ロボットとし,カメラは後輪から車体前方に向かって 0.24~[m],地面から 0.13~[m] の位置に鉛直下向きより前方へ $60~[\deg]$ 傾けた状態で搭載されているもの とした.また,ロボットの状態は世界座標上で (x,y,θ) と表し,動作モデルには次式のように動作ノイズ $\delta_{\rm m}$ を与えた.

$$\dot{x} = s \cos \theta \cdot (1 + \delta_{\rm m})
\dot{y} = s \sin \theta \cdot (1 + \delta_{\rm m})
\dot{\theta} = \frac{s \tan \phi}{L} \cdot (1 + \delta_{\rm m})$$
(4.7)

ただし, δ_{m} は $\mathcal{N}(0,\sigma_{\mathrm{m}}^2)$ の正規分布に従うものとした.

カメラは実機を参考にしてフレームレートは $20~{\rm [fps]}$ とし,水平画角は $52~{\rm [deg]}$,垂直画角は $40~{\rm [deg]}$,画像のサイズは $320\times 240~{\rm [pixel]}$ としたピンホールモデルを用いることとした.ロボットは取得したカメラ画像から求めた真の計測値に計測ノイズ δ_o を次式のように与えた α_a , β_a を基に制御入力を決定する.

$$\alpha_a = \alpha + \delta_0, \quad \beta_a = \beta + \delta_0$$
 (4.8)

ただし, δ_o は $\mathcal{N}(0,\sigma_o^2)$ の正規分布に従うものとした.なお,評価関数の計算に用いる状態変数の RMS 誤差は α 、 β より求めることとした.

付加するノイズの大きさとロボットの動作の関係を調べるために,ノイズを与えない場合,動作ノイズ $\sigma_{\rm m}$ のみを 0.2,0.4 の 2 通り付加した場合,観測ノイズ $\sigma_{\rm o}$ のみを 0.2,0.4 の 2 通り付加した場合の計 5 通りについて GA によるファジィ制御器の設計を行った.いずれの場合も初期個体は同じものを用いて,ロボットの初期状態は (x,y,θ) = (0 [m], 0 [m], 0 [rad]),1 世代の試行の制限時間は 40 [s] とし,200 世代で打ち切りとした.

個体の評価値 V は目標軌道を追従した際の状態変数 α , β それぞれの RMS 誤差 $\alpha_{\rm e}$, $\beta_{\rm e}$ と制限時間内に x 方向に進んだ距離 $x_{\rm f}$ および軌道の終端へと到達した時刻 $t_{\rm f}$ に基づく以下の評価関数によって求める .

$$V_{1} = k_{11} \left(\frac{1}{\alpha_{e} + \varepsilon} + \frac{1}{\beta_{e} + \varepsilon} \right) + k_{12} x_{f} + \frac{k_{13}}{t_{f}}$$
 (4.9)

ただし, k_{11} , k_{12} , k_{13} は重み係数であり, ε はゼロ除算を防ぐための値である.この評価関数の値が最大となるようなファジィ制御器を GA によって設計する.親の選択はルーレット選択によって行い,交叉は交叉率 0.9 の一様交叉とした.突然変異は染色体のある要素の値を,その要素がとり得る範囲内で生成したランダムな値に置き換えるものとし,その発生確率は要素数 N_c の逆数とした.1 世代の個体数は 100,エリートは 10 体保存するものとした.式 (4.9) の評価関数の重み係数はそれぞれ, $k_{11}=0.5$, $k_{12}=10.0$, $k_{13}=100.0$, $\varepsilon=0.025$ とした.

図 4.5 に,それぞれのノイズを付加した場合の世代数と最良個体の評価値との関係を示す.ノイズを付加した場合は付加しない場合と比べて評価値が小さいが,いずれの場合も目標軌道の終点まで到達出来ていた. $\sigma_o = 0.4$ の場合に,初期個体から高い評価値が得られているのは偶然付加されたノイズが軌道追従に良い影響を与えたためであると思われる.

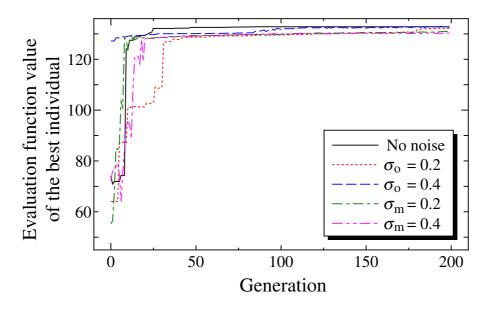


Fig. 4.5: History of the evaluation function values

4.4.2 シミュレーション上のロボットの動作

図 4.6,図 4.7 はそれぞれ設計されたファジィ制御器によって目標軌道を追従した際の状態変数の値の変化を示したものである.ここでは,ロボットは真の計測値 α , β を基に軌道追従を行っている.図 4.6 は動作ノイズを与えて設計した制御器を用いた場合であり,図 4.7 は計測ノイズを与えて設計した制御器を用いた場合である.動作ノイズの大きさを変えた場合,終端への到達時刻は早くなるが状態変数の値については大きな変化は見られない.一方で,計測ノイズを与えて設計した場合は σ 。の値が大きすぎると,状態変数の値が激しく振動している事がわかる.これは,ロボットが左右に振動していることを意味する.

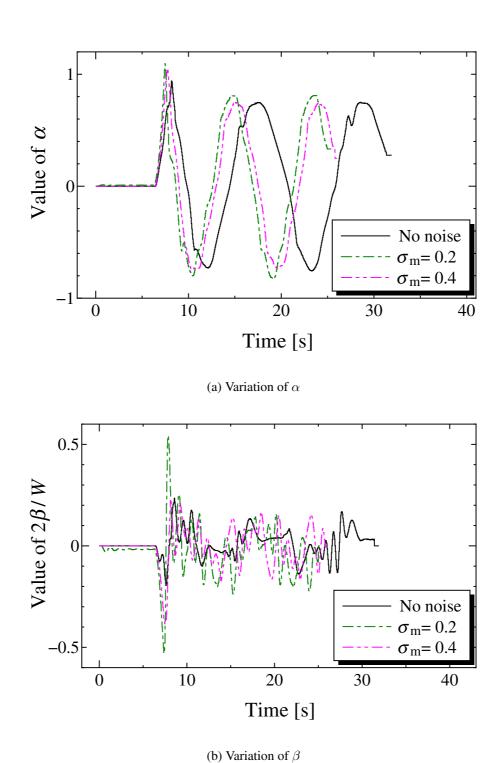


Fig. 4.6: Simulation result of using the designed fuzzy controllers with additional motion noise $\sigma_{\rm m}$

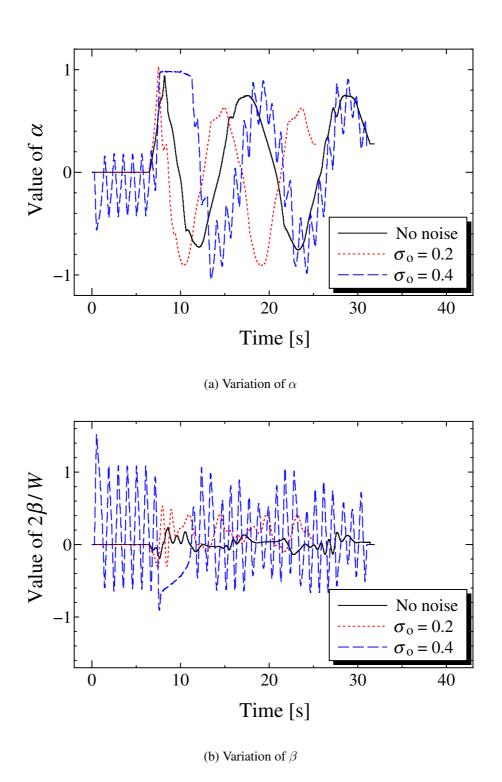


Fig. 4.7: Simulation result of using the designed fuzzy controllers with additional observation noise $\sigma_{\rm o}$

4.4.3 ノイズの種類とロボットの動作

シミュレーションによって設計された 5 通りのファジィ制御器を実機に搭載した際のロボットの動作を評価する.使用するロボットは図 3.11 に示すように市販のラジコンカーを改造したものであり, USB カメラ, 画像処理を行うための小型PC,機体の制御を行うためのマイクロコントローラを搭載している.シミュレーションと同じ目標軌道を地面に描き,設計されたファジィ制御器によりこれを追従し,計測した状態変数の値によりロボットの動作を評価する.

図 4.8 および図 4.9 はそれぞれ動作ノイズを与えて設計された制御器および計測ノイズを与えて設計された制御器による軌道追従の結果を示している.ノイズを与えずに設計した制御器は,15 [s] 付近で目標軌道が直線から正弦波形に切り替わる位置で停止してしまったため,それ以降の α , β の値が一定値となっている.動作ノイズを与えて設計した場合について,シミュレーションでは α の値に振動は見られなかったが,実機実験では α の値が振動し β の振動も大きくなっていることが分かる. $\sigma_{\rm m}=0.4$ の場合は目標軌道の終端まで追従することができたが, β の値が ± 1 に近い値まで振動しているため,目標軌道を見失う危険性がある.一方,計測ノイズを与えて設計した場合について, $\sigma_{\rm o}=0.4$ の場合はシミュレーションと同様に初期状態から大きく振動しており,18 [s] 付近で目標軌道を見失ってしまった. $\sigma_{\rm o}=0.2$ の場合については,目標軌道の終端まで追従することができたが,この場合も β の値が振動してしまっていることから目標軌道を見失う危険性がある.

以上の結果より,シミュレーションに動作ノイズと計測ノイズを付加することでモデル化誤差を吸収し,GAにより設計したファジィ制御器を実機に搭載しても目標軌道を追従することが可能となることを示した.しかし,シミュレーションにはない小刻みな振動が確認された.これについては,5 通りの制御器によるシミュレーション結果で状態変数の値の変化に大きな違いがあるにもかかわらず,図 4.5 に示すように評価値の差が小さいことから,評価関数の重み係数 k_{11} の値が小さく状態変数の RMS 誤差を減少することによる評価値の上昇が小さすぎるためであると考えられる.しかし,この値を大きくすると設計時に初期位置から動かない個体の評価値が高くなるため,初期位置をランダムに設定するなどの工夫が必要である.

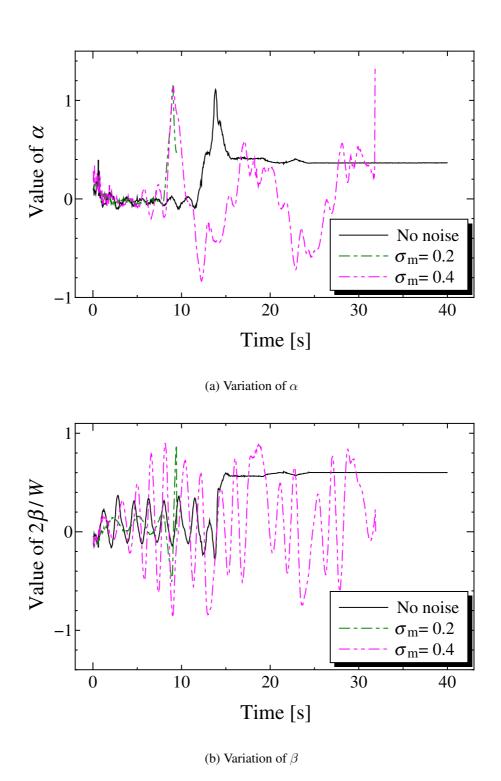


Fig. 4.8: Experimental result of using the designed fuzzy controllers with additional motion noise $\sigma_{\rm m}$

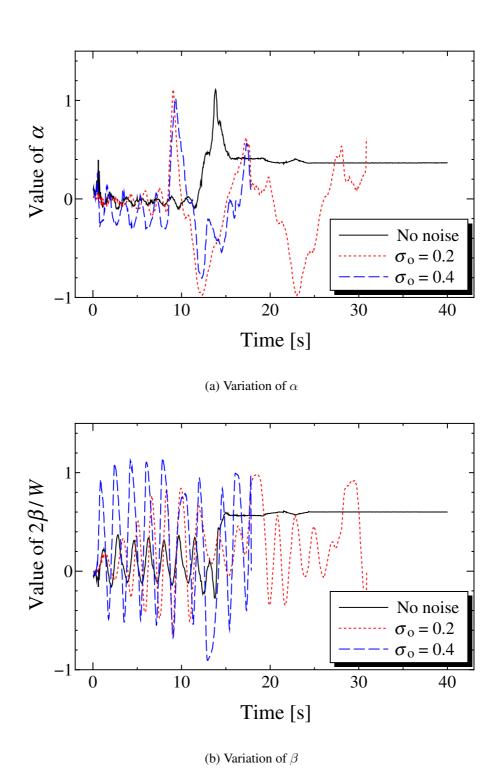


Fig. 4.9: Experimental result of using the designed fuzzy controllers with additional observation noise σ_o

4.5 実験2

この実験では,評価関数と動作モデルへのノイズの与え方を変更した場合のロボットの挙動について調べる.制御対象は前章と同様に図 3.11 に示す前輪操舵型の 4 輪移動ロボットであり,シミュレーションには式 (4.11) に示す動作モデルを用いた.カメラは後輪から車体前方に向かって 0.24 [m],地面から 0.13 [m] の位置に鉛直下向きより前方へ 60 [deg] 傾けた状態で搭載されているものとした.カメラは実機の仕様を基にフレームレートは 10 [fps] とし,水平画角は 52 [deg],垂直画角は 40 [deg],画像のサイズは 320×240 [pixel] としたピンホールモデルを用いることとした.また,制御入力の最大値はそれぞれ $s_{\max}=0.5$ [m/s], $\phi_{\max}=0.436$ [rad] とした.

個体の評価値 V は,目標軌道を追従した際の状態変数 β の RMS 誤差 $\beta_{\rm e}$,制限時間内に ${\bf x}$ 方向に進んだ距離 $x_{\rm f}$ および軌道の終端へ到達した時刻 $t_{\rm f}$ に基づく以下の評価関数によって求める.

$$V_2 = \frac{k_{21}}{\beta_e + \varepsilon} + k_{22}x_f + \frac{k_{23}}{t_f}$$
 (4.10)

ただし, k_{21} , k_{22} , k_{23} は重み係数であり, ε はゼロ除算を防ぐための値である. α の RMS 誤差が含まれていないのは目標軌道が直線ではなく, α を常に 0 付近に保つことが困難なためである.ここでは,この評価関数の値が最大となるようなファジィ制御器を GA によって設計する.親の選択はルーレット選択によって行い,交叉は交叉率 0.9 の一様交叉とした.突然変異は染色体のある要素の値を,その要素が取り得る範囲内で生成したランダムな値に置き換えるものとし,その発生確率は要素数 N_c の逆数とした.1 世代の個体数は 100,エリートは 10 体保存するものとした.目標軌道は式(4.6)の軌道とし,ロボットの初期位置は (x,y,θ) = (0 [m], 0 [m], 0 [m], 0 [m], 0 [m], 0 [m], 0 [m], n [m], m], m0 [m], m0 [m], m0 [m], m0 [m], m0 [m0], m0 [m0], m0 [m0], m0], m0 [m0], m0], m0 [m0], m0], m0 [m0], m0], m0],

GA を用いてファジィ制御器を設計する場合,評価用の目標軌道に特化した制御器となることがあるため,ロバスト性を向上させるためにシミュレーションにノイズを付加することとする.ノイズ $\delta_{\rm m}$ は $N(0,\sigma_{\rm m}^2)$ の正規分布に従うものとし,制御入力 s と ϕ に加法的に加わる誤差の形で次式のように与えることとした.

$$\dot{x} = (s + \delta_{\rm m} s_{\rm max}) \cos \theta
\dot{y} = (s + \delta_{\rm m} s_{\rm max}) \sin \theta
\dot{\theta} = (s + \delta_{\rm m} s_{\rm max}) \frac{\tan (\phi + \delta_{\rm m} \phi_{\rm max})}{L}$$
(4.11)

ただし, s_{\max} , ϕ_{\max} はそれぞれ制御入力 s, ϕ の最大値である.この動作モデルへの ノイズの付加には,シミュレーションを通して設計した制御器を実機に搭載する際のモデル化誤差による影響を軽減する目的も含まれている.

4.5.1 重み係数とロボットの挙動

評価関数の重み係数 (k_{21}, k_{22}, k_{23}) の値を Case A (1, 250, 500), Case B (2, 250, 500), Case C (1, 250, 1000) のように設定した場合について,それぞれ設計された制御器 A,制御器 B,制御器 C によるロボットの動作の違いを調べる.ここでは重み係数の影響を調べるために動作ノイズ $\delta_{\rm m}$ は 0 とした.

図 4.10 に設計されたそれぞれの制御器を用いた場合のシミュレーション結果を 示す.図 4.10(a) は目標軌道とロボットの軌道を示しており,制御器 A および制御 器 C は目標軌道の終端に達しているが、制御器 B は直線部分で停止していること が分かる、また、正弦波形部分のロボットの軌道は目標軌道の内側を通っている が、これはカメラが前方を向いており、目標軌道の真上を通ると目標軌道を見失 うためである.目標軌道の真上を通るためにはカメラを真下に向けることが必要 であるが、視野が狭くなるため目標軌道を見失わないようにするために移動速度 をさらに遅く設定する等の対処が必要である.図 4.10(b) は時刻に対するロボット の位置の x 座標値を示したものであり,制御器 C が最も早く目標軌道の終端に達 していることが分かる.これは, Case C にて評価関数の重み係数 k_3 を大きく設定 しているため,終端に達する時刻t_fをより小さくする個体がより高い評価値を得 たためだと思われる.これに対して Case B では k_1 を大きく設定しているため誤差 $eta_e = 0$ を維持することができる直線上を低速で前進する制御器が最良個体として 残ってしまったものと思われる. 図 4.10(c) および図 4.10(d) はそれぞれ状態変数 α および β の時間変化を示したものである.制御器 C の場合は,直線部分のみを 前進しているため α も β も共に 0 を維持している . 制御器 A は制御器 C と比べて α も β も共に振幅が小さく,特に β の方は ± 0.1 未満に収まっている.したがって 制御器 A を用いた場合はカメラ画像上で目標軌道の下端は常に中央に捉えられて いることがわかる.

上記のシミュレーション結果より,GA を用いたファジィ制御器の設計において評価関数の重み係数を調整することでロボットの挙動に変化が生じることが分かった.ただし, β の誤差に関する重み係数 k_1 の値を大きく設定しすぎると直線部分に停留するような制御器が設計されてしまうため,直線部分を超えて軌道を追従する個体の方が高い評価値を得られるように, $k_{21}/\varepsilon \leq 4k_{22}$ となるように設定する必要がある.この関係式は式 (4.10) の評価関数において,第 1 項で $\beta_e=0$ としたものと,第 2 項に直線部分の長さ $x_f=4$ [m] を代入したものを比較することにより得られる.

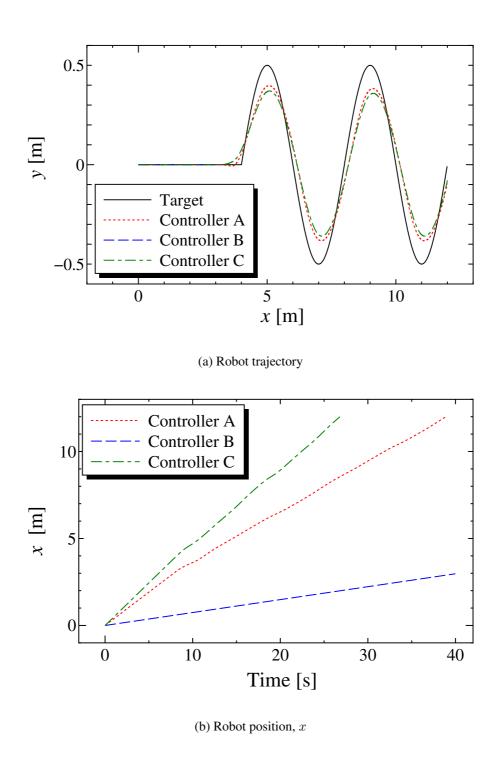


Fig. 4.10: Simulation results when using the fuzzy controllers A, B and C

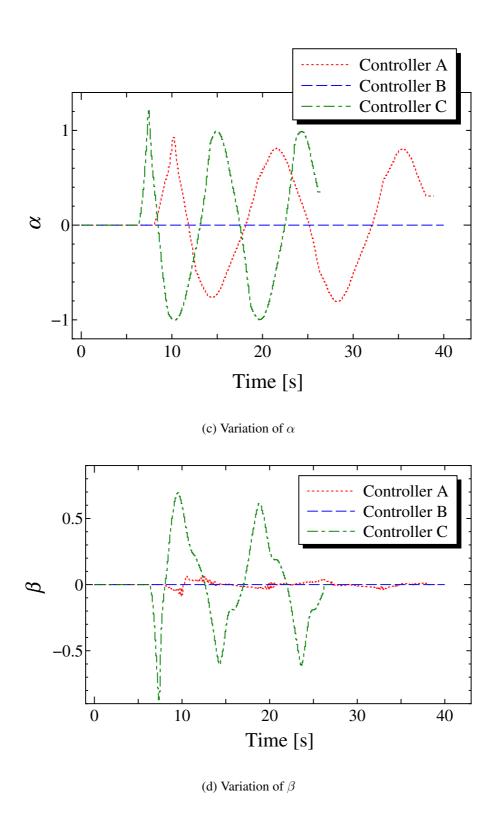


Fig. 4.10: Simulation results when using the fuzzy controllers A, B and C

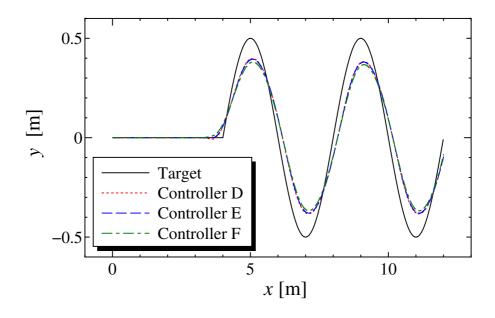
4.5.2 付加するノイズの大きさとロボットの挙動

シミュレーションに動作ノイズを付加して設計した制御器を実機に搭載した際の制御性能を評価する.付加する動作ノイズ $\delta_{\rm m}$ の標準偏差 $\sigma_{\rm m}$ を 0.0, 0.2, 0.4 と設定した場合をそれぞれ Case D, Case E, Case F とし,設計された制御器をそれぞれ制御器 D, 制御器 E, 制御器 F とした.評価関数の重み係数は実験 1 の結果を基に,小さい誤差で目標軌道の終端まで達することのできた (k_{21},k_{22},k_{23}) = (1,250,500) の組み合わせを用いることとした.使用する実機は図 3.11 に示すように市販のラジコンカーを改造したものであり,USB カメラ,画像処理を行うための小型 PC,機体の制御を行うためのマイクロコントローラを搭載している.ロボットの仕様は表 3.5 のとおりである.ロボットの挙動をシミュレーションと実機で比較するために,式 (4.6) の軌道と同じものを地面に描き目標軌道とした.

図 4.11 はシミュレーションの結果であり,図 4.11(a) はそれぞれの制御器を用いた場合のロボットの移動軌跡を示している.移動軌跡について大きな違いは見られないが,大きな動作ノイズを付加した制御器 F が最も軌道の内側を通っていることが分かる.図 4.11(b) はロボットの位置の x 座値の時間変化を示したものであり,付加する誤差を大きくするにつれて終端に達する時刻が早くなっていることがわかる.図 4.11(c) および図 4.11(d) はそれぞれ状態変数 α および β の時間変化を示したものであり,付加する動作ノイズを大きくするにつれてそれぞれの値の振幅が大きくなっていることが分かる.これらのシミュレーション結果より,動作ノイズが誤差 β_e に影響するため,これが大きい場合は誤差を小さくすることには限界があり,終端への到達時刻 t_f をより小さくする個体が高い評価値を得たのではないかと考えられる.

図 4.12 と図 4.13 はそれぞれの制御器を実機に搭載した場合の結果を示してい る.図4.12(a)は地面に描かれた目標軌道を示している.図4.12(b)から図4.12(d) はそれぞれ制御器 D. E. F を搭載した際の実機の移動軌跡を 1 [s] 間隔で背景差分 により描画したものである.これより実機実験では制御器 E の場合のみが目標軌 道の終端へ到達出来ていることがわかる.また,制御器Dの場合は他の2つと比 べてロボットの移動速度が極端に遅いことがわかる.図 4.13(a) および図 4.13(b) はそれぞれ状態変数 α および β の時間変化を示したものである.制御器 D の場合 は 15 [s] 以降,両方の値が一定でありカメラ画像上に目標軌道を捉えているにも かかわらずロボットが停止していることがわかる.また,直線部分のみを追従し ているにもかかわらず β の値が大きく振動しており,これはロボットが左右に小 刻みに振動していることを意味している.一方で,制御器 E,F の場合は制御器 D の場合と比べて振幅が小さく,図4.12の画像からもわかるように,目視では確認 することが難しい程度まで振動を抑制できている.しかしながら,制御器Fの場 合は20[s]付近で目標軌道を見失ったため,付加する動作ノイズが大きすぎると誤 差に対して鈍感になるものと考えられる.これらの結果より,GAによる設計時の シミュレーションに適切な動作ノイズを付加することにより、制御器のロバスト

性が向上し実機でも目標軌道を追従することが可能であることが確認された.



(a) Robot trajectory in simulation

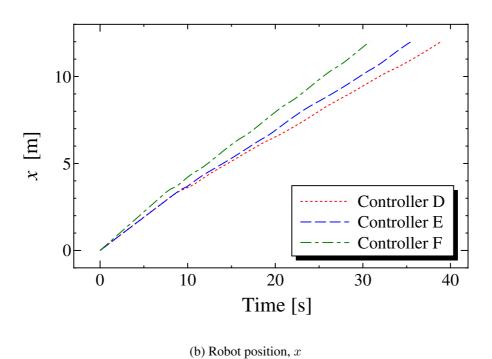


Fig. 4.11: Simulation results when using the fuzzy controllers D, E and F

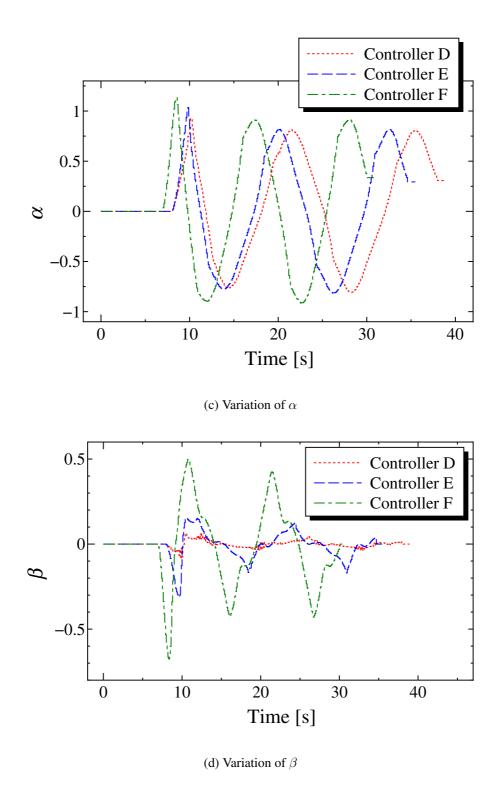


Fig. 4.11: Simulation results when using the fuzzy controllers D, E and F

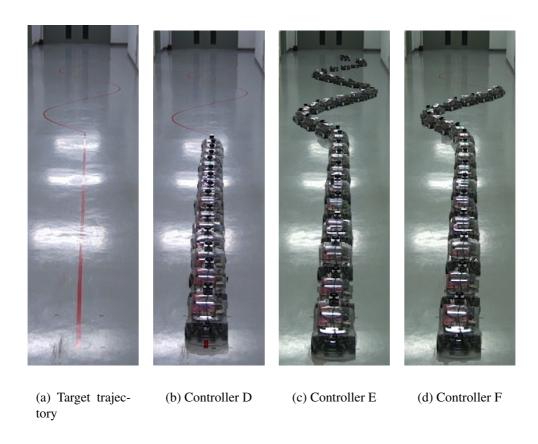


Fig. 4.12: Experimental results when using the fuzzy controllers D, E and F, where the robot motion was picked up every 1.0 [s] interval

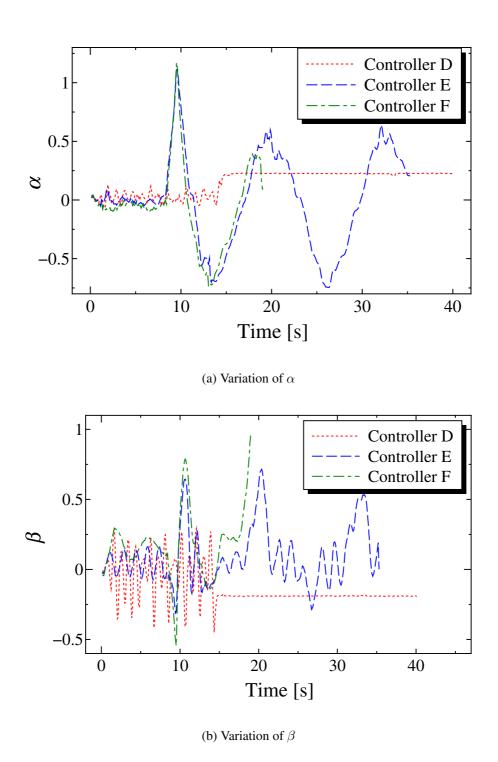


Fig. 4.13: State variables α and β in experiments when using the fuzzy controllers D, E and F

4.6 まとめ

本章では任意軌道追従を行うための画像に基づくファジィ制御器を GA によって設計する手法を提案した.また,GA を用いてファジィ制御器を設計する際にノイズを付加することで制御器のロバスト性を向上させることを試みた.付加するノイズの種類によるロボットの挙動の違いについて実験を行い,ロボットの動作ノイズを与えた場合の方が,より実機に適用するのに適した制御器が設計される事がわかった.さらに,評価関数と動作ノイズの与え方を変更した場合について,評価関数の重み係数とロボットの挙動の関係についてシミュレーションによる評価を行った.結果より,状態変数の誤差に関する重み係数を大きくしすぎると,目標軌道の直線部分で停止してしまうロボットがより高い評価値を得ることになるため,ロボットが進んだ距離に関する重み係数の大きさとの関係が重要であることがわかった.また,この場合についても実機実験を行い,適切な大きさの動作ノイズを付加することによりファジィ制御器のロバスト性が向上し,実機に搭載した場合でも目標軌道を追従することが可能であることを確認した.

第5章 画像に基づく移動ロボットの 障害物回避

本章では前章までにて提案した画像に基づくファジィ制御器を用いた軌道追従 手法の応用として,画像に基づく障害物回避の手法を提案する.これまで提案さ れてきた移動ロボットの障害物回避の手法の多くは, さまざまなセンサを利用し て障害物までの距離を計測し、その距離に基づいてロボットの動作を決定するも のであった.その例として,城殿ら (68) はステレオカメラを用いて距離画像を作成 し、移動ロボットと障害物の距離を基にロボットを目標位置まで最短距離でナビ ゲーションする手法を提案した、Blackmore ら (69) らは飛行ロボットを制御対象と し、ロボットの自己位置と障害物までの距離に関する確率的な制約の最適化を行 うことで障害物回避を行う手法を提案した、本章にて提案する手法は、カメラ画 像の画像座標上に障害物回避を行う目標軌道を作成し,その軌道を画像に基づく ファジィ制御器で追従するというものである、したがって、障害物までの正確な 距離を計測する必要がないという利点がある、物体にはそれぞれ基調となる色が あり、色の変化が激しい所は異なる物体の境目であると仮定すると、エッジ検出 によって障害物を検出することができると考えられる.この仮定は,自然界の擬 熊化する生物が周囲の物体と色を合わせて同化するということの逆を考えると妥 当であると考えられる.

5.1 問題設定

図 5.1 は提案手法の概要を表したものである.この図に示すように世界座標を x-y 座標として定義し,カメラ画像の画像座標を u-v 座標として定義する.障害物 回避を行うための目標軌道は画像座標上に始点を (u_1,v_1) とし,終点を (u_2,u_2) とする線分として表すものとする.この線分は 3 つのパラメータ傾き,切片および 距離を (α,β,λ) として表され,それらは以下の式によって求めるものとする.

$$\alpha = \frac{u_2 - u_1}{v_2 - v_1}$$

$$\beta = v_1$$

$$\lambda = \sqrt{(u_2 - u_1)^2 + (v_2 - v_1)^2}$$
(5.1)

制御対象はこれまでと同様に図 3.2 に示す操舵型の 4 輪移動ロボットとし,位置を (x,y,θ) ,制御入力を (s,ϕ) として x-y 平面上を移動するものとする.ここで,s は 前進速度であり, ϕ は前輪の操舵角である.また,カメラは進行方向前方を向いて いるものとする.

図 5.2 は提案手法のブロック線図を表したものである.ロボットには前進の命令が与えられているものと仮定する.取得したカメラ画像から画像処理によって検出されたエッジを避けるように目標軌道を生成する.この軌道を追従するための制御器として画像に基づくファジィ制御器を設計する.このブロック線図より,確かにロボットや障害物の位置推定を行うことなく画像座標上の情報を基に障害物回避が行われていることが分かる.提案手法は大きく分けて以下のように2つの要素にまとめられる.

1. 目標軌道の生成 画像座標上にエッジを避けるような目標軌道を生成する.

2. 軌道の追従

画像座標に生成された目標軌道のパラメータ (α, β, λ) を基にこの軌道を追従するための制御入力 (s, ϕ) を決定する.

次節より,それぞれの要素の詳細について述べる.

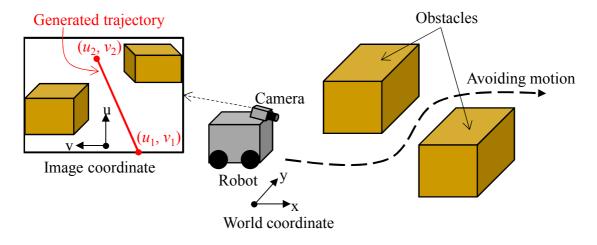


Fig. 5.1: Overview of the present image-based obstacle avoidance

5.2 障害物回避軌道の生成

目標軌道の生成の手順は以下のように3つの手順からなる.

1. ポテンシャルの計算

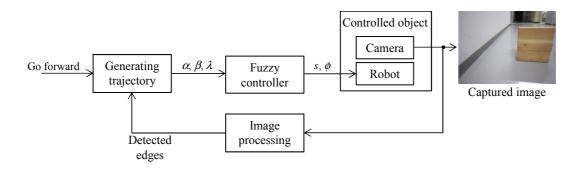


Fig. 5.2: Block diagram of the present image-based obstacle avoidance

2. 走行可能領域の判別

3. 目標軌道の生成

目標軌道を生成するために,まず,エッジに基づくポテンシャルを利用して走行可能領域の判別を行う.図 5.2 に示すようにカメラ画像の左側に壁,右側に直方体の障害物が映っている場合を例にそれぞれの手順について説明する.なお,説明の都合上,画像の大きさは 15 [pixel] \times 10 [pixel] とする.

5.2.1 ポテンシャルの計算

図 5.3(a) は取得したカメラ画像に対して画像処理を行い検出されたエッジを表している.注目画素の座標を (u_i,v_i) と表した場合,その画素のポテンシャルは関数 $U(u_i,v_i)$ として以下の式より計算される.

$$U(u_i, v_i) = \arg\min_{1 \le k \le N} (|u_i - u_k| + |v_i - v_k|)$$
(5.2)

ただし,N は検出されたエッジ画素の総数である.図 5.3(b) は求められた各画素のポテンシャルを表しており,各画素の数値は $U(u_i,v_i)$ の値を示したものである.この図より, $U(u_i,v_i)$ の値が各画素から最も近いエッジまでの距離となっていることがわかる.

5.2.2 走行可能領域の判別

各画素が有するポテンシャルはエッジまでの距離を表しており,その値が大きければよりエッジから離れていることを意味する.そこで,閾値 Θ を導入し,この値以上のポテンシャルを有する画素は走行可能領域とみなすこととする.適切な閾値 Θ は実験によって経験的に求めることが必要である.例として図 5.3(c) の 黄色い領域は閾値を 5 とした場合の走行可能領域を表している.

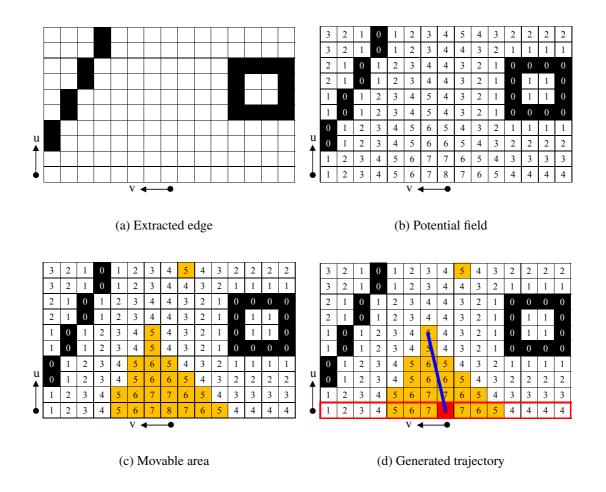


Fig. 5.3: Illustration of generating a trajectory for the obstacle avoidance

5.2.3 目標軌道の生成

目標軌道は検出された走行可能領域上に生成されるものとする. 軌道の始点 (u_1,v_1) は画像の下端の行のうち最大のポテンシャルを有する画素とする. 終点 (u_2,v_2) は始点が含まれている連続した走行可能領域内の最大のu 座標値を有する 画素とする. 図 5.3(d) には生成された目標軌道を青色の線分として描いている. 画像の上端にも走行可能領域が検出されているが,終点は始点が含まれている連続した走行可能領域内に設定するという条件によってこの不適切な点が選ばれる事を 防いでいる.この例では左側の壁の手前で停止するような軌道が生成されている.

5.3 障害物回避のためのファジィ制御器

カメラ画像上に生成した目標軌道を追従するための画像に基づくファジィ制御器を設計する、本章で設計するファジィ制御器は目標軌道の傾き α と切片 β に加

えて,目標軌道の長さλを考慮するものである.設計した制御器はシミュレーションと実機実験によって性能の評価を行う.

5.3.1 ファジィ制御器の設計

ここでは, Mamdani が行った「min-max-重心法」を用いたファジィ制御器を設計する. 予備実験を通して経験的に設計したメンバシップ関数および制御規則をそれぞれ図 5.4 および Table 5.1, Table 5.2 に示す. 図 5.4 の N, ZO, P, PS, PB はそれぞれ Negative, Zero, Positive, Positive Small, Positive Big を意味する.

これより,目標軌道のパラメータから制御入力を決定する過程について説明する.ただし,それらのパラメータは状態変数として扱うこととする.ファジィ制御器はまず,状態変数を言語表現を要素とするファジィ集合へと置き換えるファジィ化を行う.ファジィ集合へと置き換えられた状態変数をそれぞれ $(\tilde{\alpha}, \tilde{\beta}, \tilde{\lambda})$ とすると,それぞれのファジィ集合は以下のように言語表現を要素とする.また,制御入力も一度ファジィ集合 $(\tilde{s}, \tilde{\phi})$ として求められるため,ここで要素を定義しておくこととする.

$$\tilde{\alpha} = \{N, ZO, P\}$$

$$\tilde{\beta} = \{N, ZO, P\}$$

$$\tilde{\lambda} = \{ZO, PS, PB\}$$

$$\tilde{s} = \{ZO, PS, PB\}$$

$$\tilde{\phi} = \{N, ZO, P\}$$

$$(5.3)$$

メンバシップ関数は状態変数の値から前件部への適合度を求める.前件部が " $\tilde{\alpha}=N$ である" の場合,その適合度を求めるメンバシップ関数は $h_{\tilde{\alpha},N}(\alpha)$ と表すこととする.注意すべきは,図 5.4 中の α に関するメンバシップ関数のグラフは $h_{\tilde{\alpha},N}(\alpha)$, $h_{\tilde{\alpha},ZO}(\alpha)$, $h_{\tilde{\alpha},P}(\alpha)$ の 3 つの関数を含んでいるということであり,他の場合も同様である.それぞれのメンバシップ関数の値域は [0,1] である.

次に , メンバシップ関数によって求められた適合度は , 制御規則の適合度 $h_{\mathbf{R}_n}(1 \le n \le 14)$ の計算に利用され , 表 5.1 および , 表 5.2 を基に以下のように行われる

$$h_{R_{1}} = \min \left\{ h_{\tilde{\alpha},N}(\alpha), h_{\tilde{\beta},N}(\beta) \right\}$$

$$h_{R_{2}} = \min \left\{ h_{\tilde{\alpha},N}(\alpha), h_{\tilde{\beta},ZO}(\beta) \right\}$$

$$\vdots$$

$$\vdots$$

$$h_{R_{14}} = \min \left\{ h_{\tilde{\lambda},PB}(\lambda), h_{\tilde{\phi},P} \right\}$$
(5.4)

ここで , Table 5.2 の制御規則の適合度である $h_{\rm R_{10}}$ から $h_{\rm R_{14}}$ を求める式には $\tilde{\phi}$ に関する適合度の値, $h_{\tilde{\phi},\rm N}$, $h_{\tilde{\phi},\rm ZO}$, $h_{\tilde{\phi},\rm P}$ が含まれているため , 先にこれらの値を求めておくことが必要である .

求めた各制御規則の適合度は制御入力に対する後件部の適合度の計算に利用される.この適合度の表記の例として" $\tilde{\phi}=N$ とする"という後件部への適合度は $h_{\tilde{\phi},N}$ と表すこととする.後件の適合度はそれぞれ対応する制御規則の適合度に基づいて以下のように求められる.

$$h_{\tilde{\phi},N} = \max\{h_{R_{1}}, h_{R_{2}}, h_{R_{4}}\}$$

$$h_{\tilde{\phi},ZO} = \max\{h_{R_{3}}, h_{R_{5}}, h_{R_{7}}\}$$

$$h_{\tilde{\phi},P} = \max\{h_{R_{6}}, h_{R_{8}}, h_{R_{9}}\}$$

$$h_{\tilde{s},ZO} = h_{R_{10}}$$

$$h_{\tilde{s},PS} = \max\{h_{R_{11}}, h_{R_{12}}, h_{R_{14}}\}$$

$$h_{\tilde{s},PB} = h_{R_{13}}$$

$$(5.5)$$

最後に,求められた後件部の適合度を用いて非ファジィ化を行いファジィ集合としての制御入力 $(\tilde{s},\tilde{\phi})$ を実数としての (s,ϕ) とする.図 5.4 に示すように制御入力のためのメンバシップ関数はシングルトンを用いているため,非ファジィ化は以下のように重み付き平均を求める計算によって行われる.

$$s = \frac{0 \cdot h_{\tilde{s},ZO} + 0.5 \cdot h_{\tilde{s},PS} + 1.0 \cdot h_{\tilde{s},PB}}{h_{\tilde{s},ZO} + h_{\tilde{s},PS} + h_{\tilde{s},PB}}$$

$$\phi = \frac{-0.436 \cdot h_{\tilde{\phi},ZO} + 0 \cdot h_{\tilde{\phi},PS} + 0.436 \cdot h_{\tilde{\phi},PB}}{h_{\tilde{\phi},ZO} + h_{\tilde{\phi},PS} + h_{\tilde{\phi},PB}}$$
(5.6)

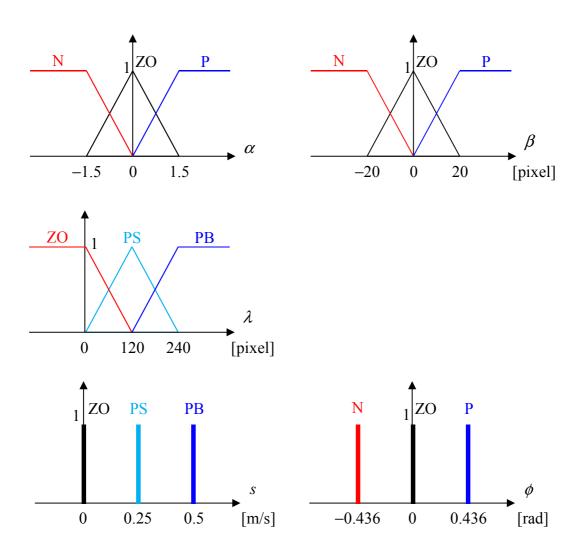


Fig. 5.4: Membership functions for obstacle avoidance

Table 5.1: Fuzzy rules for the steering angle ϕ

No.	Antecedent		Consequent
	\tilde{lpha}	$ ilde{eta}$	$ ilde{\phi}$
R_1	N	N	N
R_2		ZO	N
R_3		P	ZO
R_4	ZO	N	N
R_5		ZO	ZO
R_6		P	P
R ₇	P	N	ZO
R_8		ZO	P
R_9		P	P

Table 5.2: Fuzzy rules for the translational velocity \boldsymbol{s}

No.	Antecedent		Consequent
	$\tilde{\lambda}$	$ ilde{\phi}$	\tilde{s}
R ₁₀	ZO	_	ZO
R ₁₁	PS	_	PS
R_{12}	РВ	N	PS
R_{13}		ZO	PB
R ₁₄		P	PS

5.3.2 ファジィ制御器の性能評価実験

本章で設計したファジィ制御器を障害物回避問題へ適用する前に,ここでは軌道追従問題でのシミュレーションと実機実験を行い,その性能を評価した.シミュレーションでは以下に示すロボットの動作モデルを用いた.

$$\dot{x} = s \cos \theta
\dot{y} = s \sin \theta
\dot{\theta} = \frac{s \tan \phi}{L}$$
(5.7)

ここで,L は前輪と後輪の間の距離である.この制御の目的は地面に描かれた線分を目標軌道として追従することである.ロボットは搭載されたカメラの画像を基に設計した制御器を用いて制御される.目標軌道は世界座標上に振幅 $0.15~[\mathrm{m}]$, 周期 $4.0~[\mathrm{m}]$ の矩形波 y=f(x) として以下の式で表すこととする.

$$y = 0.15\operatorname{sgn}\left\{\sin\left(2\pi\frac{x}{4}\right)\right\} \tag{5.8}$$

ただし,この振幅の大きさはカメラの視野を考慮して設定された値である.ロボットに搭載されているカメラは高さ 0.2~[m] の位置に鉛直下向きより前方へ 60~[deg] 傾けた状態で搭載されているものとした.カメラの水平画角は 52~[deg] とし,垂直画角は 40~[deg] とした.ロボットの初期状態は目標軌道をカメラ画像に映すために原点ではなく $(x,y,\theta)=(-0.5,0,0)$ とした.

シミュレーションの結果を図 5.5 と図 5.6 に示す.図 5.5 は目標軌道とロボットが移動した軌跡を表したものである.この結果よりロボットが確かに目標軌道を大きいオーバーシュートを発生することなく追従していることが分かる.図 5.6 はロボットの前進速度の記録を表したものである.図 5.5 と図 5.6 を見比べると,軌道の終点 $(x=6\ [m]\ 付近)$ で s=0 でロボットが停止している.ロボットの向きが大きく変わっているところ $(x=2\ [m], 4\ [m]\ 付近)$ では減速していることが分かる.結果より,設計したファジィ制御器は理想的な環境下で確かに目標軌道を追従出来ることが証明された.

次に,実環境下での制御器の性能を評価するために実機実験を行った.実験条件はシミュレーションと同じであり,目標軌道である矩形波 (式 (5.8)) は床に描いた.実験用のロボットはこれまでと同様に 3 章にて使用した市販のラジコンカーを改造して製作したものを用いた.図 3.11 に示すようにボディの上部に USB カメラが取り付けられている.搭載された小型ノート PC は画像処理を行いカメラ画像から目標軌道のパラメータ (α,β,λ) を抽出し,設計した制御器に基づいて制御入力 (s,ϕ) を決定する.その画像処理には直線を抽出するためのハフ変換が含まれており,画像内に複数の線分が映っている場合は,投票数が最も多いものを目標軌道として採用することとした.マイクロコントローラはその制御入力を実現するためにサーボモータの制御を行う.

実験結果を図 5.7 に示す.図 5.7(a) は目標軌道の写真であり,図 5.7(b) はビデオカメラで撮影した制御結果の動画を基に,背景差分を用いて 0.5 [s] 間隔でロボットの部分を抜き出して表示したものである.結果より,設計したファジィ制御器は実環境下でも同じように機能することが証明された.結果より,実機においても設計したファジィ制御器によって地面に描かれた目標軌道を追従することが可能であることが分かった.したがって,取得したカメラ画像に対して障害物を回避するための適切な目標軌道を生成することができれば,設計した制御器によってその目標軌道を追従することができると思われる.

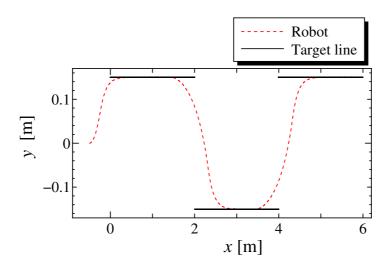


Fig. 5.5: Simulation result of the trajectory of the robot

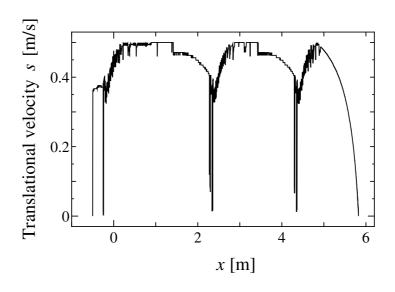


Fig. 5.6: Translational velocity of the robot on the simulation





(a) Target trajectory for the experiment to evaluate the fuzzy controller

(b) Appearance of trajectory tracking of the real robot was picked up every 0.5 s interval

Fig. 5.7: Actual experimental results for testing the fuzzy controller

5.4 実機実験

提案した画像に基づく障害物回避手法を評価するために,実機実験を行った.実験環境は図5.8に示すように平坦な廊下に障害物として角材を設置したものである.実験に使用されるロボットは上述の実験と同じものとした.取得する画像の大きさは320 [pixel] \times 240 [pixel] とし,走行可能領域を抽出するための閾値は画像の横幅の画素数の半分の $\Theta=160$ と設定した.ロボットの初期位置は図5.8 に示すようにA, B, C の3 通りとした.望まれるロボットの動作は障害物回避を行いながら前進し,最終的に壁や障害物の手前で停止することである.

実験結果を図 5.9 から図 5.11 に示す.これらはビデオカメラで撮影した動画を基に,背景差分を用いて 0.5 [s] 間隔でロボットの部分を抜き出して表示したものである.図 5.9 は初期位置を B とした場合の結果であり,特に,(1) 初期位置,(2) 左旋回を行う瞬間,(3) 停止位置における画像処理の様子を示している.この図では,検出されたエッジは赤色の点としてプロットされ,ポテンシャル値の大きさは緑色の明るさによって表現されており,算出された走行可能領域は黄色に塗られている.生成された目標軌道は青色の線分で表されており,確かにエッジを避けるように目標軌道が生成されていることが分かる.図 5.10 と図 5.11 はそれぞれ初期位置を A とした場合と B とした場合の結果である.すべての結果において,ロボットは初期位置から左右の障害物の間の空間に向かって移動し,その後,正面と右側にある障害物を回避し,それらの障害物の先にある壁の前で停止した.この動作は望み通りの結果であり,提案手法の有用性を示すことができた.

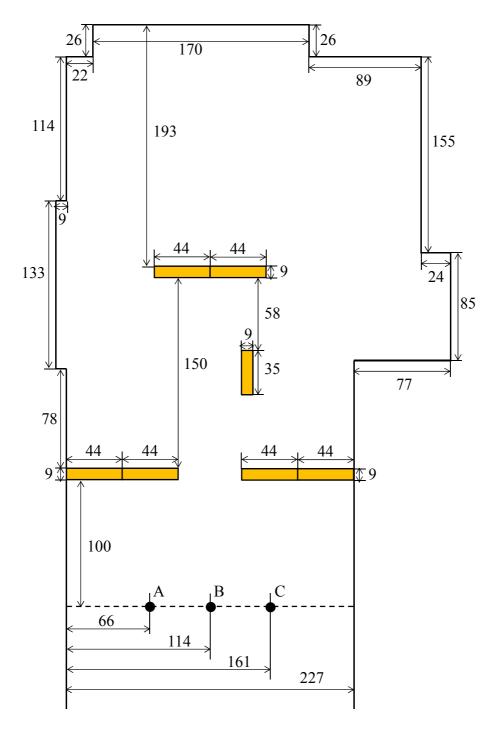


Fig. 5.8: Experimental environment (unit in cm)

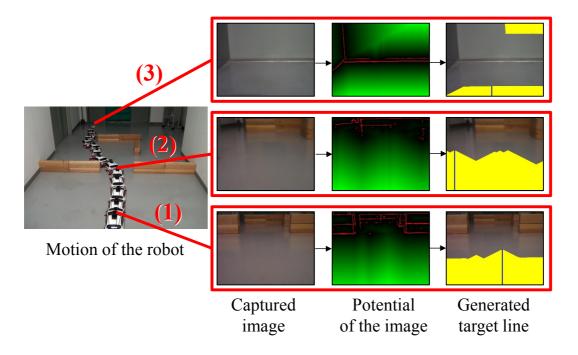


Fig. 5.9: Robot movement and image processing in the case when the initial position was assigned to B



Fig. 5.10: Experimental result in the case when the initial position was assigned to A



Fig. 5.11: Experimental result in the case when the initial position was assigned to C

5.5 まとめ

本章では画像に基づくファジィ制御器の応用として,画像に基づく移動ロボットの障害物回避手法を提案した.提案手法のコンセプトは画像内のエッジを避けるようにロボットの制御入力を決定するという単純なものであるが,実機実験によってその有用性を示した.この手法の利点として,ロボットから障害物までの距離計測が不要であることが挙げられる.

第6章 結論

本論文では視覚サーボの概要について述べ,その主な手法である位置に基づく 視覚サーボと画像に基づく視覚サーボの特徴について述べた.特に画像に基づく 視覚サーボは移動ロボットの制御手法として従来から行われてきた自己位置推定 を行うことなく移動ロボットを制御することが可能な制御手法であった.また,正 確な自己位置を参照しないという点で,より人間的な制御手法であると考えられ るため,経験に基づくファジィ制御器によって画像に基づく視覚サーボ行う手法 を提案した.

ファジィ制御器を用いて画像に基づく視覚サーボを行う利点を述べるために,まず,移動ロボットによる直線軌道追従の問題をモデルベースの視覚サーボによって行う方法について述べた.この方法では,画像特徴量とロボットの動作の関係をモデル化した相互行列を用いるが,その導出にはカメラが鉛直下向きであるという条件が必要であり,カメラの向きが変化した場合は回転行列を用いて座標変換を行うことが必要であった.また,注目画素の奥行きを推定する必要があるため,画像に基づく視覚サーボの位置推定が不要であるという利点を損なうことが懸念された.これに対して,提案手法であるファジィ制御器を用いた画像に基づく視覚サーボの手法は相互行列を用いることなく制御器を設計することが可能であった.

操舵型の4輪移動ロボットを制御対象とした直線軌道追従問題について2種類のファジィ制御器を設計した.1つ目は人間が行う自動車の運転を基に設計したものであり,人間が行うハンドルの操作をロボットの前輪の操舵角の角速度と対応させ,アクセルペダルの操作をロボットの前進加速度に対応させたファジィ制御器を設計した.この制御器についてシミュレーションを行い,ファジィ制御器によって画像に基づく視覚サーボを行うことが可能であることを示した.また,速度制御を行うことでオーバーシュートが軽減し,速度制御を行うことの有用性を示した.もう1つの制御器は,実機実験を行うことを目的として,制御入力を前輪の操舵角と前進速度に置き換え,状態変数の変化量を考慮したものであった.この制御器を用いてシミュレーションと実機実験の両方を行った結果,残留誤差がわずかに残ったものの,提案した画像に基づくファジィ制御器を用いてロボットが目標の直線軌道上を前進することを示した.

次に,任意軌道追従を行うためのファジィ制御器をGAによって設計する手法について述べた.GAを用いた制御器の設計では,個体の評価を行うための目標軌道に特化した制御器が設計されてしまう傾向にあるため,設計時のシミュレーション

にノイズを付加することで制御器のロバスト性を向上させる手法を提案した.付加するノイズの種類や大きさと設計される制御器の関係をシミュレーションと実機実験によって調査し,適切な大きさのノイズを付加することでシミュレーションだけでなく実機によって任意軌道を追従することが可能であることを示した.

提案した画像に基づくファジィ制御器の応用として,画像に基づく障害物回避の手法を提案した.画像のエッジを地面と障害物との境目とみなし,エッジを避けて前進する目標軌道を画像上に生成し,その軌道のパラメータを基にロボットの動作を決定するファジィ制御器を設計した.実機実験を行った結果,提案手法によって障害物を回避しながら前進することが可能であることを示した.

画像に基づく視覚サーボの特徴として,自己位置推定を行わないことによる処理速度の速さが挙げられるが,上記のように直線追従や任意軌道追従を行った結果,完全に目標軌道の真上を通ることはできなかった.一方で,位置に基づく視覚サーボの手法は処理速度が遅い半面,より正確な位置制御を行うことができる.したがって,カメラを用いて移動ロボットを制御する場合には,まず,画像に基づく視覚サーボを用いて大まかな位置決めを高速に行い,その後,より詳細な位置決めを位置に基づく視覚サーボによって行うことが必要であると考えられる.また,画像に基づく視覚サーボによって制御されるロボットに,従来の位置に基づくナビゲーション手法を適用することはできないため,新たなナビゲーション手法を考える必要がある.

軌道追従での目標軌道に接近した場合に、減速してカメラの向きを鉛直下向きすることで誤差をより小さくすることが考えられる.この方法についても、最適なカメラの向きを理論的に決定することは容易ではないため、GAを用いてカメラの向きを調節する制御規則を探索することでファジィ制御器を設計できると思われる.また、現存するロボットのナビゲーション手法には障害物やロボットの位置に基づくものが数多く提案されているが、これらの手法に画像特徴量の情報を追加することで画像に基づく視覚サーボによって制御されるロボットのナビゲーションを行うことが可能であると考えられる.さらに、移動ロボットの軌道追従だけではなく、より複雑な問題設定としてハンドアイシステムのエンドエフェクタの姿勢制御をファジィ制御器によって行うことが考えられる.

謝辞

本研究の遂行において終始懇切なるご指導とご鞭撻を賜り,本論文をまとめる に際して親身なご助言を頂きました自然科学研究科産業創成工学専攻の渡辺桂吾 教授に,心より感謝を申し上げます.

本論文の審査過程において,数々のご助言とご指導を承りました,自然科学研 究科産業創成工学専攻の則次俊郎教授と鈴森康一教授に深く感謝いたします.

学部在籍時より様々なご助言を賜りました自然科学研究科産業創成工学専攻の 先生方に感謝いたします.

本研究の遂行において,日々熱い激励を頂きましたメカトロニクスシステム学研究室の三賢人こと甲斐照人氏,田村祥氏,上井康弘氏および山本泰史氏に心より感謝を申し上げます.

本研究の遂行において,結果が出ず,精神的に辛いときにも明るく私を支えて下さいましたペリカン津高店のみなさん,本当にありがとうございました.

最後に6年間という長い間,楽しいときも苦しいときも共に過ごし,支え続けてくださったメカトロニクスシステム学研究室の先輩,同期,後輩のみなさんに心から感謝します.本当にありがとうございました.

付録A リヤプノフの安定性理論

ある非線形システムの n 次元状態ベクトルを x とし , そのシステムが次式で表されるものとする .

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}) \tag{A.1}$$

このとき, $f(\mathbf{0})=\mathbf{0}$ となる原点の安定性を考える.もし,目標の状態 $x_{\mathbf{d}}$ が原点ではない場合でも, $e=x-x_{\mathbf{d}}$ として, $\dot{e}=f(e)$ のシステムを考えるとよい.あるスカラ関数 V(x) について,その時間 t に関する 1 階偏導関数 $\dot{V}(x)$ を考えると,

$$\dot{V}(\boldsymbol{x}) = \frac{\partial V(\boldsymbol{x})}{\partial x_1} \frac{dx_1}{dt} + \frac{\partial V(\boldsymbol{x})}{\partial x_2} \frac{dx_2}{dt} + \dots + \frac{\partial V(\boldsymbol{x})}{\partial x_n} \frac{dx_n}{dt}$$

$$= \sum_{i=1}^n \frac{\partial V(\boldsymbol{x})}{\partial x_i} \dot{x}_i$$

$$= \sum_{i=1}^n \frac{\partial V(\boldsymbol{x})}{\partial x_i} f_i(\boldsymbol{x}) \tag{A.2}$$

と書くことができる.このとき, $\dot{V}(x)$ が常に負となるような V(x) を見出すことができれば,時間の経過と共に V(x) は次第に小さな値をとる.ここで,V(x) が正定値であれば V(x) が小さくなると $\|x\|$ も共に小さくなる.したがって, $t\to\infty$ のときに $V(x)\to 0$ となり,同様に $x\to 0$ となる.つまり,あるスカラ関数 V(x) が以下の条件を満たす場合,そのシステムは原点付近で漸近安定である $^{(74)}$.

- 1. $V(\mathbf{0}) = 0$
- 2. $\boldsymbol{x} \neq \boldsymbol{0}$ のとき , $V(\boldsymbol{x}) > 0$
- 3. $oldsymbol{x}
 eq oldsymbol{0}$ のとき, $\dot{V}(oldsymbol{x}) < 0$

この条件を満たす関数 V(x) はリヤプノフ関数と呼ばれる.したがって,あるシステムに関して,リヤプノフ関数の候補を立て,上記の条件を満たすように制御入力を設計することでシステムの安定化を行うことが可能となる.このリヤプノフ関数は一意ではなく,あるリヤプノフ関数が他のリヤプノフ関数よりも有用な結果を与えることもある.また,あるスカラ関数 V(x) について上記の条件を満たすことができなかったとしても,そのシステムが不安定であるとは限らない.

参考文献

- (1) Ward, C. C. and Iagnemma, K., "A Dynamic-Model-Based Wheel Slip Detector for Mobile Robots on Outdoor Terrain", *IEEE Trans. on Robotics*, Vol. 24, No. 4 (2008), pp. 821–831.
- (2) Nasrallah, D. S., Michalska, H., and Angeles, J., "Controllability and Posture Control of a Wheeled Pendulum Moving on an Inclined Plane", *IEEE Trans. on Robotics*, Vol. 23, No. 3 (2007), pp. 564–577.
- (3) Magid, E., Tsubouchi, T., Koyanagi, E., Yoshida, T., and Tadokoro, S., "Controlled Balance Losing in Random Step Environment for Path Planning of a Teleoperated Crawler-Type Vehicle", *J. of Field Robotics*, Vol. 28, No. 6 (2011), pp. 932–949.
- (4) Arai, M., Tanaka, Y., Hirose, S., Kuwahara, H., and Tsukui, S., "Development of "Souryu-IV" and "Souryu-V: "Serially Connected Crawler Vehicles for in-Rubble Searching Operations", *J. of Field Robotics*, Vol. 25, No. 1 (2007), pp. 31–65.
- (5) Erden, M. S. and Leblebicioglu, K. "Torque Distribution in a Six-Legged Robot", *IEEE Trans. on Robotics*, Vol. 23, No. 1 (2007), pp. 179–186.
- (6) Valdastri, P., Webster, R. J., Quaglia, C., Quirini, M. Menciassi, A., and Dario, P., "A New Mechanism for Mesoscale Legged Locomotion in Compliant Tubular Environments", *IEEE Trans. on Robotics*, Vol. 25, No. 5 (2009), pp. 1047–1057.
- (7) Neo, E. S., Yokoi, K., Kajita, S., and Tanie, K., "Whole-Body Motion Generation Integrating Operator 's Intention and Robot 's Autonomy in Controlling Humanoid Robots", *IEEE Trans. on Robotics*, Vol. 23, No. 4 (2007), pp. 763–775.
- (8) Morimoto, J., Endo, G., Nakanishi, J., and Cheng, G., "A Biologically Inspired Biped Locomotion Strategy for Humanoid Robots: Modulation of Sinusoidal Patterns by a Coupled Oscillator Model", *IEEE Trans. on Robotics*, Vol. 24, No. 1 (2008), pp. 185–191.

- (9) Trulls, E., Murtra, A. C., Perez-lbarz, J., Ferrer, G., Vasquez, D., Mirats-Tur, J. M., and Sanfeliu, A., "Autonomous Navigation for Mobile Service Robots in Urban Pedestrian Environments", *J. of Field Robotics*, Vol. 28, No. 3 (2011), pp. 329–354.
- (10) Huang, S. and Dissanayake, G., "Convergence and Consistency Analysis for Extended Kalman Filter Based SLAM", *IEEE Trans. on Robotics*, Vol. 23, No. 5 (2007), pp. 1036–1049.
- (11) Razali, S., Watanabe, K., and Maeyama, S., "The Spherical Simplex Unscented Transformation for a FastSLAM", *J. of System Design and Dynamics, The Japanese Society of Mechanical Engineering*, Vol. 6, No. 5 (2012), pp. 713–728.
- (12) Bryson, M. and Sukkarieh, S., "Building a Robust Implementation of Bearing-only Inertial SLAM for a UAV", *J. of Field Robotics*, Vol. 24, No. 1 (2007), pp. 113–143.
- (13) Zhao, S. and Yuh, J., "Experimental Study on Advanced Underwater Robot Control", *IEEE Trans. on Robotics*, Vol. 21, No. 4 (2005), pp. 695–703.
- (14) Bingham, B., Foley, B., Singh, H., Camilli, R., Delaporta, K., Eustice, R., Mallios, A., Mindell, D., Roman, C., and Sakellariou, D., "Robotic Tools for Deep Water Archaeology: Surveying an Ancient Shipwreck with an Autonomous Underwater Vehicle", *J. of Field Robotics*, Vol. 27, No. 6 (2010), pp. 702–717.
- (15) Zhang, Z. G., Yamashita, N., Gondo, M., Yamamoto, A, and Higuchi, T., "Electrostatically Actuated Robotic Fish: Design and Control for High-Mobility Open-Loop Swimming", *IEEE Trans. on Robotics*, Vol. 24, No. 1 (2008), pp. 118–129.
- (16) Krieg, M. and Mohseni, K., "Dynamic Modeling and Control of Biologically Inspired Vortex Ring Thrusters for Underwater Robot Locomotion", *IEEE.Trans. on Robotics*, Vol. 26, No. 3 (2010), pp. 26, No. 3 (2010), pp. 542–554.
- (17) Frazzoli, E., Dahleh, A. D., and Feron, E., "Maneuver-Based Motion Planning for Nonlinear Systems With Symmetries", *IEEE Trans. on Robotics*, Vol. 21, No. 6 (2005), pp. 1077–1091.
- (18) Watanabe, K., Tanaka, K., Izumi, K., Okamura, K., and Syam, R., "Discontinuous Control and Backstepping Method for the Underactuated Control of VTOL Aerial Robots with Four Rotors", *Intelligent Unmanned Systems: Theory and Applications*, (2009), pp. 83–100, Springer.
- (19) Grzonka, S., Grisetti, G., and Burgard, W., "A Fully Autonomous Indoor Quadrotor", *IEEE Trans. on Robotics*, Vol. 28, No. 1 (2012), pp. 90–100.

- (20) Cabecinhas, D., Naldi, R., Marconi, L., Silvestre, C., and Cunha, R., "Robust Take-Off for a Quadrotor Vehicle", *IEEE Trans. on Robotics* Vol. 28, No. 3 (2012), pp. 734–742.
- (21) 山田誠二, 斎藤淳也, "マルチロボットによる箱押しのための明示的通信を用いない適応的行為選択", 日本ロボット学会誌, Vol. 17, No. 6 (1999), pp. 818–827.
- (22) Huntsberger, T., Stroupe, A., Aghazarian, H., Garrett, M., Younse, P., and Powell, M., "TRESSA: Teamed Robots for Exploration and Science on Steep Areas", *J. of Field Robotics*, Vol. 24, No. 1 (2007), pp. 1015–1031.
- (23) Burgard, W., Moors, M., Stachniss, C., and Schneider, F. E., "Coordinated Multi-Robot Exploration", *IEEE Trans. on Robotics*, Vol. 21, No. 3 (2005), pp. 376–386.
- (24) Kwon, J.-W. and Chwa, D., "Hierarchical Formation Control Based on a Vector Field Method for Wheeled Mobile Robots", *IEEE Trans. on Robotics*, Vol. 28, No. 6 (2012), pp. 1335–1345.
- (25) Savkin, A. V. and Teimoori, H., "Decentralized Navigation of Groups of Wheeled Mobile Robots With Limited Communication", *IEEE Trans. on Robotics*, Vol. 26, No. 6 (2010), pp. 1099–1104.
- (26) Kato, T., Watanabe, K., Maeyama, S., and Habib, M. K., "A Forming Algorithm and its Position Estimation for Triangle-based Robot Formation", *Int. J. Mechatronics and Manufacturing Systems*.
- (27) 村田智, 広瀬武志, "差動オドメタによる無人搬送車のデッドレコニング", 機械技術研究所所報, Vol. 47, No. 1 (1993), pp. 1-9.
- (28) Barshan, B., "Inertial Navigation Systems for Mobile Robots", *IEEE Trans. on Robotics and Automation*, Vol. 11, No. 3 (1995), pp. 382–342.
- (29) Hutchinson, S., Hager, G., and Corke, P., "A Tutorial on Visual Servo Control", *IEEE Trans. on Robotics and Automation*, Vol. 12, No. 5 (1996), pp. 651–670.
- (30) Zergeroglu, E., Dawson, D. M., de Queiroz, M. S., and Setlur, P., "Robust Visual-Servo Control of Robot Manipulators in the Presence of Uncertainty", *J. of Robotic Systems*, Vol. 20, No. 2 (2003), pp. 93–106.
- (31) Bajracharya, M., DiCicco, M., Backes, P., and Nickels, K., "Visual End-Effector Position Error Compensation for Planetary Robotics", *J. of Field Robotics*, Vol. 24, No. 5 (2007), pp. 399–420.

- (32) 菅野道夫,ファジィ制御,初版(1988),pp. 1-4,日刊工業新聞社.
- (33) Zadeh, L. A., "Fuzzy Algorithms", *Information and Control*, Vol. 12 (1968), pp. 94–102.
- (34) Mamdani, E.H., "Advances in the Linguistic Synthesis of Fuzzy Controllers", *Int. J. Man-Machine Studies*, Vol. 8, No. 6 (1976), pp. 669–678.
- (35) Chaumette, F. and Hutchinson, S., "Visual Servo Control, Part I: Basic Approaches", *IEEE Robotics and Automation Magazine*, Vol. 13, No. 4 (2006), pp. 82–90.
- (36) Chaumette, F. and Hutchinson, S., "Visual Servo Control, Part II: Advanced Approaches", *IEEE Robotics and Automation Magazine*, Vol. 14, No. 1 (2007), pp. 109–118.
- (37) Thuilot, B., Martinet, P., Cordesses, L., and Gallice, J., "Position Based Visual Servoing: Keeping the Object in the Field of Vision", *in Proc. IEEE Int. Conf. Robotics and Automation*, (2002), pp. 1624–1629.
- (38) 小川陽子,白井良明,島田伸敬,"カメラ位置姿勢の高速な推定のためのランドマーク選択法と単眼視によるロボットのナビゲーション",日本ロボット学会誌, Vol. 29, No. 9 (2011), pp. 811–820.
- (39) Bajracharya, M., DiCicco, M., Backes, P., and Nickels, K., "Visual End-Effector Position Error Compensation for Planetary Robotics", *J. of Field Robotics*, Vol. 24, No. 5 (2007), pp. 399–420.
- (40) Ha, I.-J., Park, D.-H., and Kwon, J.-H., "A Novel Position-based Visual Servoing Approach for Robust Global Stability with Feature Points Kept within the Field-of-view", *Int. Conf. Control Automation Robotics and Vision*, (2010), pp. 1458–1465.
- (41) Cherubini, A., Chaumette, F., and Oriolo, G., "A Position-based Visual Servoing Scheme for Following Path with Nonholonomic Mobile Robots," *IEEE/RSJ Int. Conf. on Intelligent Robotics and Systems*, (2008), pp. 1648–1654.
- (42) Wang, H., Liu, Y.-H., and Chen, W., "Vision-based Robotic Tracking of Moving Object with Dynamic Uncertainty", *IEEE/RSJ Int. Conf. on Intelligent Robotics and Systems*, (2010), pp. 2896–2901.
- (43) Tarokh, M. and Merloti, P., "Vision-based Robotic Person Following under Light Variations and Difficult Walking Maneuvers, *J. of Field Robotics*, Vol. 27, No. 4 (2010), pp. 387–398.

- (44) Mariottini, G. L., Oriolo, G., and Prattichizzo, D., "Image-based Visual Servoing for Nonholonomic Mobile Robots Using Epipolar Geometry", *IEEE Trans. on Robotics*, Vol. 23, No. 1 (2007), pp. 87–100.
- (45) Malis, E., Mezouar, Y., and Rivers, P., "Robustness of Image-based Visual Servoing with a Calibrated Camera in the Presence of Uncertainties in the Three-dimentional Structure", *IEEE Trans. on Robotics*, Vol. 26, No. 1 (2010), pp. 112–120.
- (46) Espiau, B., "Effect of Camera Calibration Errors on Visual Servoing in Robotics", *Int. Symp. Exp. Robot.*, (1993), pp. 182–192.
- (47) Espiau, B., Chaumette, F., and Rives, P., "A New Approach to Visual Servoing in Robotics", *IEEE Trans. on Robotics and Automation*, Vol. 8, No. 3 (1992), pp. 313–326.
- (48) 倉舗圭太,深尾隆則,"非ホロノミック移動ロボットの画像に基づくロバストな任意曲線追従制御",計測自動制御学会論文集, Vol. 47, No. 5 (2011), pp. 238-246.
- (49) Luca, A. D., Oriolo, G., and Giordano, P. R., "Feature Depth Observation for Image-based Visual Servoing: Theory and Experiments", *Int. J. of Robotics Research*, Vol. 27, No. 10 (2008), pp. 1093–1116.
- (50) Holmblad, L.P. and Ostergaard, J.-J., "Control of a Cement Kiln by Fuzzy Logic", *Fuzzy Information and Decision Processes*, (1982), pp. 389–399.
- (51) Sugeno, M. and Nishida, M., "Fuzzy Control of a Model Car", *Fuzzy Sets and Systems*, Vol. 16, No. 2 (1985), pp. 103–113.
- (52) Tsai, P.-S., Wang, L.-S., and Chang, F.-R., "Modeling and Hierarchical Tracking Control of Tri-Wheeled Mobile Robots", *IEEE Trans. on Robotics*, Vol. 22, No. 5 (2006), pp. 1055–1062.
- (53) Bezine, H., Derbel, N., and Alimi, A. M., "Fuzzy Control of Robot Manipulators:: Some Issues on Design and Rule Base Size Reduction", *Engineering Applications of Artificial Intelligence*, Vol. 15, No. 5 (2002), pp. 401–416.
- (54) Seraji, H., "Fuzzy Traversability Index: A New Concept for Terrain-Based Navigation", *J. of Robotic Systems*, Vol. 17, No. 2 (2000), pp. 75–91.
- (55) Alasty, A. and Sepehri, A., "Fuzzy Control of Robot Manipulator with a Flexible Tool", *J. of Robotic Systems*, Vol. 22, No. 7 (2005), pp. 367–382.

- (56) Burn, K., Short, M., and Bicker, R., "Adaptive and Nonlinear Fuzzy Force Control Techniques Applied to Robots Operating in Uncertain Environments", *J. of Robotic Systems*, Vol. 20, No. 7 (2003), pp. 391–400.
- (57) Chen, G.-M., Lin, P.-Z., Wang, W.-Y., Lee, T.-T., and Wang, C.-H., "Image-based Fuzzy Control System", *Electronics Letters*, Vol. 44, No. 7 (2008), pp. 461–462.
- (58) Agin, G. J., "Real Time Control of a Robot with a Mobile Camera", Technical Note 179, SRI International (Feb. 1979).
- (59) Weiss, L. E., Sanderson, A. C., and Neuman, C. P., "Dynamic Sensor-Based Control of Robots with Visual Feedback", *IEEE J. of Robotics and Automation*, Vol. 3, No. 5 (1987), pp. 404–417.
- (60) 美多勉, 非線形制御入門, 初版 (2000), 昭晃堂.
- (61) 徐剛, 辻三郎, 3次元ビジョン, 初版(1998), pp. 7-11, 共立出版株式会社.
- (62) Slotine, J.-J.E., and Li, W., "Applied Nonlinear Control", Prentice Hall (1991), pp. 122–126.
- (63) Duda, R. O. and Hart, P. E., "Use of the Hough Transformation to Detect Lines and Curves in Pictures", *Graphics and Image Processing*, Vol. 15, No. 1 (1972), pp. 11–15.
- (64) Holland, J. H., "Adaptation in Natural and Atrificial Systems", (1975), The University of Micigan Press.
- (65) Homaifar, A. and McCormick, E., "Simultaneous Design of Membership Functions and Rule Sets for Fuzzy Controllers Using Genetic Algorithms", *IEEE Trans. on Fuzzy Systems*, Vol. 3, No. 2 (1995), pp. 129–139.
- (66) Wong, C.-C., Wang, H.-Y., Li, S.-A., and Cheng, C.-T., "Fuzzy Controller Designed by GA for Two-wheeled Mobile Robots", *Int. J. of Fuzzy Systems*, Vol. 9, No. 1 (2007), pp. 22–30.
- (67) Press, W. H., Teukolsky, S. A., Vetterling, W. T., Flannery, B. P., (丹慶勝市,奥村晴彦,佐藤俊郎,小林誠訳),ニューメリカルレシピ・イン・シー,初版(1993), pp. 483–492,技術評論社.
- (68) 城殿清澄, 三浦純, 白井良明, "誘導による移動経験に基づく視覚移動ロボットの自律走行", 日本ロボット学会誌, Vol. 19, No. 8 (2001), pp. 1003-1009.

- (69) Blackmore, L., Ono, M., and Williams, B. C., "Chance-Constrained Optimal Path Planning With Obstacles", *IEEE Trans. on Robotics*, Vol. 27, No. 6 (2011), pp. 1080–1094.
- (70) Courbon, J., Mezouar, Y., and Martinet, P., "Autonomous Navigation of Vehicles from a Visual Memory using a Generic Camera Model", *IEEE Trans. on Intelligent Transportation Systems*, Vol. 10, No. 3 (2009), pp. 392–402.
- (71) Huang, W. H., Fajen, B. R., Fink, J. R., and Warren, W. H., "Visual Navigation and Obstacle Avoidance using a Steering Potential Function", *Robotics and Autonomous Systems*, Vol. 54, No. 4 (2006), pp. 288–299.
- (72) Lee, D., Lim, H., and Kim, H. J., "Obstacle Avoidance using Image-based Visual Servoing Integrated with Nonliniear Model Predictive Control", *IEEE Conf. on Decision and Control and European Control Conference*, (2011), pp. 5689–5694.
- (73) Cherubini, A. and Chaumette, F., "Visual Navigation with Obstacle Avoidance", *IEEE/RSJ Int. Conf. on Intelligent Robotics and Systems*, (2011), pp. 1593–1598.
- (74) 久村富持, 状態関数と線形制御系, 初版 (1970), pp. 153-172, 学献社.

本論文に関する発表論文

発表論文

- 1. Tatsuya Kato, Keigo Watanabe, and Shoichi Maeyama, "Image-based Fuzzy Trajectory Tracking Control for Four-wheel Steered Mobile Robots", *Artificial Life and Robotics*, Vol. 17, No. 1 (2012), pp. 130–135.
- 2. Tatsuya Kato, Keigo Watanabe, Shoichi Maeyama, and Maki K. Habib, "A Forming Algorithm and its Position Estimation for Triangle-based Robot Formation", *Int. J. Mechatronics and Manufacturing Systems*. (2013 年内に掲載予定)
- 3. 加藤達也,渡辺桂吾,前山祥一,"任意軌道追従のための画像に基づくファジィ制御器のGAによる設計",日本機械学会論文集(C1編)(投稿中)

国際会議

- 1. Tatsuya Kato, Shoichi Maeyama, and Keigo Watanabe, "A Formation Method for Heterogeneous Multiple Robots by Specifying the Relative Position of Each Robot", *Annual Conf. of Society of Instrument and Control Engineers*, (2010), pp. 3274–3277.
- 2. Tatsuya Kato, Keigo Watanabe, and Shoichi Maeyama, "An Image-Based Method for Controlling the Trajectory Tracking of Mobile Robots", *Int. Conf. on Soft Computing and Intelligent Systems and Int. Symp. on Advanced Intelligent Systems*, (2010), pp. 1257–1260.
- 3. Tatsuya Kato, Keigo Watanabe, and Shoichi Maeyama, "Trajectory Tracking Control for Nonholonomic Mobile Robots by an Image-Based Approach", *Int. Symp. on Artificial Life and Robotics*, (2011), pp. 403–406.
- 4. Tatsuya Kato, Keigo Watanabe, and Shoichi Maeyama, "Trajectory Tracking Control of Mobile Robots by an Image-based Fuzzy Controller", *Annual Conf. of Society of Instrument and Control Engineers*, (2011), pp. 1309–1313.
- 5. Tatsuya Kato, Keigo Watanabe, and Shoichi Maeyama, "Image-Based Trajectory Tracking with Fuzzy Control for Nonholonomic Mobile Robots", *Annual Conf. of IEEE Industrial Electronics Society*, (2011), pp. 3180–3185.

- 6. Tatsuya Kato, Keigo Watanabe, and Shoichi Maeyama, "Image-based Fuzzy Trajectory Tracking Control for Four-Wheel Steered Mobile Robots", *Int. Symp. on Artificial Life and Robotics*, (2012), pp. 180-183.
- 7. Tatsuya Kato, Keigo Watanabe, and Shoichi Maeyama, "Trajectory Tracking of Four-wheel Steered Mobile Robots by an Image-based Fuzzy Controller", *IEEE Int. Symp. on Industrial Electronics*, (2012), pp. 1546–1550.
- 8. Tatsuya Kato, Keigo Watanabe, and Shoichi Maeyama, "Arbitrary Trajectory Tracking of a Mobile Robot by an Image-based Fuzzy Controller", *Annual Conf. of Society of Instrument and Control Engineers*, (2012), pp. 160–165.

国内口頭発表

- 1. 加藤達也,渡辺桂吾,前山祥一,"画像に基づく制御による非ホロノミック 移動ロボットのためのファジィ軌道追従法",日本機械学会ロボティクス・メ カトロニクス講演会'11,(2011),1P1-L13.
- 2. 加藤達也,渡辺桂吾,前山祥一,"移動ロボットのための画像に基づくファジィ軌道追従制御",日本ロボット学会学術講演会予稿集,(2011),2I2-5.
- 3. 加藤達也,渡辺桂吾,前山祥一,"任意軌道追従のための画像に基づくファジィ制御器の GA による最適化",計測自動制御学会コンピューテーショナル・インテリジェンス研究会,(2012),pp. 131–134.