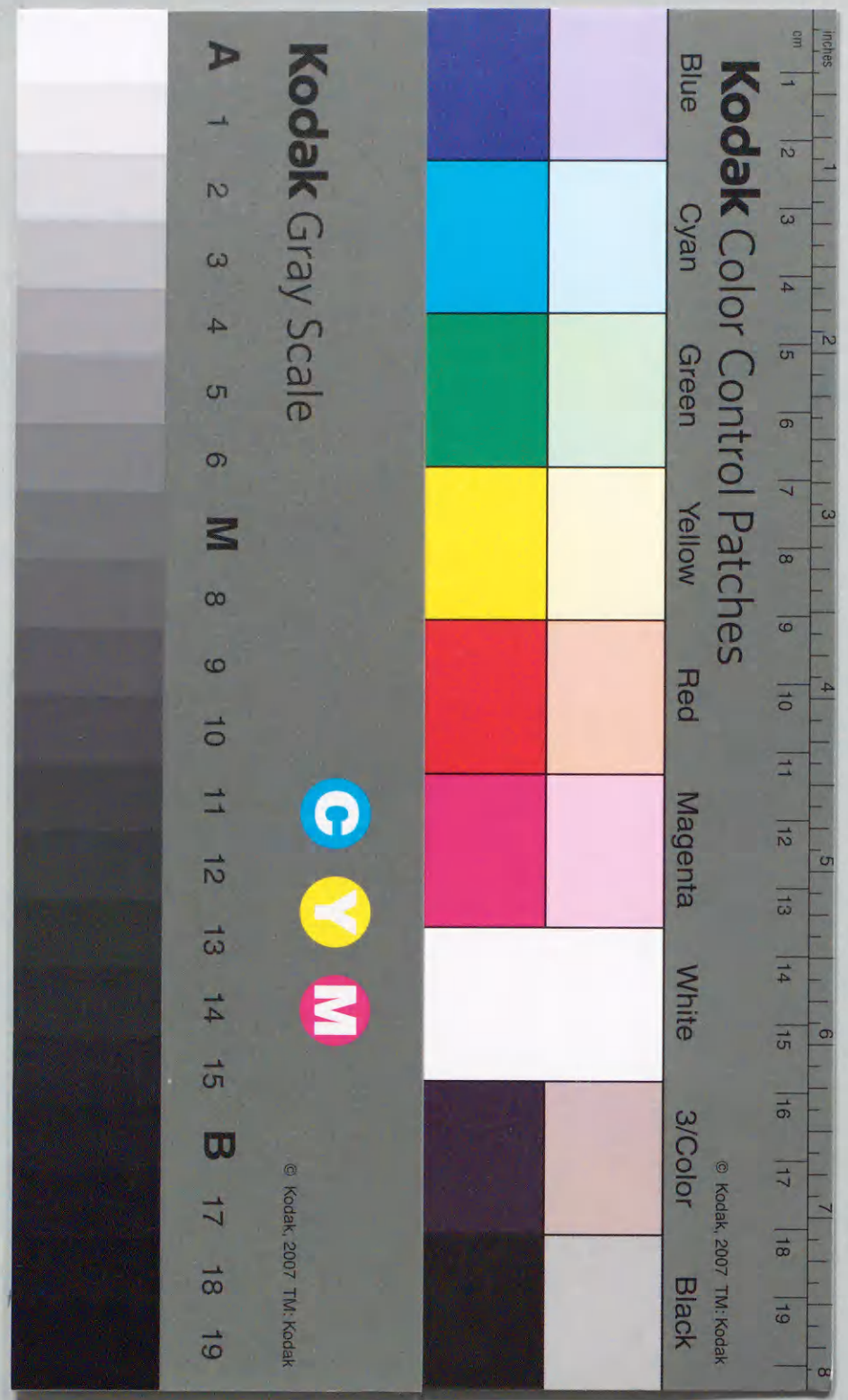


自律移動車用空間把握と誘導の
アルゴリズムに関する研究

小野口 一則



目次

1 序論	3
1.1 研究の背景	3
1.2 従来手法	5
1.2.1 空間把握アルゴリズム	5
1.2.2 誘導アルゴリズム	9
1.3 本研究の目的	13
1.4 本研究の概括	15
2 ステレオ視による移動可能空間検出アルゴリズム	19
2.1 従来手法	19
2.2 平面投影ステレオ法	21
2.2.1 平面の設定	22
2.2.2 ステレオ画像の投影	23
2.2.3 平面投影差分の作成	26
2.2.4 道路領域の抽出	31
2.3 実験結果	36
3 ステレオ視による移動障害物位置検出アルゴリズム	47
3.1 従来手法	47
3.2 移動障害物位置検出法	49
3.2.1 移動可能空間マップの作成	49
3.2.2 移動障害物領域の抽出	50
3.2.3 移動障害物領域の投影	54
3.2.4 移動障害物位置算出	54
3.3 実験結果	57

4 手続き型地図による誘導アルゴリズム	66
4.1 従来の手法	66
4.2 手続き型地図	68
4.3 手続き型地図の作成手法	72
4.3.1 位置確認地点の設定	72
4.3.2 自己位置計測手順の生成	72
4.3.3 位置確認地点間の走行手法の設定	78
4.4 実験結果	81
4.4.1 位置確認地点における自己位置計測精度	81
4.4.2 走行実験結果	84
5 行動指向型誘導アルゴリズム	89
5.1 従来の手法	89
5.2 設計の基本方針	91
5.3 各エージェントの詳細	95
5.3.1 反射行動エージェント群	95
5.3.2 目的達成行動エージェント群	95
5.3.3 適応行動エージェント群	104
5.4 行動の選択	110
5.5 実験結果	112
6 結論	121
謝辞	124
参考文献	125
著者論文リスト	134

1 序論

1.1 研究の背景

定型作業しか行なえない従来の産業用ロボットに対し、病院内搬送や警備等のための屋内移動ロボットのニーズが近年高まっている。また、車両や道路インフラを高度化し、車両運行の安全性・効率性を高める ITS(Intelligent Transport Systems) 技術の一環として、自動車の衝突回避のための障害物検出技術や運転支援技術の実用化が強く期待されている。

これら移動ロボットや自律移動車に関する研究は、1960年代より人工知能(AI)研究の一例として始められた。初期の代表例としては、SRI(Stanford Research Institute)が1960年代後半に開発した Shakey が挙げられる [1]。Shakey はレンジセンサと TV カメラからの情報を用いて環境の地図を作成するが、計算機パワーの不足により単純な障害物を回避するのに1日の時間を要した。

1970年代後半には、Stanford Cart やその後継機種である CMU Rover が開発された [2]。これらは、interest operator (全ての方向において輝度の分散値が高い画像中の点を検出するオペレータ) を用いて画像中の特徴点(角点)を検出し、スライド式カメラによるステレオ視で特徴点までの距離を計測する。ロボットの周囲に存在する対象物までの距離を全て獲得した後、ロボットが移動できる場所を検出し、移動する。この手順を繰り返すことにより、環境の地図を3次元格子モデルの形で表現する。この当時も計算機パワーがまだ不足していたため、次の移動場所を考えるのに、10~15分間停止している必要があり、20 m 移動するのに約5時間を要した。

これらの研究は、周囲の環境を観測し、環境のモデル(地図)を自動生成することが主題であり、1980年代以降も様々な研究成果 [3][4][5] が報告されている。しかし、環境を学習するというアプローチは、AI的な興味としては面白いが、目的地へロボットを誘導する等の実応用を考えると、計算コストが高くなり過ぎるという問題がある。このため、1980年代に入ると、実環境における移動ロボットや自律移動車の誘導を主眼とした研究

表 1: 自律移動ロボットと自律移動車の要求性能

	主な動作環境	観測範囲	アクチュエータ	走行の円滑性	障害物への対処
自律移動ロボット	屋内	狭い	小回りが利く	軽視	障害物検知回避レベル
自律移動車	屋外	広い	小回りが利かない	重視	移動可能空間検出

が行われ始めた。一つは、屋外の道路を高速に走行する自律移動車に関する研究であり、一つは、動的に変化する屋内環境を移動する移動ロボットに関する研究である。特に、自律移動車に関する研究は、米国 DARPA プロジェクトの一つに取り上げられたり、次世代の先進安全車 ASV (Advanced Safety Vehicle) の開発を目指す自動車メーカー各社のニーズにより、多くの研究機関で進められた。

表 1 を用いて、自律移動車と自律移動ロボット間の要求性能の違いを説明する。自律移動車は、主として屋外の道路を走行するため、広い範囲を観測する必要がある。また、小回りが利かないため、走行経路を遠方まできちんと策定する必要があり、乗客の安全性を考えると急減速、急ハンドルは避けなければならないため、誘導に円滑性が求められる。

これに対し、自律移動ロボットは、主として屋内環境を移動するため、見通しが利かず狭い範囲しか観測できない場合が多い。しかし、狭い環境で移動するため、その場回転(ピボットターン)ができる等小回りの利くアクチュエータを用いる場合が多く、停止・回転・前進・後進を組み合わせた円滑性を欠く移動も許容される。このため、障害物を回避するための経路を遠方まできちんと策定しなくても、近傍の障害物検出と回避行動を繰り返すことにより、目的地へ誘導することが可能である。

自律移動車及び自律移動ロボットを安全に誘導するためには、障害物等に衝突しないよう周囲の状況を把握するための空間把握アルゴリズムと目的地へ接近させるための誘導アルゴリズムが必要である。空間把握アルゴリズムは、広範囲な空間において密な距離

表 2: 空間把握アルゴリズムにおける移動ロボットと自律移動車の分類

	障害物検出・回避		移動可能空間検出 (距離マップ作成)	
	移動ロボット	自律移動車	移動ロボット	自律移動車
TVカメラ	原子力発電所 保守点検ロボット (東芝、他) 晴信 (山梨大)	VaMoRs (BW Munich Univ.) 知能自動車 (EIL) PVS (富士通&日産) ASV (ダイハブ)	多重視覚 エージェントロボット (大阪大) 全方位視覚ロボット (大阪大) Stanford Cart (Stanford Univ.) VISOCAR (Ruhr-Universität)	ASV (スバル)
アクティブレンジセンサ (超音波、レーザ、ミリ波)	AGV 山産 (筑波大)		ARK (Tront Univ.)	NAVLAB (Carnegie Mellon Univ.) ASV (トヨタ、三菱、 マツダ、ホンダ、 日産、メルセデス)

マップを作成する移動可能空間検出アルゴリズムと、障害物上の一部の距離情報だけ獲得する障害物検出・回避アルゴリズムに大別される。また、誘導アルゴリズムは走行経路から逸脱しないよう進行方向に対し横方向の位置を制御する走行路維持制御アルゴリズムと走行経路上の自己位置を同定し、目的地まで接近していくための位置確認アルゴリズムに分けられる。外界を観測するためのセンサとしては、TVカメラまたは超音波、レーザ、ミリ波等を照射するアクティブレンジセンサが通常用いられる。

1.2 従来の手法

1.2.1 空間把握アルゴリズム

これまで開発されてきた代表的な自律移動車及び自律移動ロボットを空間把握アルゴリズムの点で分類した表 2 を用い、従来手法の説明を行う。

障害物検出・回避

障害物検出・回避アルゴリズムでは、障害物上の一部の距離しか得られないため、障害物を回避するための経路を一度の計測で策定することはできない。このため、障害物が検知されたら停止するか、周囲を再探索して空いている空間を検出し、移動するという動作

の繰り返しで障害物を回避する。

自律移動ロボットの場合、前進・後進・転回等の制御が比較的容易であり、ある程度円滑さを欠く移動も容認されるため、障害物検出・回避アルゴリズムを用いることができる。

TVカメラにより障害物までの距離情報を得るには、複数台のカメラで得た画像、あるいは、視点位置を変えて入力した複数枚の画像に対し、ステレオ視を用いるのが一般的である。東芝等が開発した原子力発電所保守点検ロボット [6] は、ステレオカメラを用いて障害物上の縦エッジまでの距離を計測し、障害物回避を行っている。山梨大が開発した自律移動ロボット晴信 [7] は、画像中で道路の色と異なる領域を障害物の候補として抽出する。抽出された候補には、道路上の影や模様領域も含まれるため、頭部と胸部に設置された2台のTVカメラによるステレオ視で障害物候補の高さをそれぞれ計測し、道路面と異なる高さを持つ障害物候補のみを真の障害物として選択する。

アクティブレンジセンサとしては、超音波センサが良く用いられる。これは、屋内環境では遠距離計測が必要無く、コストや安全面を考えるとレーザやミリ波より有利なためである。工場等で用いられる無人搬送車 (AGV:Automatic Guidance Vehicle) は、車体の前後左右に設置された超音波センサで障害物の存在を検知し、停止する。筑波大が開発した自律移動ロボット山彦 [8] は、超音波センサにより通路の壁面等の障害物を検知している。

障害物検出・回避アルゴリズムでは、滑らかな障害物回避行動を実現することは困難である。このため、滑らかな移動が要求される自律移動車に対しては、障害物が検知されたら停止する程度の制御しか実現できない。

TVカメラをセンサとして用いた最も初期の例としては、電子技術総合研究所の知能自動車が挙げられる [9]。上下に配置された2台のTVカメラを普通乗用車の前面に設置し、水平エッジをステレオ対応付けすることにより障害物検出を行った。専用の画像処理ハードウェアを搭載することにより時速30km/hで走行したが、走行できる環境はテストコースのような整備された道路に限定されていた。BW Munich Universityで開発された自律移動車 VaMoRs [10] は、モーションステレオの原理を用い、1台のTVカメラで障害物までの距離を算出する。障害物上の複数の特徴点を前進しながら撮影した画像中で追跡し、特徴点間の長さの変化から障害物までの距離を求める。時速30km/hで走行しながら障害

物を検出する実験を行っている。富士通と日産により開発された自律移動車 PVS(Personal Vehicle System) [11] は、知能自動車と同様な縦型ステレオカメラにより、障害物を検出している。

超音波センサは、高速走行時や遠距離の計測が困難なため、自律移動車の障害物検出には不向きである。このため、レーザやミリ波をスキャンするアクティブレンジセンサが通常用いられるが、これらのセンサは広範囲な距離マップを直接得ることができるため、移動可能空間検出アルゴリズムの範疇に含まれる。このため、障害物検出・回避アルゴリズムのみをアクティブレンジセンサにより実現した自律移動車の例は無い。

移動可能空間検出

移動可能空間検出アルゴリズムは、広範囲で密な距離マップを獲得することが目的である。このため、遠方まで考慮した滑らかな走行経路が策定でき、自律移動車の障害物回避に適している。

TVカメラを用いて広範囲な距離情報を獲得する手法にはステレオ視があるが、屋外等複雑なシーンでは画像間の対応付けが難しく、かつ計算コストも高いため、密な距離マップを高速に作成することはこれまで困難であった。このため、移動速度が遅く、停止しながら計測することも許される屋内移動ロボットには、TVカメラをセンサとして用いたものもあるが、屋外の高速走行を前提とした自律移動車には、レーザやミリ波等のアクティブレンジセンサが通常用いられている。

TVカメラにより移動可能空間を検出している屋内移動ロボットとしては、前述した Stanford Cart に加え、大阪大の多重視覚エージェントロボット [5]、全方位視覚ロボット [12]、及び Ruhr University の VISOCAR [13] 等がある。

多重視覚エージェントロボットは、屋内環境には垂直エッジが多数存在することに着目し、画像中で注視した点を中心に垂直エッジの位置を計測することで、移動環境の粗い地図を作成している。搭載しているTVカメラは1台であるが、床面と垂直な軸を中心に回転させることで距離情報を得ている。

全方位視覚ロボットは、双曲面ミラーを用いた全方位視覚センサ COPIS により移動し

ながら獲得した複数枚の全方位画像から、環境内の垂直エッジの位置を計測し、移動環境の地図を作成している。

工場用の無人搬送車 AGV にステレオカメラを搭載した移動ロボット VISOCAR は、ステレオ画像を床面上に逆投影し、床面と高さが異なる対象物の領域を障害物領域として抽出している。ステレオ画像間の対応探索が不要なため、距離マップを高速に獲得することができるが、床面の微妙な凹凸や設定した平面及びカメラパラメータの誤差等により発生する誤抽出領域を除去するための有効な手段を持たないため、走行中に車体が振動する自律移動車には適用できない。

アクティブレンジセンサにより移動可能空間を検出する移動ロボットの例は少ない。これは、狭い屋内環境において、レーザは安全面に不安があり、ミリ波は相互反射により正確な距離を得ることが難しいからである。Tront Univ. で開発された移動ロボット ARK[14] は、TV カメラとレーザレンジファインダを併用し移動可能空間を求めている。色情報を用いて画像を分割し、各分割領域までの距離をレーザレンジファインダで計測し、距離マップを求めている。

自律移動車の大多数は、レーザやミリ波等のアクティブレンジセンサにより、前方の移動可能空間を検出している。初期の例としては、Carnegie Mellon Univ. が米国の DARPA プロジェクトの一部として 1985 年より開発した自律移動車 Navlab (Navigation Laboratory) がある [15]。ERIM センサと呼ばれるレーザレンジファインダにより前方の距離マップを作成している。近年、自動車メーカー各社が開発を進めている先進安全車 ASV にもアクティブレンジセンサによる移動可能空間検出アルゴリズムが搭載され始めた [16]。トヨタ、三菱はレーザレーダーにより前方の距離マップを獲得する障害物検出システムを製品化している。マツダ、ホンダも同様のシステムを搭載した ASV を開発している。日産及びメルセデスベンツは、ミリ波レーダーによる障害物検出システムを製品化している。スバルのみが、TV カメラを用いた移動可能検出アルゴリズムを搭載した ASV を開発している [17]。エッジ点を高速に相関マッチングするハードウェアを開発し、ステレオ視により走行路に進入してくる他車や歩行者の存在位置を求めているが、2 枚の画像間の対応点探索に多大な計算量が必要なため、ハードウェアコストを下げるのが難しい。

表 3: 誘導アルゴリズムにおける移動ロボットと自律移動車の分類

	走行路維持制御		位置確認	
	移動ロボット	自律移動車	移動ロボット	自律移動車
TVカメラ	原子力発電所 保守点検ロボット (東芝、他) 多重視覚 エージェントロボット (大阪大) 晴信 (山梨大)	VaMoRs (BW Munich Univ.) Alvin (Maryland Univ. 他) PVS (富士通&日産) NAVLAB (Carnegie Mellon Univ.) ASV (トヨタ、三菱、マツダ、 ホンダ、日産) ASV (スバル)	View-based Robot (東京大、大阪大、 筑波大) 全方位視覚ロボット (大阪大、 New York Univ.) クローラ型ロボット (日立)	NAVLAB (Carnegie Mellon Univ.)
アクティブレンジセンサ (超音波、レーザ、ミリ波)	山彦 (筑波大) 晴信 (山梨大)		山彦 (筑波大)	

1.2.2 誘導アルゴリズム

誘導アルゴリズムは走行経路から逸脱しないよう進行方向に対し横方向の位置を制御する走行路維持制御アルゴリズムと走行経路上の自己位置を同定し目的地まで接近していくための位置確認アルゴリズムに分けられる。センサとしては、TV カメラを用いる場合が圧倒的に多く、アクティブレンジセンサは移動ロボットの誘導に超音波センサを用いる例が少数あるにすぎない。これは、白線等の走行レーンやランドマークをアクティブレンジセンサで検出することが困難なためであり、壁面等で囲まれた走行環境を移動するロボットにのみ超音波センサを用いた誘導アルゴリズムが開発されている。これまで開発されてきた代表的な自律移動車及び自律移動ロボットを誘導アルゴリズムの点で分類した表 3 を用い、従来手法の説明を行う。

走行路維持制御

道路や通路から逸脱しないよう、進行方向に対して横方向の位置を制御するアルゴリズムであり、道路端の白線や通路と壁面の境界線等を検出し、道路端や通路端に対する自車の方向と位置を算出する。

自律移動車は、TVカメラにより白線等の走行レーンや道路端等のエッジを検出している。BW Munich University で開発された自律移動車 VaMoRs[18] は、道路端の白線を画像中で高速に追跡することが可能な並列型の画像処理システムにより、建設中の高速道路を最高時速 100km/h で走行できる走行路維持制御を実現した。

米国での DARPA プロジェクトに関連し、Martin Marietta Denver Aerospace と Maryland University が開発した自律移動車 Alvin[19] は、パン・チルトが制御可能な TV カメラとレーザレンジファインダを備え、4.5km の舗装道路を時速 10km/h 以上の速度で走行した。Maryland University により開発された画像処理システムは、移動開始時 (Bootstrap stage) に画像全体でエッジを抽出する。そして、道路両端のエッジは 3 次元空間において平行であるという条件を用い、道路端のエッジ対を求める。エッジ対が得られたならば、エッジ対の各々に小さな矩形ウィンドウを配置する。移動時 (Feed-forward stage) は、画像全体ではなく、この矩形ウィンドウ内のみで道路端エッジを検出し、処理の高速化を図っている。矩形ウィンドウを配置する位置は、一時点前に入力した画像中で検出した道路端エッジの位置から予測する。

富士通と日産により開発された自律移動車 PVS(Personal Vehicle System)[11] は、汎用の高速画像処理装置 FIVS/VIP を用い、画像中で白線を高速に追跡しながら走行路維持制御するシステムを備えており、テストコースにおいて時速 60km/h (直線路)、15km/h (カーブ)、5km/h (交差点) で走行するデモンストレーション実験を行っている。

Carnegie Mellon University の Navlab[15] は、色とテクスチャーの情報を用いて前方の道路領域を検出し、道路を逸脱しないよう走行路維持制御を行っている。ニューラルネットワークを用い、入力画像から直接車のステアリング角度を出力するシステムも搭載している。

近年では、自動車メーカー各社が開発している ASV (Advanced Safety Vehicle) の車線逸脱時警報システムに走行路維持制御アルゴリズムが適用されている [16]。このシステムは、走行レーンを示す白線の位置を専用ハードウェアにより画像中で高速に検出し、車線逸脱の危険性を判定する。トヨタ [20] は、入力画像中の特徴点と走行レーンのテンプレートを高速にマッチングするハードウェアを用い、天候の変化に強い白線検出を実現してい

る。日産 [21] は、リアルタイムでハフ変換を行うハードウェアを開発し、ロバストな白線検出を実現している。三菱 [16] は、白線は道路よりも明るく画像の決まった位置に存在するという仮定を用い、水平方向の輝度プロファイルの変化から白線位置を求めている。マツダ [22] も、白線は道路より明るいという仮定に基づき、予め設定した閾値で画像を 2 値化し、白線を検出している。本田はエッジ検出により画像中の白線位置を求めている。

これら主要メーカーは、1 台の TV カメラによる単眼視で走行レーンを検出しているが、スバルは 2 台の TV カメラによるステレオ視を採用している [17]。コスト面やカメラキャリブレーションの手間を考えると単眼視が有利であるが、ステレオ視は道路上のエッジを、高さの情報を用いて確実に選択することができるため、別レーンを走行している大型車の側面エッジ等を誤抽出するといった単眼視特有の問題点を回避することができる。

自律移動ロボットに関しては、TV カメラや超音波センサを用いたシステムが開発されている。東芝等が開発した原子力発電所保守点検ロボット [6] は、単眼カメラにより床面と壁面との間の境界エッジを抽出する。そして、このエッジの位置及び方向から壁面に対するロボットの位置・姿勢を算出し、壁面に沿ってロボットを移動させている。

大阪大の多重視覚エージェントロボット [23] は、ロボットの前方と後方を撮影する 2 台の TV カメラの画像中でそれぞれ 1 つの特徴点を設定する。そして、これらの特徴点が画像内に収まるようロボットの向きを制御することにより、2 つの特徴点間を結ぶ直線上の誘導を実現している。

山梨大が開発したキャンパス内移動ロボット晴信 [7] は、歩道の縁石や路肩の白線等のサインパターンを色による領域分割で抽出し、サインパターンに沿った移動を実現している。

筑波大の山彦、山梨大の晴信は、超音波センサにより壁面までの距離を算出し、この距離が一定になるようステアリング制御することで、壁面や駐車車両等に沿った移動を実現している。

位置確認

目的地へ車を自律的に誘導するためには、自己の現在位置及び走行すべき方角を知る必

要がある。走行環境全体の情報を記述したデータ（地図）を予め作成しておき、走行中に入力したセンサ情報との照合で環境内の自己の現在位置を獲得する手法がいくつか提案されている。

東京大 [24]、大阪大 [25]、筑波大 [26] がそれぞれ開発した View-based Robot は、ロボットに搭載された TV カメラにより走行環境全体の画像を予め撮映しておき、入力画像が環境内のどの位置で獲得された画像に最も近いかを調べることで、環境内の自己位置を同定している。

大阪大 [12]、New York Univ.[27] がそれぞれ開発した全方位視覚ロボットは、異なる地点で獲得した 2 枚の全方位画像からロボット周囲の垂直エッジ位置を求め、予め作成しておいた環境全体の垂直エッジ位置が記述されたマップと照合することで、環境内の自己位置を求めている。

日立が開発したクローラ型ロボット [28] は、予め作成しておいた走行環境全体の 3 次元のワイヤフレームモデルと走行中に入力した画像から生成したエッジ画像を照合することで、環境内の自己位置を同定している。

筑波大の山彦 [8] は、通路の幅や長さといった走行環境の幾何情報を記述した地図と超音波センサにより求めた壁面までの距離情報を照合し、環境内の自己位置を求めている。

これらの位置確認手法は走行環境全体のデータを予め作成しておく必要があるため、屋内の移動ロボットには適用できるが、走行環境が桁外れに大きい自律移動車には不向きである。屋外環境での位置確認手法としては、Carnegie Mellon Univ. の Navlab に搭載された Annotated Map [29] を用いた手法がある。Navlab は、経路中のどこで何の処理を起動するかを記述した手順 (Annotated Map) を予め走行前に準備しておき、この手順に従い、処理プログラムを順次切り替えながら目的地へ接近していく。しかし、この Annotated Map 自体をどのように作成するかについては検討されていない。人間が制御プログラムを作成する要領でこの Map を構築すると、多数の走行経路が存在する場合、多大な時間が必要となる。

1.3 本研究の目的

以上述べたように、自律移動車及び自律移動ロボットを目的地まで安全かつ円滑に移動させるためには、空間把握のための移動可能空間検出アルゴリズムと誘導のための走行路維持制御アルゴリズム及び位置確認アルゴリズムが必要である。走行路維持制御に関しては、これまで多くのアルゴリズムが提案されており、走行レーンの白線追跡等実用化段階に近づいているものもあるが、移動可能空間検出及び位置確認に関しては有効なアルゴリズムが提案されていない。

移動可能空間検出に関しては、密な距離マップを TV カメラで得ることが困難なため、従来手法の多くがアクティブレンジセンサを用いている。このため、これまで開発されてきた自律移動車には、移動可能空間検出用のアクティブレンジセンサ（レーザ、ミリ波）と走行路維持制御及び位置確認用の TV カメラが搭載されている。TV カメラにより移動可能空間が検出できれば、高価なアクティブレンジセンサを搭載する必要が無く、コスト面で非常に有利である。また、アクティブレンジセンサには、計測できる範囲が狭い、障害物の材質や面の角度によっては十分な反射波が得られないため障害物を見落す、他車両が照射したレーザやミリ波を誤検知する等といった欠点がある。さらに、障害物の存在位置しか計測できないため、自分の動きが未知な場合、検出された障害物が人間等の移動物体なのか、あるいは駐車車両等背景に含まれる静止物体なのか判断することができない。道路等屋外環境において適用できる移動可能空間検出アルゴリズムを実現するためには、他車両のセンサの影響を受けず、広い範囲で障害物が検出できるとともに、歩行者や走行中の車等を移動障害物として切り出し、その進行方向や位置を予測することが必要である。

自律移動車が走行する環境は、屋内移動ロボットに比べて桁外れに広いため、環境全体の画像や形状データを用いる位置確認アルゴリズムを適用することは困難である。走行環境全体の画像や形状データを記述した地図の作成には多大な労力を必要とする。また、地図に蓄積されるデータ量が大きいため、地図との照合時間も長くなる。このため、位置確認に必要な要素情報のみを使い易い形で記述した地図を用い、環境内の自己位置

を同定する位置確認アルゴリズムが必要である。

本論文は、これらの問題点を解決するため、TVカメラをセンサとして用い、広範囲な空間においても密な距離マップが高速に作成できる移動可能空間検出アルゴリズム、歩行者や走行中の車等を移動障害物として切り出し、その進行方向や位置を予測する移動障害物位置検出アルゴリズム、及び屋外等走行環境が広い場合に対しても適用できる位置確認アルゴリズムを与えることを目的としている。

本論文で提案する移動可能空間検出アルゴリズムは、TVカメラをセンサとして用いるため、他車両のセンサの影響を受けず、広い範囲の障害物を一度に検出することができる。また、ステレオ視における対応探索が不要なため、計算量が少なく、安価なCPUでビデオレート処理が実現できる。ステレオ画像を道路平面に投影し、障害物を検出する手法を基本的には用いているが、この手法には従来、車体の振動により、道路面上のペイント、汚れ、影等の周辺に誤抽出領域が多数発生するという問題点があり、自律移動車の移動可能空間検出には適用できなかった。本アルゴリズムは、この問題点を解決し、車体の振動に強い移動可能空間検出を実現している。走行路維持制御や位置確認等の誘導アルゴリズムはTVカメラのみで実現できるため、一つのセンサで目的地まで自律走行するシステムが構築でき、コスト面において有利である。

本論文で提案する移動障害物位置検出アルゴリズムは、移動障害物の進行方向や位置を予測することができるため、円滑な走行が求められる自律移動車の障害物回避に有効である。フレーム間の追跡やステレオ画像間の対応探索が必要無いため、画像中で見え方やサイズが大きく変化する対象物や、形状が常に変形する非剛体に対しても、安定にその位置を求めることができる。また、高さの情報を用いて障害物領域を求めるため、道路面上に現われる移動体の影を誤って抽出することがない。このため、歩行者、自転車、車といった一般道路上に通常存在する移動障害物の位置検出に適している。

本論文で提案する位置確認アルゴリズムは、要所要所で設定した位置確認地点において予め設定しておいた目標物（ランドマーク）をTVカメラで検出することにより、走行経路上の自己位置を同定する。位置確認地点で検出すべきランドマークの種類やその抽出手順等を記述した手続き型地図を用い、自己位置を同定するため、走行環境全体の画像や

形状データ等を保持する従来手法と比べ、データ量を大幅に減らすことができ、屋外等のような広い走行環境にも適用することが可能である。病院及びオフィス内等のような環境では、人間等の移動障害物及び椅子等といった可動物体が多数存在し、位置確認のために設定したランドマークが隠されてしまう可能性が高い。本論文では、ランドマーク検出の失敗が頻発するような動的環境においても、目的地へ向かって走行できる誘導アルゴリズムも提案している。ランドマーク検出の失敗等により発生するデッドロック状態から効率良く復帰し、次のランドマーク位置まで移動することができる。

本論文で提案するアルゴリズムを統合することにより、自律移動車及び自律移動ロボットを目的地まで安全かつ円滑に移動させることができる。また、将来的には、自動車の衝突回避用安全装置や自動運転等の実現に役立つものと考えられる。

1.4 本研究の概括

本論文の概要を各章毎に述べる。

第2章では、ステレオ視による移動可能空間検出アルゴリズムを与える。屋外には樹木等複雑な形状の対象物が多数存在するため、ステレオ画像間の対応点探索に多大な計算コストが必要となる。従って、シーン全体の3次元形状を復元した後、障害物の存在しない領域を検出する手法は得策ではない。道路や屋内通路等の走行環境は、未舗装路等の例外を除き、平面の組み合わせで近似できる。また、歩行者、自転車、車等道路上に存在する大半の障害物は、道路面に接触した道路面と高さの異なる物体である。このため、ステレオ画像を道路平面上へ投影することにより、道路平面上に乗っている点か、道路平面とは高さが異なる点かを高速に判別する移動可能空間検出アルゴリズムを提案する。平面へステレオ画像を投影し、障害物を検出する手法は、これまでも提案されているが[13]、路面の微妙な凹凸や設定した平面及びカメラパラメータの誤差等により発生する誤抽出領域を除去するための有効な手段を持たなかったため、走行中に車体が振動する自律移動車には適用できなかった。本論文で提案するアルゴリズムは、設定平面上に投影されたステレオ画像間の差分画像をステレオカメラの座標原点を中心に放射状に走査し、道路面上

のペイント、汚れ、影等の周辺に発生する誤抽出領域を除去するため、車体が振動しても移動可能空間が安定に求まる。

第3章では、ステレオ視による移動障害物位置検出アルゴリズムを与える。本アルゴリズムは、歩行者や走行中の車等を移動障害物として切り出し、その進行方向や位置を予測する。オプティカルフローを用いて検出した画像中の移動障害物領域を、移動可能空間検出アルゴリズムにより求めた移動可能空間マップに投影することで、移動障害物の3次元位置を検出する。本アルゴリズムでは、フレーム間の追跡やステレオ画像間の対応探索が必要無いため、画像中で見え方やサイズが大きく変化する対象物や、形状が常に変形する非剛体に対しても、安定にその位置を求めることができる。また、高さの情報を用いて障害物領域を求めるため、道路面上に現われる移動体の影を誤って抽出することがない。このため、歩行者、自転車、車といった一般道路上に通常存在する移動障害物の位置検出に有効である。

第4章では、手続き型地図による誘導アルゴリズムを与える。本アルゴリズムは、位置確認に必要な情報を使い易い形で記述した地図を用いることにより、屋外環境にも適用することができる。人間が目的地まで道を歩く場合を考えると、代表的な建物や看板等で自己位置を推定し、曲がるべき交差点等を識別して目的地へ近付いて行く。つまり、道沿いに存在するすべての建築物等を見ながら常に自分の位置を認識し、歩いている訳ではなく、ポイントとなる目標物（ランドマーク）を用いて要所要所（曲がり角や交差点等）で現在位置及び進むべき方向の確認を行なう。そして、その間は障害物を避けながら舗道に沿って歩いたり、目標物に向かって歩いたりしていると考えられる。このため、目的地までの経路上に予めいくつかのランドマークを設定しておき、TVカメラにより獲得した画像中でこれらのランドマークを検出し、自己の現在位置や進むべき方向を求める手法が有効である。特に、高所にある看板や標識等をランドマークとして使用することが可能な道路環境、及び原子力発電所内部の通路や工場の専用レーン等のような人間の往来がほとんど無い静的環境においては、障害物による隠蔽でランドマークを見失う可能性が少ないため効率の良い移動が実現できる。本アルゴリズムは、位置確認に必要なデータ（地図）を簡便に作成する機能を有している。自律走行する前に、走行環境の簡単なレイアウトを

記述した地図から、目的地までの走行経路を設定する。そして、交差点や曲がり角といった自己位置同定が必要な位置確認地点を定める。次いで、遠隔操作で移動車や移動ロボットを位置確認地点まで誘導し、ステレオ画像を収集する。このステレオ画像を人間が対話的に処理し、位置確認に用いるランドマークの種類やその抽出手順等を記述した手続き型地図を作成する。位置確認地点でのステレオ画像が得られれば、位置確認のための処理手順を記述した手続き型地図が対話処理により簡単に作成できるため、地図の作成や走行経路の変更に伴う地図の変更が容易である。自律走行時は、各位置確認地点において手続き型地図に記述された処理手順に基づきランドマークを検出する。そして、検出されたランドマークと移動車との間の位置関係をステレオ視により求め、走行経路上の自己位置を算出する。

第5章では、行動指向型誘導アルゴリズムを提案する。病院及びオフィス内等のような環境では、人間等の移動障害物及び椅子等といった可動物体が多数存在し、位置確認のために設定した目標物（ランドマーク）が隠されてしまう可能性が高い。このような動的環境では、経路に沿ってランドマークを順次検出し、自己位置を確認する手続き型地図による誘導アルゴリズムでは対処できない場合が生じる。このため、移動車の動きを障害物回避、徘徊、目標物探索・追跡等といったいくつかの行動に分類し、これらの行動を適宜切り替えることにより、障害物を回避しつつ目的地へ接近する行動指向型誘導アルゴリズムが提案されている。MITのBrooks[30]が提案したアルゴリズムが代表的であるが、選択された行動の失敗から立ち直る部分が組み込まれていないため、デッドロックに陥る場合が生じる。例えば、障害物の出現により目標物探索・追跡が失敗した場合、障害物に衝突しないよう動き回る徘徊行動が選択されるが、同じ障害物と障害物の間を往復する等、同一経路を繰り返し動き回り、目的地へ到達できなくなるといったデッドロックに陥る可能性がある。本論文では、行動選択の失敗から発生するデッドロック状態から効果的に復帰できる機能を有した行動指向型誘導アルゴリズムを提案する。デッドロック状態を検知し、抜け出すためのヒューリスティックとして、周囲の全ての障害物から最も離れた位置へ移動する自由空間探索行動、ドアの外といった開けた空間へ移動する開空間探索行動、及び障害物に沿って移動する沿障害物走行行動を新たに基本行動として付加した。目標物

探索・追跡が失敗した場合、むやみに動き回るのではなく、できるだけ広い場所へ出る、ドアの外へ出る、壁に沿って動く等の行動を取ることで、デッドロックから効果的に抜け出すことができる。

各章とも、車載カメラで撮影した画像を用いた実験、または車輪型の移動ロボットを用いた実験により、アルゴリズムの有効性を実証している。

2 ステレオ視による移動可能空間検出アルゴリズム

TVカメラより得られた画像から周囲の距離や形状を復元する問題は、一般に、不良設定問題であり何らかの条件を与えないと解くことはできない。これまで、①複数台のTVカメラを用い三角測量の原理で解くステレオ視 [34][35]、②光源の位置を既知とし反射光から形状を求める shape from shading[36][37]、③合焦点位置の変化から距離を求める shape from focus (defocus)[38][39]、④カメラや対象物の動きから形状を復元する shape from motion[40][41] 等種々の手法が提案されている。屋外環境では、光源の位置を特定することが困難なため、手法②の適用は難しい。また、移動しながら障害物位置を検出する必要があるため、手法③も適用できない。さらに、自己の正確な動きを知ることが困難なため、手法④も移動車の空間把握には適さない。

このため、自律移動車及び移動ロボットの空間把握には、手法①のステレオ視の適用が考えられるが、従来のステレオ視は、画像間での対応点探索に多大な計算量を必要とし、移動空間の密な距離マップをリアルタイムで生成することは困難であった。本章では、移動車前方の密な距離マップを画像間の対応点探索を行わず作成する「平面投影ステレオ法」を提案し、障害物の存在しない道路領域を抽出する実験により自律移動車の移動可能空間検出に有効であることを示す。

2.1 従来の手法

ステレオ視は広範囲な空間の距離情報を2枚の画像から得ることができるため、応用範囲が広く、古くから盛んに研究されてきている。しかし、対象物上の同一点を2枚の画像中で対応付ける問題が難しく、これを解決するための様々な手法がこれまで提案されてきている。これらは、領域ベース法と特徴ベース法という2つの手法に分けることができる。

領域ベース法 [42] は、画像中で設定した小領域内で相関をとり、画像間の対応位置を探索する手法であるが、小領域内に明確な特徴が存在しない場合、誤対応が発生する。こ

のため、明確な特徴（例えば、エッジ上の点）を含む領域のみを選択し、画像間の対応付けを行う手法が良く用いられるが、領域内に距離が大きく変化する部分が存在すると、その周辺領域の見え方が2枚の画像で異なり、正確な対応位置が求まらない。このような状況は、障害物と背景との境界部に良く発生し、障害物の輪郭位置が正しく得られない原因となる。

特徴ベース法は、画像からエッジやエッジセグメントといった特徴を抽出し、特徴毎に画像間の対応位置を探索する手法であり、奥行きが不連続な箇所でも対応付けが可能である。エッジセグメントの形状やその組合せで対応付けを行う手法 [43][44] が提案されているが、これらの手法は、エッジの切れ等によるエッジセグメントの形状変化に弱い。特に、類似形状のエッジセグメントが密集している領域では、エッジの切れによる誤対応を防ぐため、多数のエッジセグメントの組合せを考慮する必要があるが、計算量が膨大になる。エッジ点毎に対応付けを行えば、エッジの切れ等の影響は少ないが、局所的な特徴であるため、対応探索の際に対応付け候補が多数出現する多対応問題が生じる。この多対応問題を回避するため、画像から抽出された垂直エッジ点を走査線毎に動的計画法を用いて対応付ける手法 [45] や、小領域内で連結しているエッジ点間の視差ヒストグラムを用いて対応付けを行う手法 [46] 等が提案されているが、これらの手法は、垂直エッジ部分の距離しか計測できなかつたり、屋外のような多数のエッジが存在する複雑な環境下では、処理量が膨大となり、環境の密な距離マップを高速に得ることは難しい。以上の様に、従来のステレオ視では対応探索に時間がかかり、前方の移動可能な空間を示す密な距離マップ（以後、移動可能空間マップと呼ぶ）を高速かつ高精度に求めることは困難であった。

道路や屋内通路等の走行環境は、未舗装路など極一部の例外を除き、平面の組合せで近似することが可能であり、障害物が存在しない移動可能な領域は、道路や通路を示す平面上に存在すると仮定できる。このため、ステレオ画像中で、平面上に乗っている点か、平面とは高さが異なる点かが判別できれば、移動車前方の移動可能な領域が求まり、移動可能空間マップを作成することができる。ステレオ画像中の各点が設定平面上の点か否かを判別するには、平面への逆透視変換投影モデルが有効であり、いくつかの手法が提案されている [13][47][48]。これらの手法は、まず、一方の画像 I_1 中の点が全て設定した

平面（例えば、道路面）上に存在すると仮定し、他方の画像 I_2 からの見え方に変換する。設定平面上に真に存在する点は、変換画像と画像 I_2 において画像中の位置が同一となるが、設定平面とは高さが異なる点は、これらの画像において異なる位置を占める。このため、変換画像と画像 I_2 との間の差分画像を求め、差の小さい領域を設定平面と同一な平面上に存在する移動可能領域として出力する。

しかし、これらの手法は路面の微妙な凹凸、及び設定した平面やカメラパラメータが持つ誤差に敏感であり、道路平面上のペイント、汚れ、影等の周辺に障害物として誤抽出される領域が多数現れるといった欠点を持っている。Storjohann[13] は画像中で面積の小さい領域を誤抽出領域として除去する処理を施しているが、判定に画像中の領域サイズを用いているため、遠くの障害物を誤抽出領域として除去したり、近くの誤抽出領域を障害物として検出したりする可能性がある。Weber[47] は画像からカメラの高さと傾きを推定することにより、設定平面とカメラとの間の誤差を減らす手法を組み込んでいるが、リアルタイムな誤差修正は難しく、誤抽出領域を完全に除去することは困難である。この様に平面への逆透視変換モデルを用いた従来手法は、路面上のテクスチャーを障害物として誤検出するため、ペイントや影が頻繁に現れる実際の道路に適用することが困難であった。

本章で提案する「平面投影ステレオ法」[50][51] は、平面への逆透視変換投影モデルを用いた場合に現れるこれらの誤抽出領域を効果的に除去することが可能である。

2.2 平面投影ステレオ法

平面投影ステレオ法は、基本的には平面への逆透視変換投影モデルを用いおり、まず、ステレオ画像を道路等の設定平面へ投影する。もし対応する点が設定平面上に真に存在するならば道路平面上の同一位置に投影される。一方、設定平面とは高さが異なる点の場合、対応する点は設定平面上の異なる位置へ投影される。このため、道路平面上に投影されたステレオ画像間の差を求める。以後、これを「平面投影差分」と呼ぶ。平面への逆透視変換投影モデルは設定平面とカメラとの間の位置関係が持つ誤差に弱いため、平面投影差分上には多数の誤抽出領域が現れる。しかし、本章で提案する平面投影ステレオ法は、ス

テレオカメラの座標原点を中心に平面投影差分を放射状に走査していく。そして、平面投影差分上の障害物領域が持つ以下の特性を用いることによりこれらの誤抽出領域を除去する。

- (1) 平面投影差分上では、障害物領域は障害物が設定平面と接地する位置から後方に放射状に倒れ込んだ形状を示す (図6参照)
- (2) 平面投影差分上では、後方に倒れ込んだ障害物領域の長さはステレオカメラに対しての障害物の位置と設定平面に対しての障害物の高さに依存する。

障害物の高さは、平面投影差分上の障害物の長さから推定できるので、障害物の高さを閾値とすることで道路平面上に現れるペイント、汚れ、影等の誤抽出領域を効果的に除去することができる。このため、道路の凹凸等によりTVカメラが振動しても、道路上の移動可能空間を確実に求めることができる。

図1に平面投影ステレオ法の処理の流れを示す。まず、移動車に搭載されたステレオカメラの位置姿勢から、ステレオカメラの座標系における道路領域の平面式を算出する。次いで、ステレオ画像中の各点をその平面上に投影する。そして、左画像から作成した投影画像と右画像から作成した投影画像との間の差を求めることにより平面投影差分を作成する。そして、この平面投影記述を移動車の位置から放射状に走査することにより道路領域を抽出する。以下に、各ステップの詳細を記述する。

2.2.1 平面の設定

道路は平面の連なりに近似できるため、カメラの傾き及び高さからステレオカメラの座標系における道路領域の平面式を算出する。設定された平面の方程式を $P: ax+by+cz+d=0$ とする。現在、カメラの傾きは固定値を用いているが、移動に伴う振動が大きい場合には、傾斜計、またはジャイロ等によりカメラの傾きを常に計測し、道路平面式を更新する。

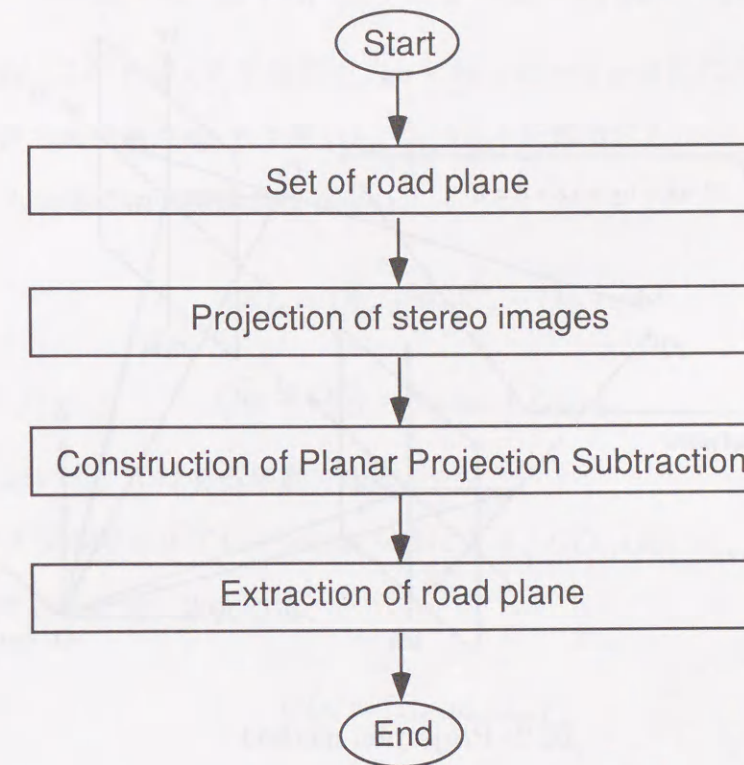


図1: Road extraction block diagram

2.2.2 ステレオ画像の投影

ステレオの左右画像を平面 P 上に投影する手法を図2を用いて説明する。ステレオカメラの座標系を $O-XYZ$ 、左右画像中心を O_l, O_r 、左右カメラの焦点位置を C_l, C_r 、左右画像面上の点 q_l, q_r を平面 P 上へ投影した点を Q_l, Q_r とする。左右画像上の画像中心を原点とする座標系をそれぞれ x_l-y_l, x_r-y_r とし、 O_l を起点とした x_l 軸方向の単位ベクトルを n_{lx} 、 O_r を起点とした x_r 軸方向の単位ベクトルを n_{rx} とする。また、 O_l を起点とした y_l 軸方向の単位ベクトルを n_{ly} 、 O_r を起点とした y_r 軸方向の単位ベクトルを n_{ry} とする。さらに、 q_l, q_r の画像面上の位置をそれぞれ $(k_{lx}, k_{ly}), (k_{rx}, k_{ry})$ とする。点 q_l, q_r を平面 P 上へ投影した点 Q_l, Q_r のステレオカメラ座標系での位置を以下の式より求める。まず、 q_l に対しては次の関係が成り立つ。

$$O\vec{Q}_l = O\vec{q}_l + t(O\vec{C}_l - O\vec{q}_l) \quad (1)$$

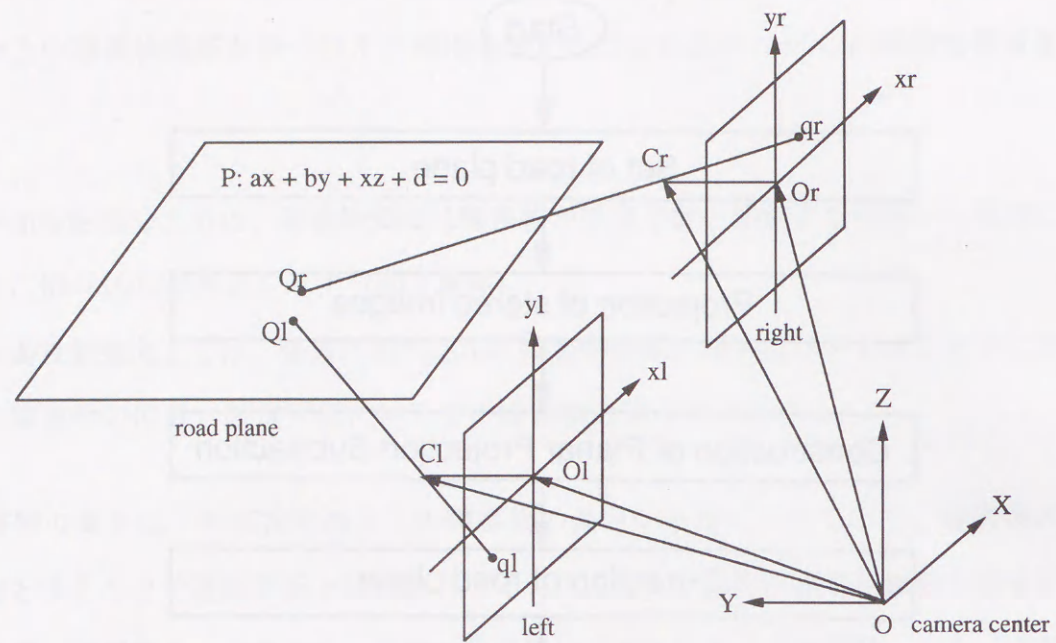


図 2: Projection method

$$\vec{O}q_l = \vec{O}O_l + k_{lx}n_{lx} + k_{ly}n_{ly} \quad (2)$$

尚、式 (1) において、 t はある定数である。

ステレオカメラのキャリブレーション [49] により、 $\vec{O}O_l, \vec{O}C_l, n_{lx}, n_{ly}$ の各要素の値は求めることができるため、本文では、

$$\vec{O}O_l = (x_{ol}, y_{ol}, z_{ol}) \quad (3)$$

$$\vec{O}C_l = (x_{cl}, y_{cl}, z_{cl}) \quad (4)$$

$$n_{lx} = (x_{nlx}, y_{nlx}, z_{nlx}) \quad (5)$$

$$n_{ly} = (x_{nly}, y_{nly}, z_{nly}) \quad (6)$$

を既知であるとする。式 (1) により、 Q_l と q_l を結ぶ直線上の点 (x, y, z) は、

$$x = x_{ol} + k_{lx}x_{nlx} + k_{ly}x_{nly} + t(x_{cl} - x_{ol} - k_{lx}x_{nlx} - k_{ly}x_{nly}) \quad (7)$$

$$y = y_{ol} + k_{lx}y_{nlx} + k_{ly}y_{nly} + t(y_{cl} - y_{ol} - k_{lx}y_{nlx} - k_{ly}y_{nly}) \quad (8)$$

$$z = z_{ol} + k_{lx}z_{nlx} + k_{ly}z_{nly} + t(z_{cl} - z_{ol} - k_{lx}z_{nlx} - k_{ly}z_{nly}) \quad (9)$$

を満足するため、この x, y, z を平面式 $P: ax + by + cz + d = 0$ に代入し、 t を求めることにより、 q_l の P 上の投影点 Q_l の位置 (Q_{lx}, Q_{ly}, Q_{lz}) を算出する。

q_r に対しても同様に次の関係が成り立つ。

$$\vec{O}Q_r = \vec{O}q_r + t(\vec{O}C_r - \vec{O}q_r) \quad (10)$$

$$\vec{O}q_r = \vec{O}O_r + k_{rx}n_{rx} + k_{ry}n_{ry} \quad (11)$$

尚、式 (10) において、 t はある定数である。

ステレオカメラのキャリブレーション [49] により、 $\vec{O}O_r, \vec{O}C_r, n_{rx}, n_{ry}$ の各要素の値は求めることができるため、本文では、

$$\vec{O}O_r = (x_{or}, y_{or}, z_{or}) \quad (12)$$

$$\vec{O}C_r = (x_{cr}, y_{cr}, z_{cr}) \quad (13)$$

$$n_{rx} = (x_{nrx}, y_{nrx}, z_{nrx}) \quad (14)$$

$$n_{ry} = (x_{nry}, y_{nry}, z_{nry}) \quad (15)$$

を既知であるとする。式 (10) により、 Q_r と q_r を結ぶ直線上の点 (x, y, z) は、

$$x = x_{or} + k_{rx}x_{nrx} + k_{ry}x_{nry} + t(x_{cr} - x_{or} - k_{rx}x_{nrx} - k_{ry}x_{nry}) \quad (16)$$

$$y = y_{or} + k_{rx}y_{nrx} + k_{ry}y_{nry} + t(y_{cr} - y_{or} - k_{rx}y_{nrx} - k_{ry}y_{nry}) \quad (17)$$

$$z = z_{or} + k_{rx}z_{nrx} + k_{ry}z_{nry} + t(z_{cr} - z_{or} - k_{rx}z_{nrx} - k_{ry}z_{nry}) \quad (18)$$

を満足するため、この x, y, z を平面式 $P: ax + by + cz + d = 0$ に代入し、 t を求めることにより、 q_r の P 上の投影点 Q_r の位置 (Q_{rx}, Q_{ry}, Q_{rz}) を算出する。

2.2.3 平面投影差分の作成

ステレオカメラの光軸が道路面とほぼ平行になるよう設定すると、道路面は画像中心位置よりも下に撮像される。このため、Y座標値が画像中心位置よりも下に位置する左画像及び右画像中の点 q_l, q_r を平面 P 上の点 Q_l, Q_r へ投影し、平面投影差分を作成する。キャリブレーションにより、ステレオカメラの座標系の $X-Y$ 平面が道路面と平行になるよう設定しておく。そして、移動車前方に設定した平面領域（道路面を含む）において、ステレオカメラの座標系に対し、 $X_{min} < X < X_{max}$ 、 $Y_{min} < Y < Y_{max}$ の矩形範囲を障害物の探索範囲とする。そして、この矩形範囲を格子状に分割し、図3に示す要素を持つ2次元配列 $A(i, j)$ ($i = 0, 1, \dots, m-1; j = 0, 1, \dots, n-1$)で表現する。 $A(i, j)$ の各々の要素は、そのサイズが $\frac{X_{max}-X_{min}}{m} \times \frac{Y_{max}-Y_{min}}{n}$ である格子領域を示す。ステレオカメラの座標系に対し、格子領域 $A(i, j)$ は、

$$X_{min} + \frac{X_{max} - X_{min}}{m}i < X < X_{min} + \frac{X_{max} - X_{min}}{m}(i+1) \quad (19)$$

$$Y_{min} + \frac{Y_{max} - Y_{min}}{n}j < Y < Y_{min} + \frac{Y_{max} - Y_{min}}{n}(j+1), \quad (20)$$

の範囲を表すため、投影点 Q_l, Q_r のX座標値及びY座標値がこの範囲内の値であるなら、 Q_l, Q_r のX,Y座標値の平均及び q_l, q_r の輝度値の平均をそれぞれ求め、 $A(i, j)$ に格納する。投影データを全て格納した後、データが書き込まれていない要素を、隣接する要素のデータを用いて補間する。

図4に示すように、道路平面 P 上に存在する点 Q の場合、左右画像中の対応点が点 Q に投影される。点 Q が含まれる格子領域を $A(i, j)$ とすると、 $A(i, j)$ に蓄積されている投影点の輝度の平均値 q_l, q_r は、それぞれ類似の値を示す。一方、図5に示すように、道路平面上に存在しない点 V の場合、 V の左右画像における撮像点 q_l, q_r は、それぞれ道路平面上の異なる点 Q_l, Q_r に投影される。このため、 Q_l には、右画像から V とは異なる撮像点 w_r が投影され、 Q_r には、左画像から V とは異なる撮像点 w_l が投影される。よって、 Q_l への投影点が格納される $A(i_l, j_l)$ 及び Q_r への投影点が格納される $A(i_r, j_r)$ には、左右画像からそれぞれ異なる輝度値が格納されるため、左画像からの輝度値が右画像からの輝

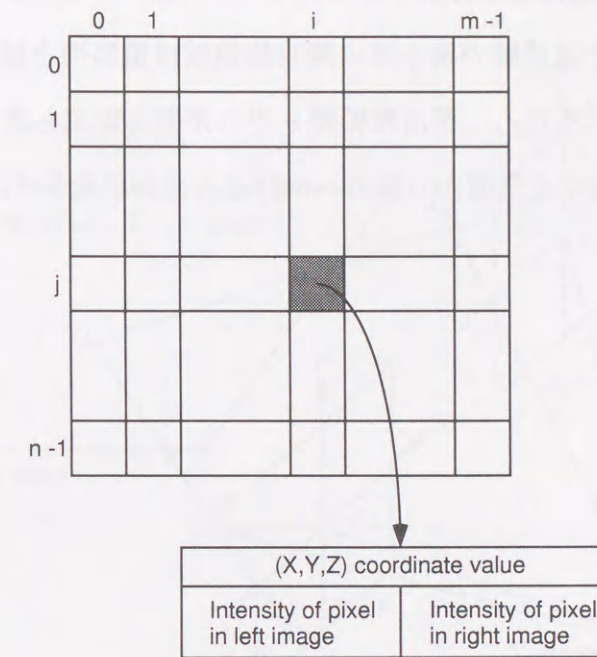


図3: Data structure of $A(i, j)$

度値と異なる $A(i, j)$ の要素を求めることで、道路面とは高さが異なる領域（以後、非道路領域と呼ぶ）を示す平面投影差分を作成する。

v_l が $A(i, j)$ に格納されている左画像からの輝度値であり、 v_r が $A(i, j)$ に格納されている右画像からの輝度値であるとする。次の条件が満たされる時、 $A(i, j)$ を設定平面上に無い領域と判定する。

$$\frac{|v_l - v_r|}{\frac{v_l + v_r}{2}} > T_v \quad (21)$$

$A(i, j)$ の全ての要素に対し、式(21)を適用し移動車前面の非道路領域を示す平面投影差分を作成する。式(21)において、 T_v は左カメラと右カメラの特性の違いを調べることでより設定する閾値である。移動車を自律走行させる前にステレオ画像中で道路上の対応点を数点選択し、これらの点の正規化輝度差を上式より求める。そして、その平均 μ と分散 σ から、閾値 T_v を $\mu + 3\sigma$ として求める。この閾値は、ステレオカメラの特性が変化しない限り、以後、固定値として用いることができる。図6(c)及び(d)に $A(i, j)$ に格納された左画像からの輝度値及び右画像からの輝度値により投影画像を作成した例を示す。図

6(c) 及び (d) では、真の道路領域は真上から見下ろした形で投影される。一方、道路平面とは高さが異なる壁面や道路脇の樹木等の障害物領域は道路面と接地した位置から後方へ倒れ込む形で投影されるため、平面投影差分中の非道路領域も図 6(e) に示す白色領域のように、障害物が接地する位置から後方へ倒れ込んだ形状を示す。

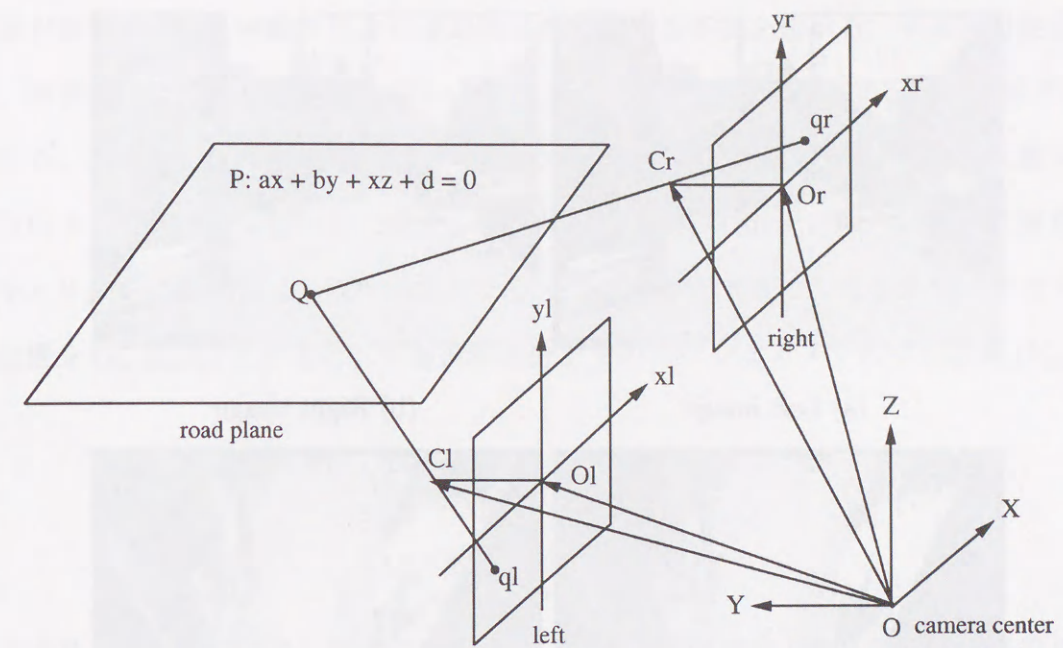


図 4: Point Q existing on the road plane

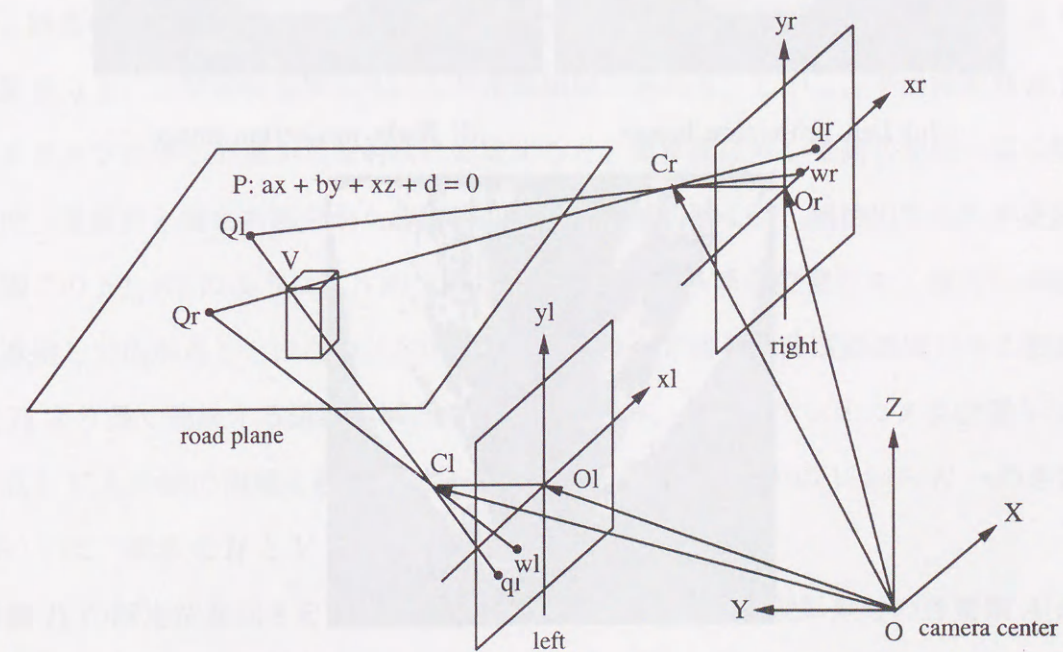


図 5: Point V not existing on the road plane

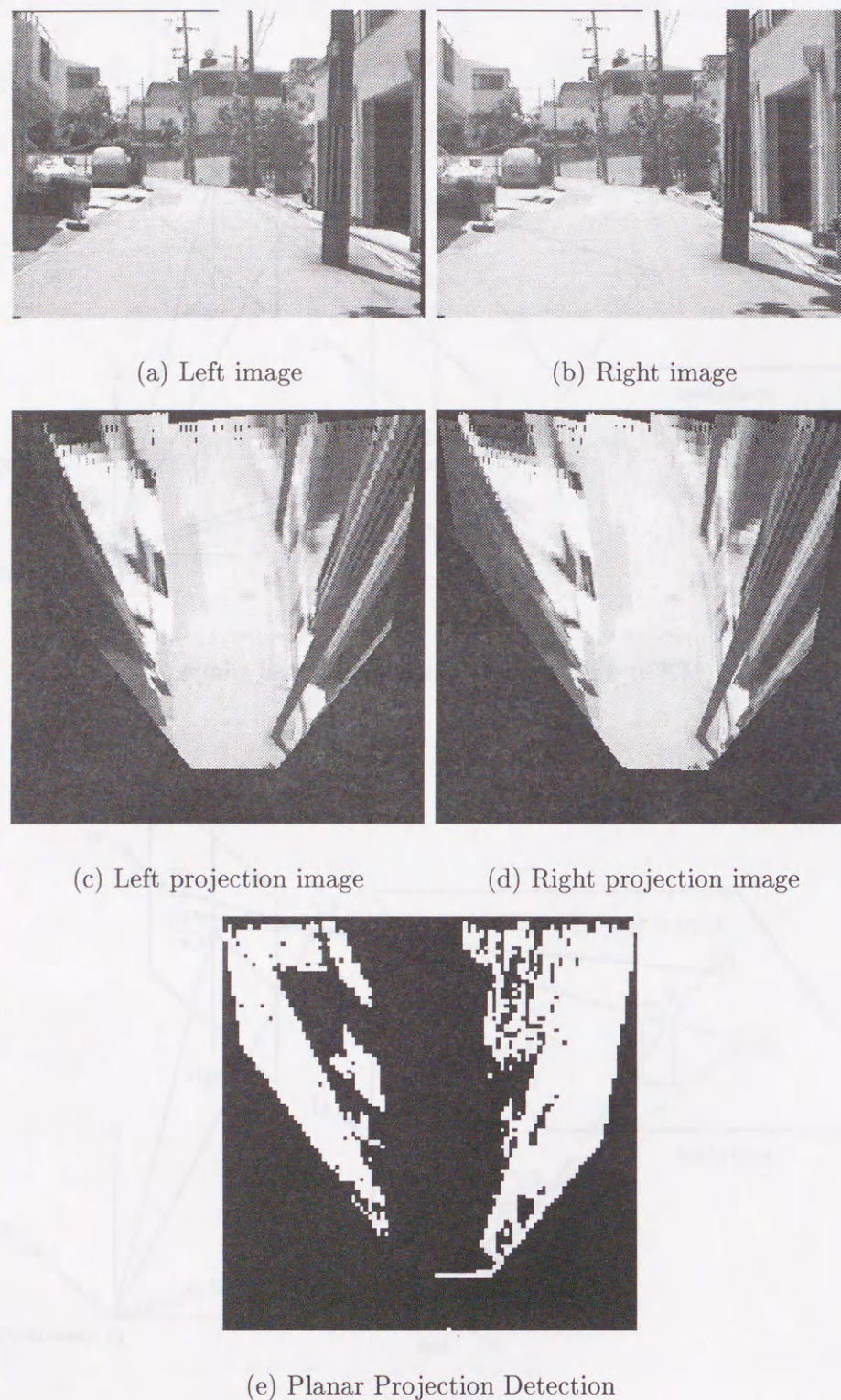


図 6: Example of projection images and Planar Projection Subtraction

2.2.4 道路領域の抽出

平面投影差分から、移動車前方の道路領域を抽出する手法を述べる。平面投影差分において、障害物は、道路平面と接している位置から後方に放射状に伸びる形で記述される。このため、ステレオカメラの中心位置（左カメラの焦点位置と右カメラの焦点位置を結んだ線分の中点位置）から放射状に走査して、非道路領域が放射状に伸び始める位置を求める。キャリブレーションにより求めたステレオカメラの座標系におけるステレオカメラの中心位置を (x_c, y_c, z_c) とすると、平面投影差分上でのステレオカメラの中心位置 (V_{px}, V_{py}) は

$$V_{px} = \frac{x_c * m}{X_{max} - X_{min}} + \frac{m}{2} \quad (22)$$

$$V_{py} = n - \frac{y_c * n}{Y_{max} - Y_{min}} \quad (23)$$

として求めることができる。図 7 に示すように、平面投影差分上をステレオカメラの中心位置 (V_{px}, V_{py}) から時計回りに放射状に走査することにより、非道路領域（図 7 の灰色領域）がステレオカメラの位置に対し後方へ伸び始める位置を求める。左右カメラの特性の違い、路面の微妙な凹凸、及び設定した平面やカメラパラメータの持つ誤差等により、平面投影差分上には障害物領域以外にも非道路領域が現れる。しかし、平面投影差分上でステレオカメラの中心位置から放射状に走査すると、道路面に対して高さを持つ真の障害物領域は、道路面と接する部分から連続的に非道路領域が続くが、誤抽出された非道路領域は、図 7 の $E1, E2$ のように断片的であり、ステレオカメラの位置に対し後方に非道路領域が連続して広がることはない。このため、走査線上で後ろに非道路領域が予め設定した閾値 T_h より長く連続する道路領域との境界点を求め、最もステレオカメラ位置 V に近い境界点と V との間の領域を移動可能な道路領域とする。図 7 中の V から K への走査線 L においては、境界点 H と V の間が移動可能領域となる。

閾値 T_h の設定法を図 8 を用いて説明する。閾値 T_h は平面投影差分の各要素 $A(i, j)$ 毎に設定される。また、障害物として検出する高さの下限値 h は、予め決めておく。実験では歩道の高さ（10cm）を h とし、高さが h 以上の物体を障害物として求めた。高さ h の障害物が平面投影差分上の要素 $A(i_k, j_k) (0 \leq i_k \leq m - 1, 0 \leq j_k \leq n - 1)$ に存在するな

ら、ステレオカメラ座標系における障害物頂点 T の X, Y, Z 座標値 (T_x, T_y, T_z) は

$$T_x = A_x + ha \quad (24)$$

$$T_y = A_y + hb \quad (25)$$

$$T_z = A_z + hc, \quad (26)$$

として得られる。ここで、 (A_x, A_y, A_z) は $A(i_k, j_k)$ の X, Y, Z 座標値、 (a, b, c) は道路平面の方程式 $P: ax + by + cz + d = 0$ の単位法線ベクトルである。ステレオカメラ座標系の $X - Y$ 平面が設定した道路平面とほぼ平行であるとすると、 $(a, b, c) = (0, 0, 1)$ となるため、 (A_x, A_y, A_z) は次式より得られる。

$$A_x = \frac{(i_k - \frac{m}{2})(X_{max} - X_{min})}{m} \quad (27)$$

$$A_y = \frac{(n - j_k)(Y_{max} - Y_{min})}{n} \quad (28)$$

$$A_z = -d \quad (29)$$

ただし、 $X_{min} < X < X_{max}, Y_{min} < Y < Y_{max}$ は移動車前方の道路領域の探索領域であり、 $m \times n$ は平面投影差分を表わす $A(i, j)$ のサイズである。また、ステレオカメラ座標系の原点位置は画像下端の中央に設定されている。

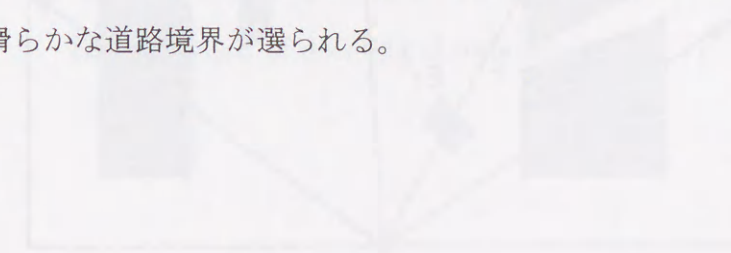
2.2.2 章で述べた手法を用い、左画像の焦点位置 C_l と T を結ぶ直線が平面 P と交差する点 T_l が求まる。同様に、右画像の焦点位置 C_r と T を結ぶ直線が平面 P と交差する点 T_r が求まる。よって、2.2.3 章で述べた手法により、 T_l を含む平面投影差分上の要素 $A(i_{kl}, j_{kl})$ 及び T_r を含む平面投影差分上の要素 $A(i_{kr}, j_{kr})$ が得られる。もし h より高い障害物が $A(i_k, j_k)$ に存在するなら、 $A(i_k, j_k)$ より後方へ倒れ込む障害物形状の長さは

$$T_h(i_k, j_k) = \min(\sqrt{(i_k - i_{kl})^2 + (j_k - j_{kl})^2}, \sqrt{(i_k - i_{kr})^2 + (j_k - j_{kr})^2}). \quad (30)$$

より長くなる。このため、平面投影差分の各要素 $A(i, j)$ において算出された $T_h(i, j)$ を閾値 T_h として用いることにより、 h よりも高い障害物と道路領域との間の境界点が走査線 L 上で得られる。実験では、歩道の高さ (10cm) に h を設定した。平面投影差分上にお

いて、誤抽出領域の長さは、高さ h の障害物が生成する障害物形状より通常短いため、閾値 T_h を用いることで、これらの誤抽出領域を除去することができる。

道路領域を抽出後、移動車が通過できない狭い領域を除去する。平面投影差分上で移動車の幅を W_v とし、 W_v より狭い領域を除去する。まず、平面投影記述上で道路領域 R_t の輪郭を $\frac{W_v}{2}$ だけ縮小する (図 9(b))。この縮小処理により、幅が W_v より狭い領域がステレオカメラの位置を含んでいる領域 R_c から分離される。次いで、 R_c を幅 $\frac{W_v}{2}$ だけ拡大する (図 9(c) の領域 R_d)。最後に、 R_d が R_t と重なっている領域を求め、移動車が走行できる移動可能空間とする。この処理により R_t の輪郭付近に発生する狭く突出した領域が除かれるので、滑らかな道路境界が選られる。



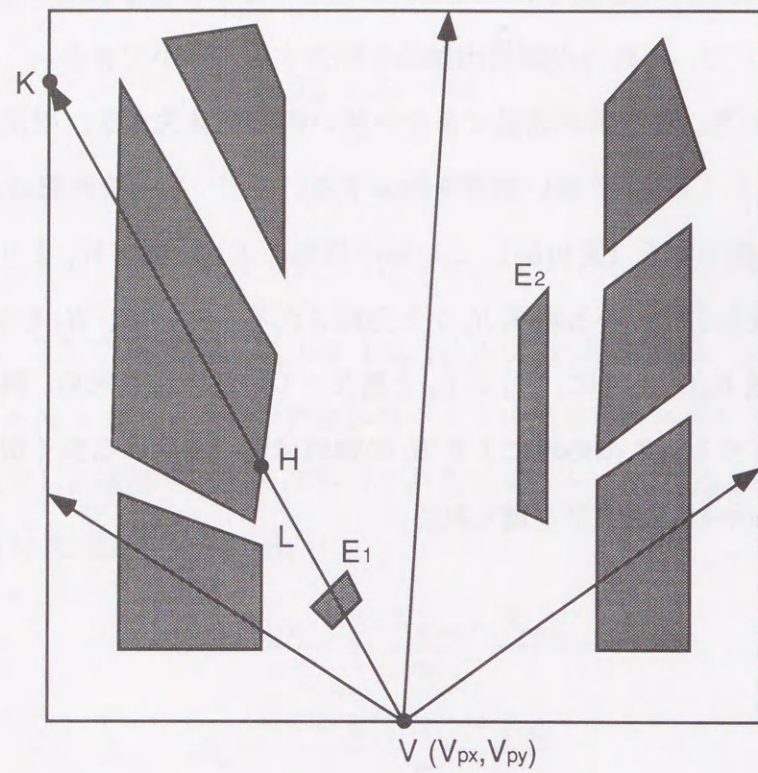


图 7: Detection of road boundaries

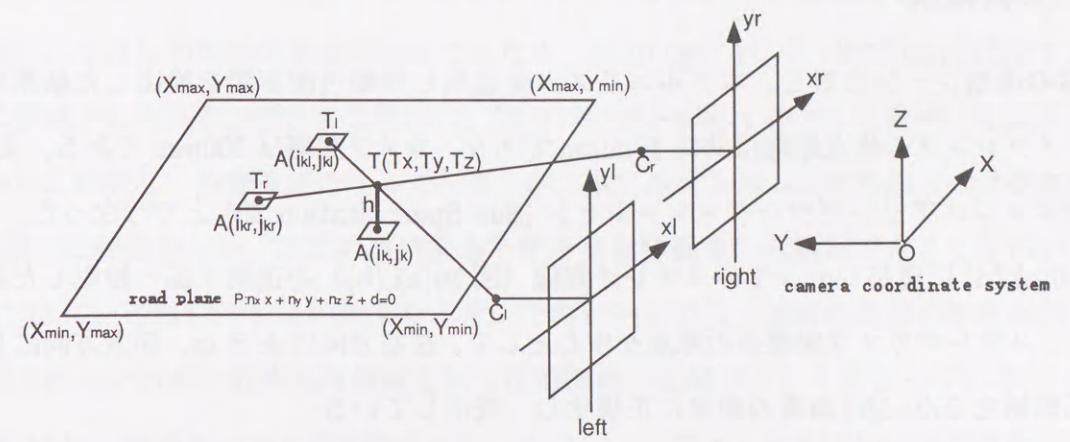
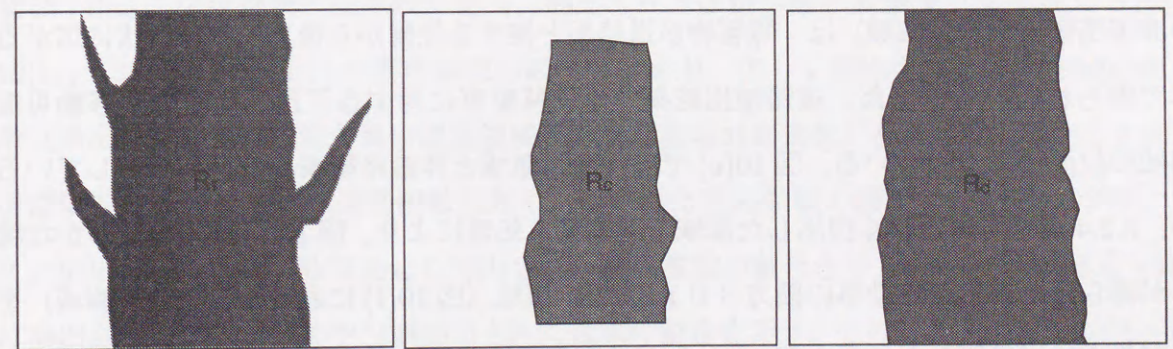


图 8: Method of deciding thresholding value T_h



(a) Region R_t

(b) Region R_c

(c) Region R_d

图 9: Elimination of tight area

2.3 実験結果

実際の道路シーンに対し、本アルゴリズムを適用し移動可能空間を検出した結果を示す。カメラレンズの焦点距離は共に 12.5mm であり、カメラ間隔は 200mm である。また、処理はエンジニアリングワークステーション (Sun Sparc Station 20) 上で行なった。

図 10(c),(d) に道路シーン 1 のステレオ画像 (図 10(a),(b)) を道路平面へ投影した画像を示す。ステレオカメラ座標系の原点を中心として、左右方向に ± 8 m、前方方向に 50m の平面領域を 500×500 画素の画像に正規化し、表示している

ステレオカメラ座標系の原点位置は画像下端の中央であり、画像の上に行くほど移動車から遠くなっている。再舗装部分やマンホールカバー等、道路表面上の模様は、左右の投影画像において、同位置に真上から見下ろす形で投影されているが、壁面、フェンス、電柱、及び交通標識等、道路平面と高さが異なる障害物は、左右の投影画像において投影位置が異なり、かつ道路面と接する位置から後方に放射状に倒れ込んだ形で投影されている。

図 10(e),(f) に道路シーン 1 の平面投影差分及び道路抽出結果を示す。平面投影差分中の非道路領域 (白色領域) は、障害物が道路面と接する位置から後方に、放射状に広がる形で得られている。また、道路抽出結果では、移動車に対する三次元空間中の移動可能領域の位置が求まっている。図 10(e) では、道路領域と非道路領域の境界が凸凹しているが、2.2.4 章で述べた細く突出した領域を除去する処理により、図 10(f) に示す滑らかな境界が得られている。移動車の前方 30 m までの領域 (図 10(f) において、下 $\frac{3}{5}$ の領域) を左画像中に逆投影した結果を図 10(g) に示す。黒色領域が、移動可能な道路と判定された領域である。図 10(h) に、左画像中の道路境界を対話的にトレースして求めた移動可能な道路領域の真値を示す。図 10(g),(f) において、共通視野から外れている画像左端部分は除外している。図 11 及び 12 に道路シーン 2 及び 3 の抽出結果を示す。これらの道路シーンでは、再舗装部分、マンホールカバー、路面にペイントされた交通標識、及び強い影領域の周辺に誤検出領域が多数発生する。このため、Storjohann[13] 等が提案した従来手法では、移動車の前面に障害物が多数誤検出され、通過できる移動可能空間が存在しなく

なってしまうが、本章で提案する平面投影ステレオ法は、平面投影差分上で障害物の形状を調べ、これらの誤抽出領域を除去するため、図 10 (g)、11(c)、及び 12(c) に示す正しい道路領域が得られる。通常の道路には、影やペイントが必ず存在するため、従来手法では誤検出が頻発し、移動不能になってしまうが、本アルゴリズムは道路面上のテクスチャー (模様) に頑健なため、実際の道路を走行する自律移動車にも適用できることがわかる。

図 13 に、道路シーン 4 の結果を示す。このシーンでは、道路の左端が駐車車両によって隠されているが、駐車車両領域を除く道路領域の位置が正しく求まっている。

図 14 に、道路シーン 5 の結果を示す。このシーンでは、前方に歩行者が存在し、その左側は移動車が走行できない狭い領域となっているが、本アルゴリズムは 2.2.4 章で述べた狭領域除去処理を備えるため、歩行者左の走行不可領域を除いた道路領域を移動可能な領域として求めることができる。

定量評価のために、図 10(g) と (h)、図 11(c) と (d)、12(c) と (d)、図 13(c) と (d)、及び 14(c) と (d) をそれぞれ比較し、道路抽出結果がどれだけ真の道路領域と一致するかを調べた。表 4 にその結果を示す。真の道路領域の画素数を A 、抽出された道路領域の画素数を B 、正しく抽出された道路領域の画素数 (抽出された道路領域と真の道路領域との共通領域の画素数) を C 、誤って抽出された道路領域の画素数 (真の道路領域以外で抽出された道路領域の画素数) を D とする。また、抽出率 $E = \frac{C}{A} * 100$ は正しく抽出された道路領域の割合を示し、誤り率 $F = \frac{D}{B} * 100$ は抽出された道路領域の中で誤抽出された領域の割合を示す。この 5 シーンにおいて、抽出率 E の平均は 93.4%、誤り率の平均は 3.1% であった。また、この 5 シーンを含む 30 の道路シーンに対し、道路抽出実験を行った結果、抽出率の平均は 86.9%、誤り率の平均は 3.3% であった。

移動車が直進する際、特に重要なのは移動車の正面方向の領域であり、横の領域ではない。つまり、移動車の正面方向の領域は確実に抽出される必要があるが、移動車から離れた横方向の領域は移動車の走行に影響を及ぼさないため、抽出精度が低くても問題ないと言える。このため、移動車の正面方向の領域か横の領域かという重みを抽出結果に付加

し、評価を行った。

抽出された道路領域の X 座標値を X_v とする。移動車に積載されたステレオカメラの座標系の X 軸は、図 2 に示すように移動車の進行方向に対して垂直であり、 $X_v = 0$ の位置が移動車の正面に相当する。このため、定量評価で画像中の道路領域の画素数を用いる代わりに、抽出された道路領域の 3 次元空間中の X 座標値の逆数 $\frac{1}{X_v}$ の総和を用いる。

この重み付け評価値では、移動車正面に誤抽出及び未抽出領域が存在すると、抽出率及び誤り率が大幅に悪化する。逆に、移動車の横方向領域の誤抽出及び未抽出領域は、抽出率及び誤り率に大きな影響を与えないため、より実際の走行に則した評価値といえる。表 4 の重み付け抽出率 G 、及び重み付け誤り率 H の項にこの重み付けされた評価結果を示す。表 4 の 5 シーンにおいて、重み付け抽出率 G の平均は 99.2%、重み付け誤り率の平均は 0.7% であった。また、この 5 シーンを含む 30 の道路シーンに対し、道路抽出実験を行った結果、重み付け抽出率 G の平均は 89.3%、重み付け誤り率 H の平均は 1.4% であった。

表 4 の抽出率 E は 88% ~ 96% であり、最悪 12% の領域が抽出されていないが、図 10(g)、11(c)、12(c)、13(c)、及び 14(c) を見ると、これらの未抽出領域は道路の端の部分に集中しており、移動車に近い領域は確実に抽出できていることがわかる。このことは、表 4 の全ての重み付け抽出率 G が抽出率 E を上回っていることから確認できる。移動車は通常、道路境界や障害物に対し距離を置いて移動するため、これらの未抽出領域が移動車の制御に影響を及ぼさないことが以上の結果からわかる。

表 4 では、道路シーン 4 と 5 の誤り率 F が高い値を示している。しかし、これらの重み付け誤り率 H が F より小さいことからわかるように、誤抽出領域は遠く離れた歩道上や (図 13)、道路の左端 (図 14) 等移動車から離れた場所に現れており、その大部分は道路境界から細く突き出した形状を示している。このため、これらの誤抽出領域が移動車の走行経路として選択される心配は無い。

安全性を考慮し、道路の幅が移動車の 2 倍以上あると仮定すると、道路端付近の 10% 程度の領域が未抽出であっても、移動車が走行できる十分な領域が残る。このため、抽出された移動可能領域周辺の 10% の領域を危険地帯として除去し、移動車を制御すること

が可能であり、抽出された領域の周辺に 1.4% 程度の誤抽出領域が存在しても移動車の走行には影響を及ぼさないことがわかる。

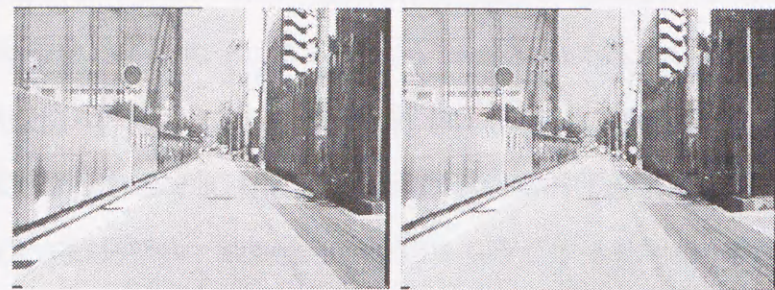
図 15 に移動しながら入力したステレオ画像系列に本アルゴリズムを適用した結果を示す。道路面の凹凸の影響で走行中に車が上下に揺れ、走行前に設定した道路面の方程式と実際の道路面との間にはずれが生じる。このため、道路面への逆投影モデルを単純に用いた従来手法では、道路端の白線、センターライン及び横断歩道等のテクスチャー周辺に誤抽出領域が多数発生し、道路上の移動可能領域を正確に抽出することは困難であったが、本アルゴリズムは平面投影差分の特性、

- (1) 平面投影差分上では、障害物領域は障害物が設定平面と接地する位置から後方に放射状に倒れ込んだ形状を示す
- (2) 平面投影差分上では、後方に倒れ込んだ障害物領域の長さは、ステレオカメラに対しての障害物の位置と設定平面に対しての障害物の高さに依存する

を用いることにより、道路面のテクスチャーに影響されず、移動可能領域が正しく抽出できる (図 15(g) ~ (l))。

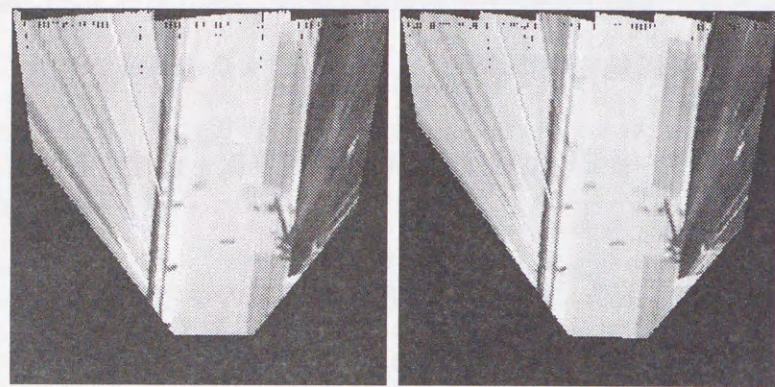
現在の処理時間は 1 秒/フレームであるが、本アルゴリズムは画像の単純な投影と差分処理から構成されているため、簡単なハードウェアによりビデオレート化できる。また、実験では、50 m 前方までを抽出対象としたが、焦点距離の長いレンズを用いれば、より遠方まで抽出することが可能である。

走行環境としては、アスファルト等により舗装された道路を想定しているが、路面には凹凸やギャップが存在するため、道路平面とカメラ間の幾何関係が振動により大きく変化する可能性がある。また、坂道の最初と最後のように、車の存在面と観測面が異なる場合も考えられる。このため、道路平面とカメラ座標系間の幾何関係を、走行中に常時校正して行くことが今後の課題である。



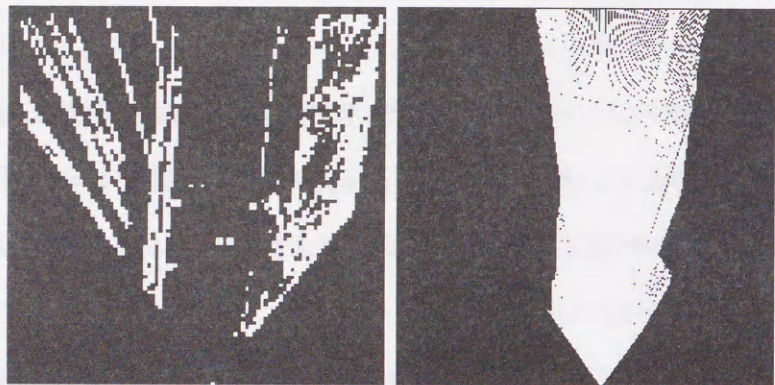
(a) Left image

(b) Right image



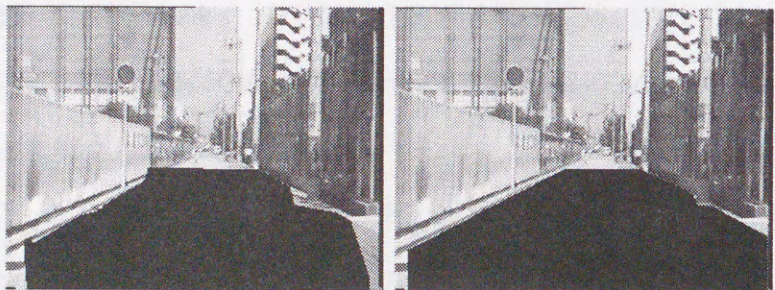
(c) Left projection image

(d) Right projection image



(e) Planar Projection Subtraction

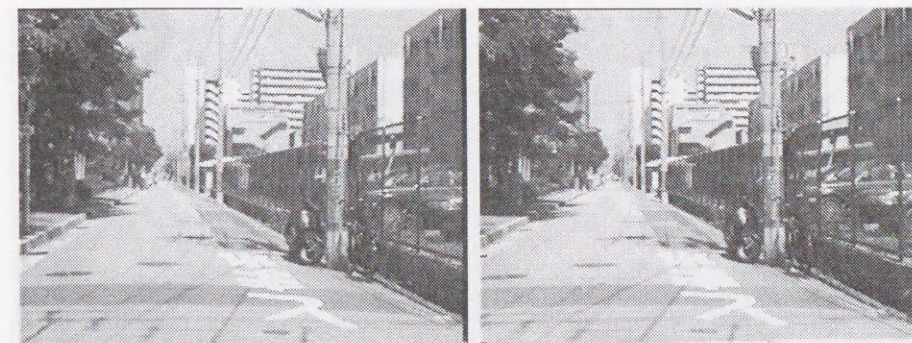
(f) Road extraction result



(g) Reverse projection image

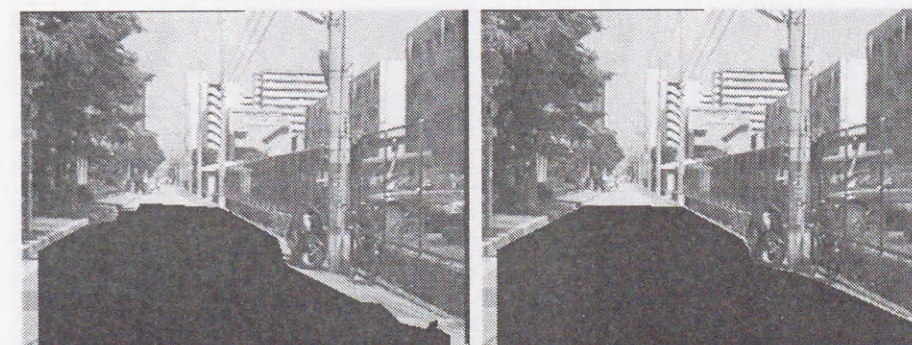
(h) Real road image

图 10: Result for the road scene 1



(a) Left image

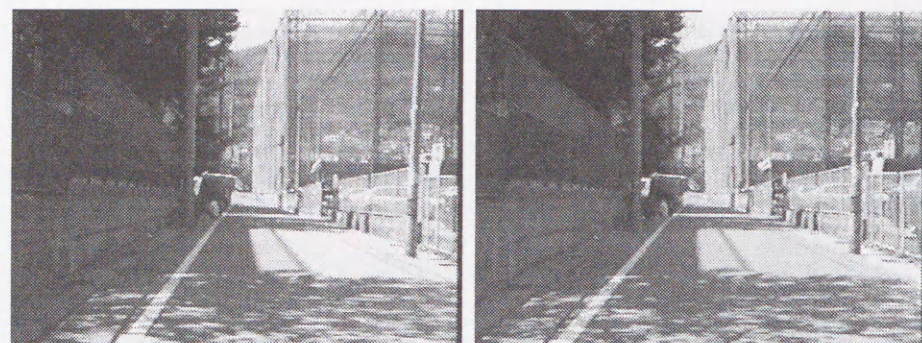
(b) Right image



(c) Reverse projection image

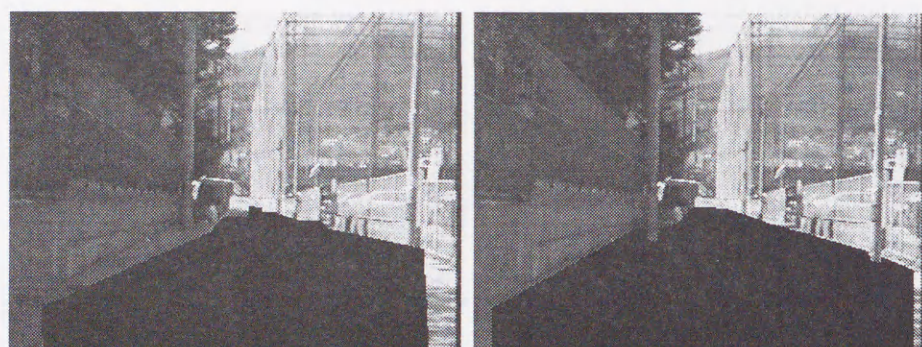
(d) Real road image

图 11: Result for the road scene 2



(a) Left image

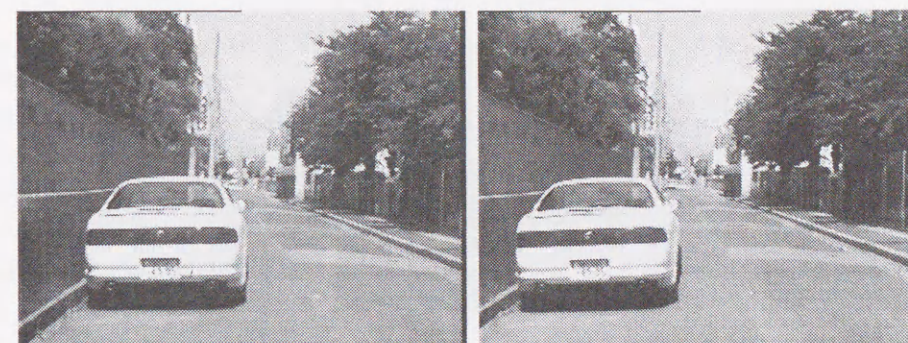
(b) Right image



(c) Reverse projection image

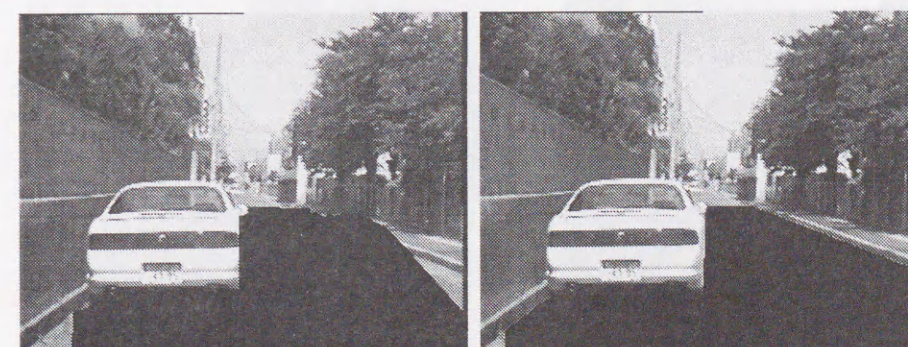
(d) Real road image

⊠ 12: Result for the road scene 3



(a) Left image

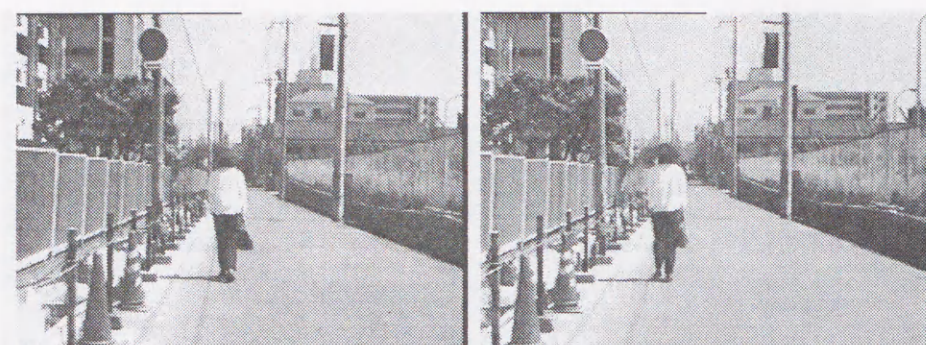
(b) Right image



(c) Reverse projection image

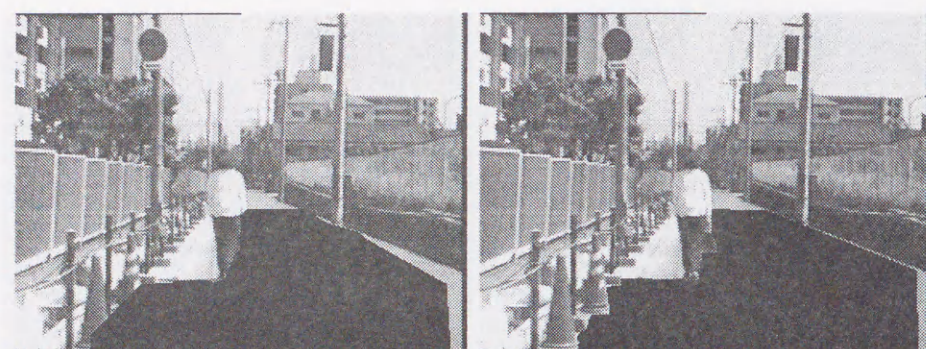
(d) Real road image

⊠ 13: Result for the road scene 4



(a) Left image

(b) Right image



(c) Reverse projection image

(d) Real road image

⊠ 14: Result for the road scene 5



(a) frame 0

(b) frame 20

(c) frame 40



(d) frame 60

(e) frame 80

(f) frame 100



(g) frame 0

(h) frame 20

(i) frame 40



(j) frame 60

(k) frame 80

(l) frame 100

⊠ 15: Result for the sequences acquired from the moving vehicle, (a)-(f) Left image, (g)-(l) Reverse projection image

表 4: Extraction ratio

Road scene	1	2	3	4	5
Total number of pixels in road area A	95078	85104	90597	71653	66892
Total number of pixels in extracted road area B	91098	80192	81137	72155	68347
Total number of correct pixels C	90107	79512	79956	67417	64419
Total number of wrong pixels D	991	680	1181	4738	3928
Extraction ratio $E = C / A * 100$ (%)	94.8	93.4	88.3	94.1	96.3
Error ratio $F = D / B * 100$ (%)	1.1	0.8	1.5	6.6	5.7
Weighted extraction ratio G (%)	99.7	99.5	98.8	99.1	99.0
Weighted error ratio H (%)	0.02	0.01	0.01	1.7	1.9

3 ステレオ視による移動障害物位置検出アルゴリズム

2章で述べたアルゴリズムは、移動車前方の移動可能な領域を表わす移動可能空間マップを平面投影差分から得ることができるが、自分の動きが未知な場合、マップ中の障害物領域が移動体なのか、あるいは駐車車両等背景に含まれる静止物体なのかまで判断することはできない。車を安全かつ円滑に制御するためには、歩行者や走行中の車等を移動障害物として切り出し、その進行方向や位置を予測することが重要である。このため、本章では、移動する車から画像を用いて歩行者、自転車、車等の移動障害物を検出し、車に対する位置を算出するアルゴリズムを与える。

3.1 従来の手法

広い範囲において移動障害物を検出し、その3次元位置や進行方向を求めるには、画像を用いる手法が有効であるが、移動しながら得た画像から移動障害物を検出する場合、背景も動くため、フレーム間差分法 [52] や背景差分法 [53] といった単純な手法は適用できない。また、移動障害物は動き方によって画像中での見え方やサイズが大きく変化し、かつ歩行者や自転車といった形状自体が常に変形する非剛体障害物も存在するため、検出が容易ではない。

移動する観測系から撮影された画像中で、移動する障害物の領域を検出するには、画像中の動き情報(オプティカルフロー)を解析する手法が有効であり、Tian 等 [54] の ATH(Adaptive Hough Transform) を用いる手法や、太田 [55] の対象物までの距離を推定する手法等が提案されている。しかし、これらの手法はカメラの動きを推定する必要があるため、ノイズに敏感であり、計算量も大きいといった問題点がある。画像中で背景の占める面積が移動障害物に比べて十分大きく、かつ、カメラが回転運動あるいは横方向へ平行移動する場合、背景のオプティカルフローから背景領域の平行移動量を求め、アフィン変換によりカメラが動く以前の背景画像を生成し、背景差分法で移動体を抽出する手法 [56] も提案されている。しかし、カメラが前後方向に平行移動する場合、背景のオプティカルフローからア

フィン変換パラメータを生成することが困難であり、移動車の障害物検出には適さない。このため、本章では、カメラの動きを推定せずにオプティカルフローから画像中の移動障害物領域が検出できるアルゴリズムを提案する [57]。カメラの動きを推定する必要が無いため、繰り返し演算が不要であり、高速な処理が実現できる。また、カメラが前後に平行移動する場合に対応できるため、移動車の障害物検出に適している。

検出された移動障害物の画像中の位置から、その3次元位置を計測する手法としては、ステレオ視が一般的であるが、画像中で抽出された移動障害物領域の3次元位置を連続的に計測するためには、ステレオ画像間と各フレーム間で移動障害物を同時に対応付ける必要がある。SAD(Sum of Absolute Difference) や相関法等を用い、ステレオ画像中で求めた移動障害物上の対応点をフレーム間で高速に追跡し、3次元位置を算出する手法 [58][59] が提案されているが、移動障害物が非剛体であったり、進行方向の変化で見え方やサイズが大きく変わる場合に、対応点をフレーム間で正確に追跡し続けることが難しく、3次元位置が安定に求まらない。ステレオ画像間で対応付けられた水平エッジをフレーム間で高速に追跡する手法 [60] や、モーションセグメンテーションされた画像中の領域の輪郭を追跡する手法 [61] 等も提案されているが、水平エッジを多く含み、かつ輪郭形状が滑らかな車等の対象物にしか適用できない。

一般道路上に存在する代表的な移動障害物としては、歩行者、自転車及び自動車等が考えられるが、これらはターン等により進行方向が変わると見え方が大きく変化する。また、歩行者や自転車は非剛体なので形状が常に変形し、従来手法では移動障害物の位置を安定に算出することが難しい。このため、本章では、2章で述べた平面投影ステレオ法を用い、対象物の形状や大きさの変化に強い移動障害物位置検出アルゴリズムを提案する。平面投影ステレオ法は、ステレオ画像を予想される道路平面上に投影することで、ステレオ画像間の対応探索を行なうことなく道路面と接触している障害物の3次元位置が算出できるため、障害物の形状やサイズが変化しても車両前方の障害物領域を除いた移動可能空間マップがフレーム単位で求まる。このため、オプティカルフローを用いて検出した画像中の移動障害物領域を、移動可能空間マップ上に投影することで、移動障害物の3次元位置を検出する。[62][63]。

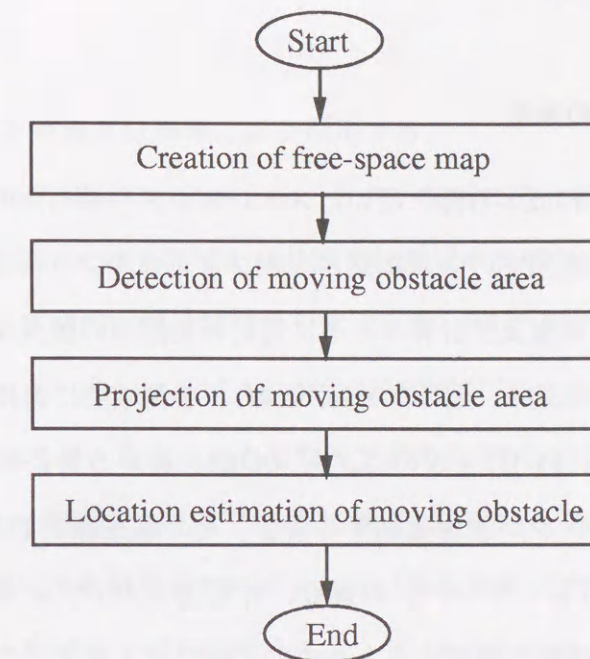


図 16: Block diagram of the proposed method

3.2 移動障害物位置検出法

図 16 に本アルゴリズムの処理の流れを示す。本アルゴリズムは、まず、2章で述べた平面投影ステレオ法を用い、前方の移動可能空間マップを作成する。次いで、オプティカルフローから画像中の移動障害物領域を抽出する。この移動障害物領域を移動可能空間マップ上に投影することで移動障害物と移動車との間の位置関係を求める。

3.2.1 移動可能空間マップの作成

平面投影ステレオ法は、ステレオ画像を道路平面へ投影することにより求めた平面投影差分上を放射状に走査することで、移動車と障害物との間の移動可能な領域を示す移動可能空間マップを作成する。この移動可能空間マップでは、移動車から観測可能な障害物が移動体も含めて移動可能領域の外周部分に存在する。このため、移動障害物が存在する方向が求めれば、この方向に伸びる移動可能領域の外周部分の位置から移動障害物の3次

元位置を得ることができる。

3.2.2 移動障害物領域の抽出

移動可能空間マップを作成する際に用いたステレオカメラの一方の画像から移動障害物領域を抽出する。まず、画像中の各点に対し、オプティカルフローを抽出する。オプティカルフローの抽出には、精度及び計算コストに優れた勾配法の改良手法[55]を用いた。次いで、画像中で小領域を設定し(図17の矩形領域)、小領域内に含まれるフローベクトルの方向から動きの消失点(FOE)を求める。同一の動きをする対象物上のフローベクトルは、観測系であるTVカメラが並進運動する場合、その延長線がFOEと呼ばれる1点に収束する。例えば、図17に示すシーンの場合、背景上のオプティカルフローはその延長線がある1点に、また、道路を横切ろうとしている歩行者上のオプティカルフローはその延長線が別の1点に収束する。このため、小領域に含まれる複数本のフローベクトルを用い、最小自乗法により動きのFOEを推定する。この過程は次のように表せる。

各フローベクトルから各フローの直線の方程式を求める。点 (x_i, y_i) においてフローベクトル (u_i, v_i) が得られたとする。これらの直線の方程式は、

$$A\vec{p} + \vec{c} = \vec{0} \quad (31)$$

$$A = \begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ \vdots & \vdots \\ a_n & b_n \end{bmatrix}, \vec{p} = \begin{bmatrix} x \\ y \end{bmatrix}, \vec{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \quad (32)$$

$$(a_i, b_i, c_i) = \frac{(-v_i, u_i, v_i x_i - u_i y_i)}{\sqrt{u_i^2 + v_i^2}} \quad (33)$$

として与えられる。ここで、 n は小領域内に含まれるフローベクトルの本数である。この式について

$$E = \|A\vec{p} + \vec{c}\|^2 \quad (34)$$

を最小とするFOE \vec{p} を特異値分解等により推定する。

推定されたFOE \vec{p} を式(34)に代入すると、その小領域に於ける推定残差が計算できる。この推定残差は、小領域内に背景領域と移動障害物領域が同時に含まれる場合(図17(a))、大きくなる。逆に、小領域内に背景領域しか含まれない場合(図17(b))、小さくなる。このため、推定残差が大きい時、小領域内に移動障害物が存在すると判定する。

小領域を画像中でラスタ走査し、その内部で求めたFOEの推定残差値を予め用意しておいた画像メモリの小領域に対応する各点に積算(投票)した推定残差画像を作成する。小領域のサイズは画像中の移動障害物のサイズ及び移動速度に依存するが、これらの情報は未知であるため、小領域サイズを固定すると、図18のように、小領域内で背景領域のフローベクトルが支配的になり移動障害物を見逃したり、移動障害物領域の内部に完全に小領域が収まり移動障害物領域内部を背景領域と誤判定したりする可能性がある。このため、小領域のサイズを複数設定し、それぞれのサイズで求めた推定残差画像を足し合わせる。つまり、画像中の各点 (i, j) において、複数サイズの少領域内で求めた推定残差値の総和を求め、点 (i, j) に書き込んだ画像を作成する。すると、移動障害物のサイズや位置に依らず、移動障害物の存在領域及びその周辺部の輝度値が大きい画像(図19(a)では、輝度値が高い領域を赤色で示している)が得られ、この画像をしきい値処理することで移動障害物の画像中の存在領域が抽出できる(図19(b))。

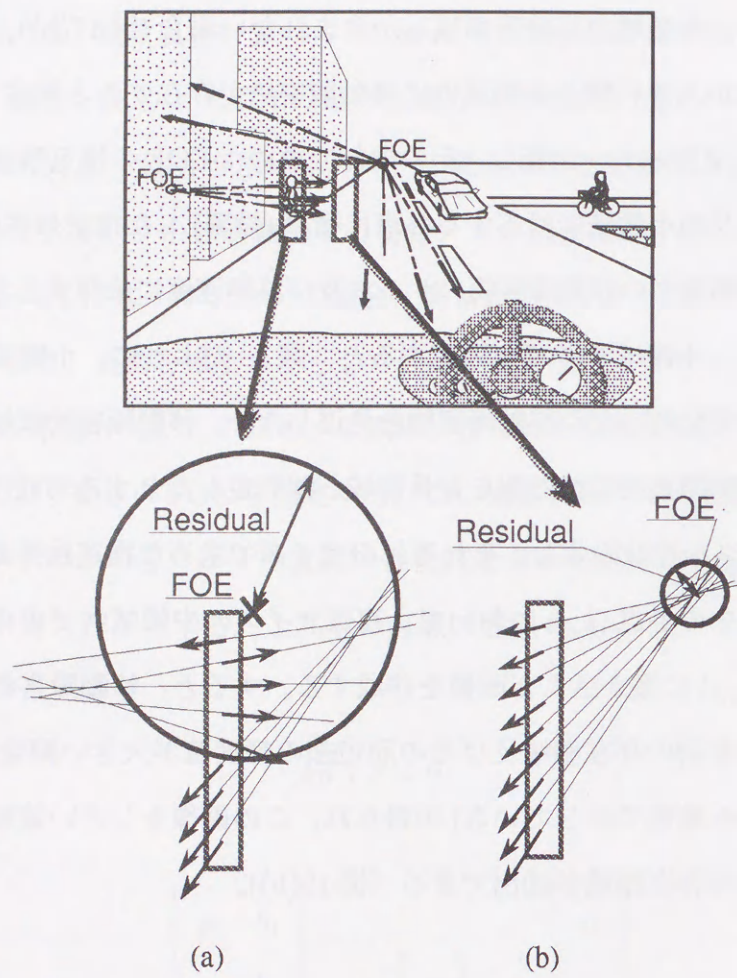


图 17: Basic idea of FOE estimation

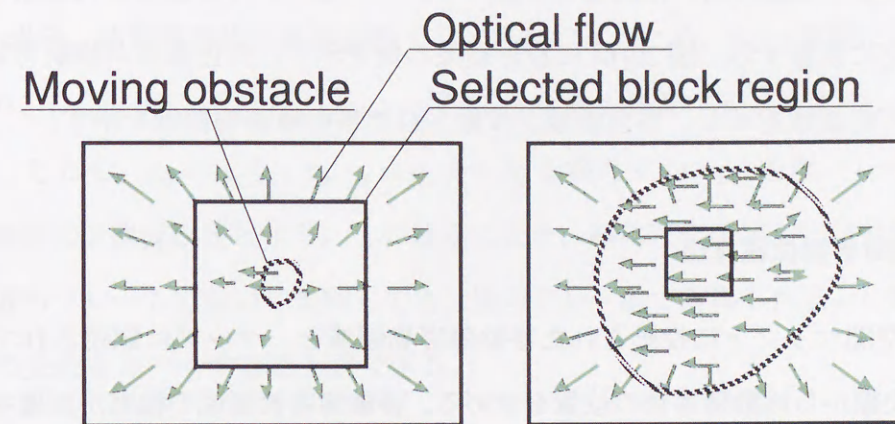
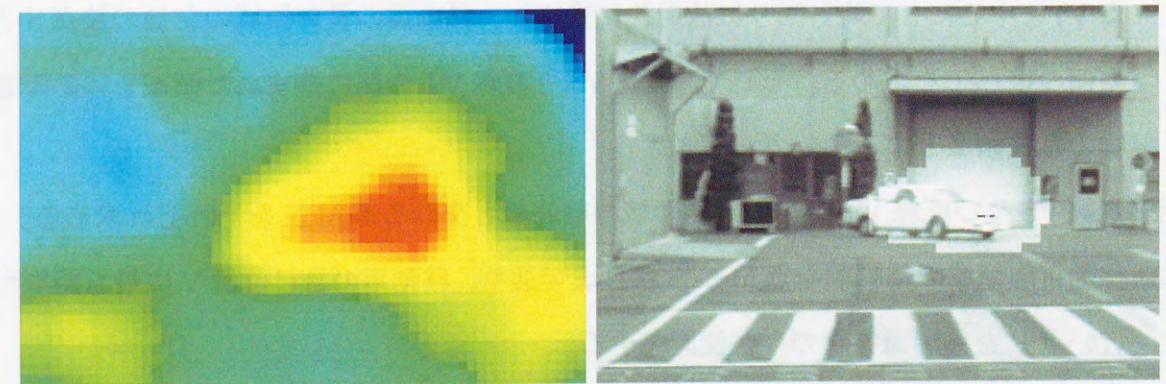


图 18: Block region size and residual error



(a) Residual error image (b) Moving Obstacle area

图 19: Extraction result of Moving Obstacle Area

3.2.3 移動障害物領域の投影

画像中で抽出された移動障害物領域を移動可能空間マップ上に投影する。移動可能空間マップを作成する際に用いた画像の投影パラメータを用いて移動障害物領域を、設定した道路平面上に投影する。図 20(b) に投影結果の例を示す。灰色領域が移動可能空間マップ上の移動可能領域を示し、黒色領域が投影された移動障害物領域を示す。

3.2.4 移動障害物位置算出

移動可能空間マップ上に投影された移動障害物領域と、マップに記述されている移動可能空間の交線から移動障害物の位置を求める。移動障害物領域の輪郭が画像中で完全に抽出されれば、移動障害物の投影領域と移動可能空間は互いに接触するが、オプティカルフローから正確に移動障害物領域の輪郭を求めることは難しい。通常、図 20(a) に示すように、画像中で切り出された移動障害物領域は、真の輪郭からはみ出していたり、欠けていたりする。このため、移動障害物の投影領域は、図 20(b) に示すように移動可能空間と接触しない場合が多い。この点を考慮し、以下の手順に従って移動障害物位置を求める。

- (1) 図 21 に示すように、移動可能空間マップ上を移動車の自己位置 (マップ上の下端中央点 V) を中心にして時計回りに放射状に走査し、移動体の投影領域の左側の接点 S と右側の接点 T を求める。そして、直線 VS と直線 VT の間に挟まれた領域を移動障害物の存在領域とする。
- (2) 直線 VS 及び直線 VT が移動可能領域の輪郭線と交わる点 M, N をそれぞれ求める。
- (3) 移動可能領域の輪郭線を点 M から点 N まで辿り、移動車の自己位置 V に最も距離が近い点 P を移動障害物の存在位置とする。

移動可能空間マップにおいて、移動可能領域の輪郭線部分に道路領域とは高さの異なる障害物の前面が存在する。また、移動車の自己位置に最も近い点が障害物回避等において重要な情報となる。このため、移動障害物の存在領域に含まれる移動可能領域の輪郭線上の点において、最も移動車の自己位置に近い点を移動障害物の位置としている。

移動可能領域及び移動障害物領域の抽出誤りの影響を軽減するため、過去のフレームで求めた移動障害物位置を用い、現フレームの算出結果に修正を施す。現フレーム f_i を含む直前の t フレームの移動障害物位置算出結果を $(x_i, y_i, f_i), \dots, (x_{i-t+1}, y_{i-t+1}, f_{i-t+1})$ とする。 t が小さい場合、移動障害物は直進運動すると仮定できるため、x-y-f 空間において最小自乗法により、 $(x_i, y_i, f_i), \dots, (x_{i-t+1}, y_{i-t+1}, f_{i-t+1})$ を通る直線式 $x = a_x f + b_x, y = a_y f + b_y$ を求める。そして、 $x_i = a_x f_i + b_x, y_i = a_y f_i + b_y$ を満たす (x_i, y_i) を現フレーム f_i における移動障害物の 3 次元位置とする。この修正により、移動可能領域及び移動障害物領域の抽出が一部のフレームにおいて失敗しても、他のフレームで獲得した正しい検出結果を用いて、その失敗をカバーすることができる。

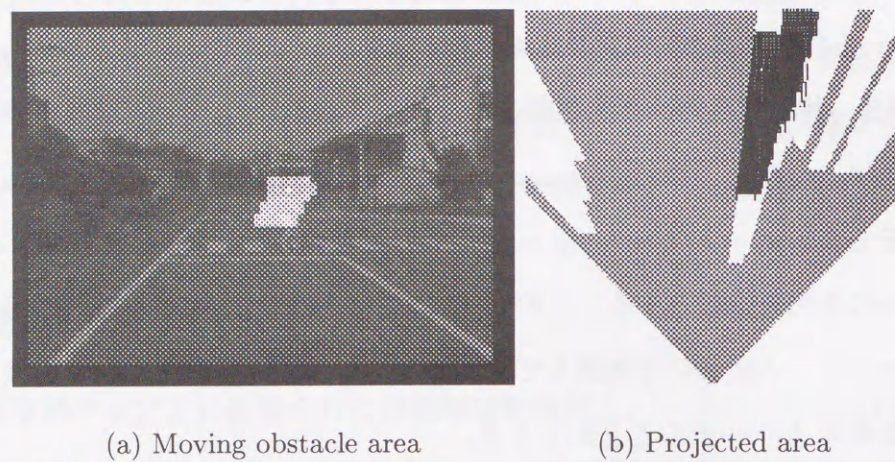


図 20: Projection of moving obstacle area to the free-space map

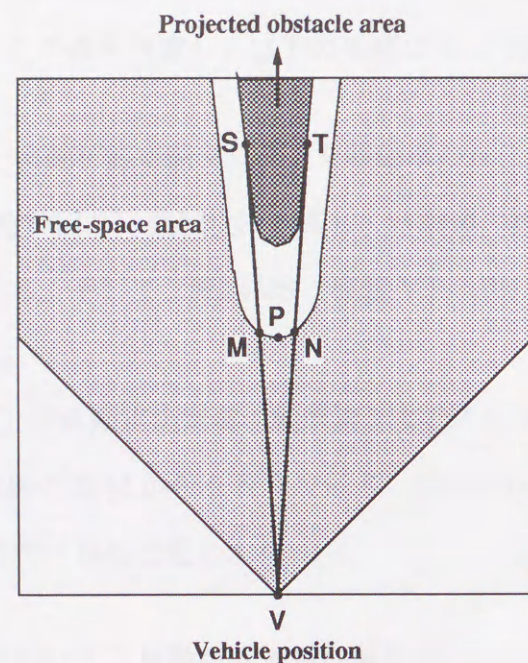


図 21: Estimating the location of the moving obstacle

3.3 実験結果

工場内の道路を 10～15km/h で走行し、収集した連続ステレオ画像を用いて移動障害物の位置を算出する実験を行なった。車のルーフキャリア上にステレオカメラを設置し、同期入力できる 2 台の VTR に走行中のステレオ画像を収録した。カメラレンズの焦点距離は 7.5mm、カメラ間隔は 300mm である。また、処理はエンジニアリングワークステーション (AS4085:SPARC Station 20 と同等) 上でソフトウェアにより行なった。

移動障害物は、歩行者、自転車、車の 3 種類であり、図 22～27 の (a)～(d) に各シーンのフレーム例 (左画像) を示す。各シーンは 40～170 フレームで構成されている。本アルゴリズムをこれらのシーンに適用し、自転車に対する移動障害物の位置をフレーム毎に算出した結果を図 22～27 の (e) に示す。6 シーンともすべて自転車は等速で直進しており、自転車の進行方向に縦軸、進行方向に対し直角方向 (右方向を正) に横軸を設定し、移動障害物の軌跡をプロットした。単位は cm であり、過去 30 フレームの位置算出結果を用いて現在位置を推定している。図中の◇印が本アルゴリズムで求めた移動障害物の位置算出結果であり、点線がステレオ視により計測した実測値である。移動障害物はすべて等速で直進している。シーン 6 に関しては、車の動きに回転運動が加わっているが、車の旋回半径が大きいため、自転車に最も近い部分の動きはほぼ直線で近似できる。このため、処理を行った最初のフレームと最後のフレームでステレオ画像間の対応点を人間が対話的に指示し、求めたステレオ計測値を直線で結ぶことにより移動障害物の軌跡の実測値を設定している。算出結果及び実測値の軌跡は、自転車に搭載されたステレオカメラ座標系を基準に求めているため、自転車から見た移動障害物の相対的な動きが示されている。つまり、シーン 1 では、歩行者が自転車の前を左から右へ横断しているが、自転車が直進しているため、図 22(e) に示すように歩行者の位置は左奥から右手前にかけて斜めに接近する軌跡となる。シーン 2 では、歩行者が交差点を左手前から右奥に斜めに横断しているが、歩行者より自転車の速度が速いため図 23(e) に示すように左奥から右手前にかけて斜めに接近する軌跡となる。シーン 3 では、歩行者が自転車に向かって接近しているため、図 24(e) に示すように自転車に急速に接近する軌跡となる。シーン 4 では、歩行者が自転車の進行方向と同じ向き

表 5: Error ratio

Scene	1	2	3	4	5	6	Ave.
Total number of frames	91	171	41	71	170	131	116
X-direction error(cm)	35	25	32	17	21	46	29
Y-direction error(cm)	113	66	129	65	70	50	81
X-direction error ratio(%)	2.1	1.4	3.2	1.6	1.7	1.9	2.0
Y-direction error ratio(%)	6.6	3.7	10.9	5.7	5.3	2.1	5.7

に歩いているため、自車との相対速度が小さくなり、図 25(e) に示すような軌跡となる。シーン 5 では、自転車が交差点を右奥から左手前に斜めに横断しているため、図 26(e) に示すような右奥から左手前にかけて斜めに接近する軌跡となる。シーン 6 では、車が交差点に右から進入してくるため、図 27(e) に示す右から左へ移動する軌跡となる。このように、各シーンとも 1 点 1 点の算出結果は多少ばらつくものの、ほぼ実測値に沿った結果が得られている。

歩行者や自転車はフレーム毎に形状が変化している。特に、シーン 3 及び 5 では、歩行者や自転車が急速に接近してくるため、画像中で大きさや形状が大幅に変化する。このため、相関法等を用いた従来手法では、フレーム間やステレオ画像間で移動障害物を安定に対応付け、その位置を算出することは困難である。テンプレート内の対象物の形が撮影した画像内の対象物の形と大きく異なる場合、相関法は照合位置を正しく求めることができない。対象物が大きい場合、数十画素程度の照合ずれが発生する可能性がある。この場合、距離計測誤差は、対象物までの距離に対して数十パーセント程度（10 m 先の障害物なら数メートル程度）となり、移動軌跡を安定に得ることが難しい。これに対し、本アルゴリズムは、これらの対応付けが不要なため、シーン 3 及び 5 においても移動障害物の 3 次元位置を安定に求めることができる。

算出した移動障害物位置と実測値との間の誤差を定量的に求めた結果を表 1 に示す。各

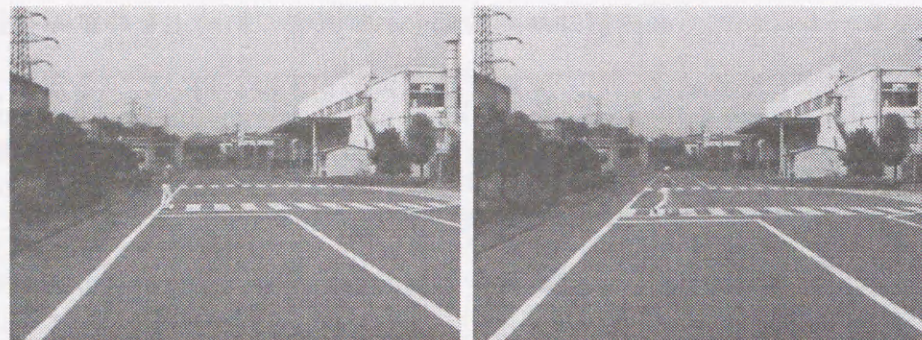
シーンとも、処理を行った全フレームの位置算出結果と実測値との間の誤差の平均を求めている。各フレームにおける実測値は、実測値を示す直線の軌跡上を移動障害物が等速で運動していると仮定して求めている。距離計測にステレオ視を用いているため、移動障害物までの距離が大きくなると誤差も大きくなる。このため、各フレーム毎に距離に対する誤差の比率 $(\frac{|x-x_d|}{y_d}, \frac{|y-y_d|}{y_d})$ を求め、距離に対する誤差率の全フレームの平均を算出した。ここで、 x は左右方向の算出結果、 y は奥行き方向の算出結果、 x_d は左右方向の実測値、及び y_d は奥行き方向の実測値を示す。この結果、全 6 シーンにおける x 方向 (左右方向) の誤差の平均は 29cm、 y 方向 (奥行き方向) の誤差の平均は 81cm、 x 方向の距離に対する誤差率の平均は 2.0%、 y 方向の距離に対する誤差率の平均は 5.7% であった。本実験では、ステレオカメラの光軸を平行に設定したため、ステレオ視が原理的に持つ誤差の範囲は、奥行き方向に長い菱形の形状となる。このため、距離に対する誤差の比率は、奥行き方向 ($y: 5.7\%$) が左右方向 ($x: 2.0\%$) より大きくなる。移動障害物の算出位置が (x_i, y_i) の場合、その存在可能領域は誤差を考慮すると、

$$x_i - \frac{2.0y_i}{100} < x < x_i + \frac{2.0y_i}{100}$$

$$y_i - \frac{5.7y_i}{100} < y < y_i + \frac{5.7y_i}{100}$$

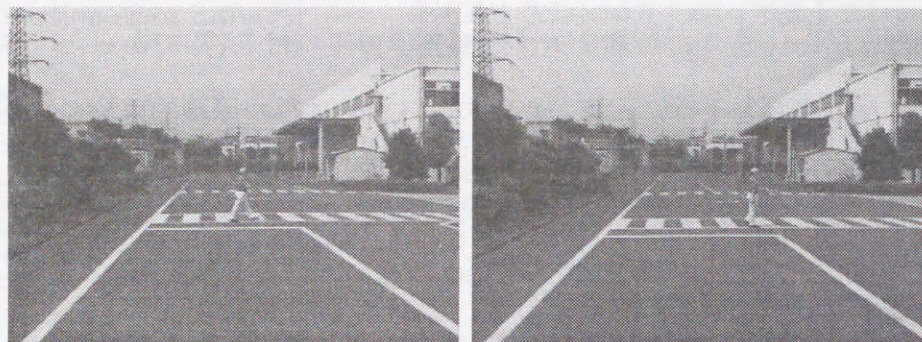
の範囲となるため、この範囲を回避するよう移動車を制御すれば、障害物に衝突せず走行することができる。

現在、移動可能空間マップの作成に 1 秒/フレーム、移動障害物領域の抽出に 7.8 秒/フレームの処理時間がかかっており、自律移動車に本アルゴリズムを搭載するには、これらの処理の実時間化が必要となるが、移動可能空間マップの作成はテクスチャマッピングハードウェアを用いることにより、また、移動障害物領域の抽出はアルゴリズムの最適化及び DSP ボードを用いることによりビデオレート化は容易である。



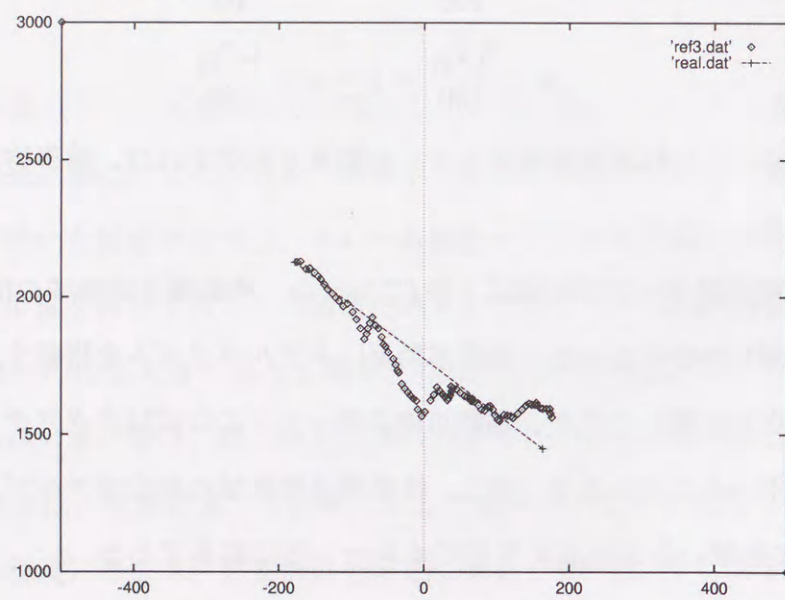
(a) frame 0

(b) frame 30



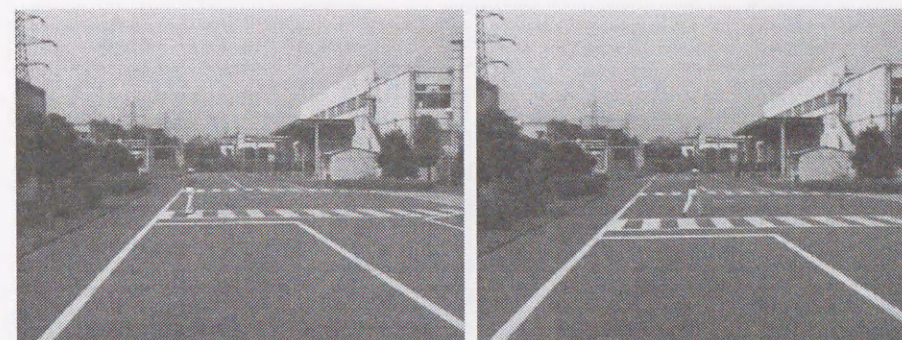
(c) frame 60

(d) frame 90



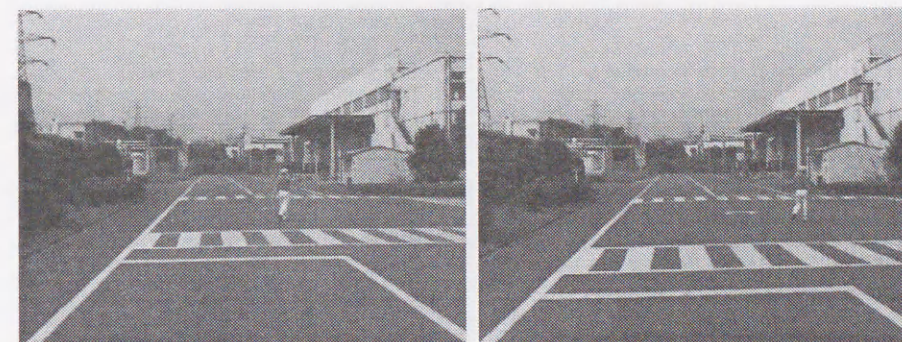
(e) Obstacle location (cm)

☒ 22: Scene 1 (Pedestrian)



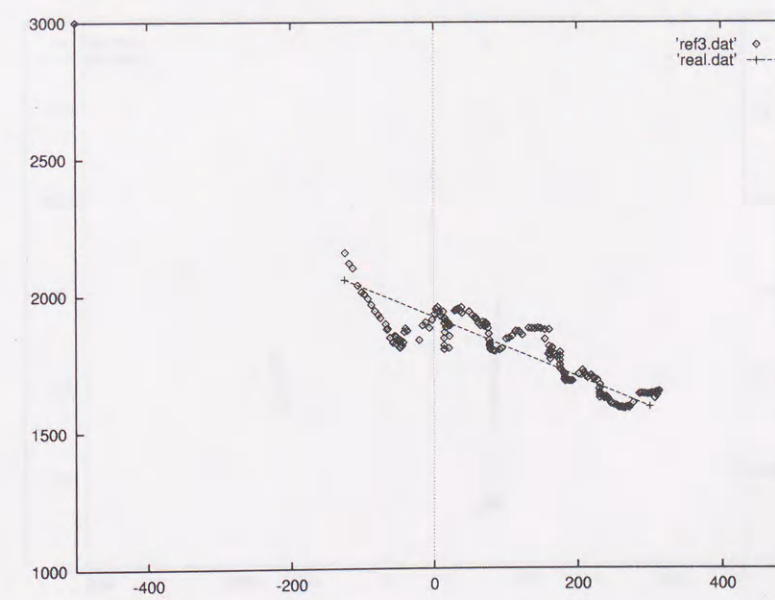
(a) frame 0

(b) frame 57



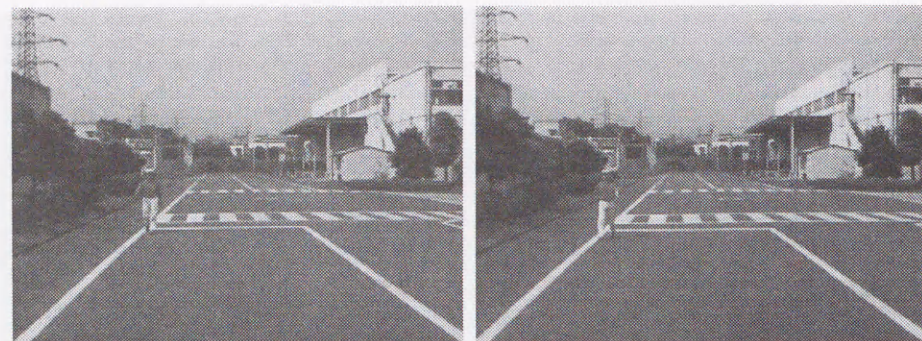
(c) frame 114

(d) frame 170



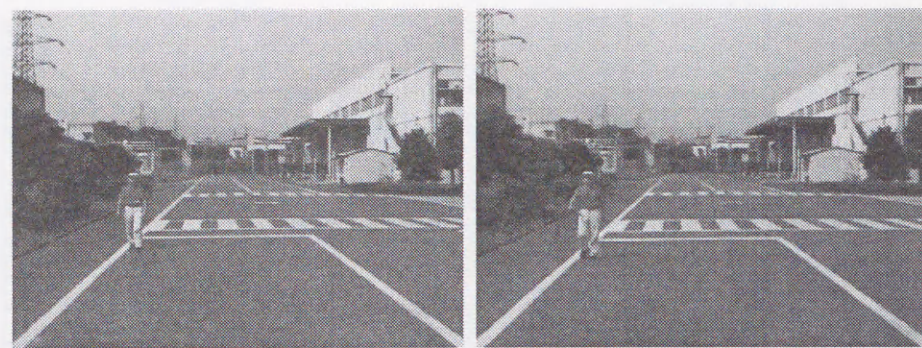
(e) Obstacle location (cm)

☒ 23: Scene 2 (Pedestrian)



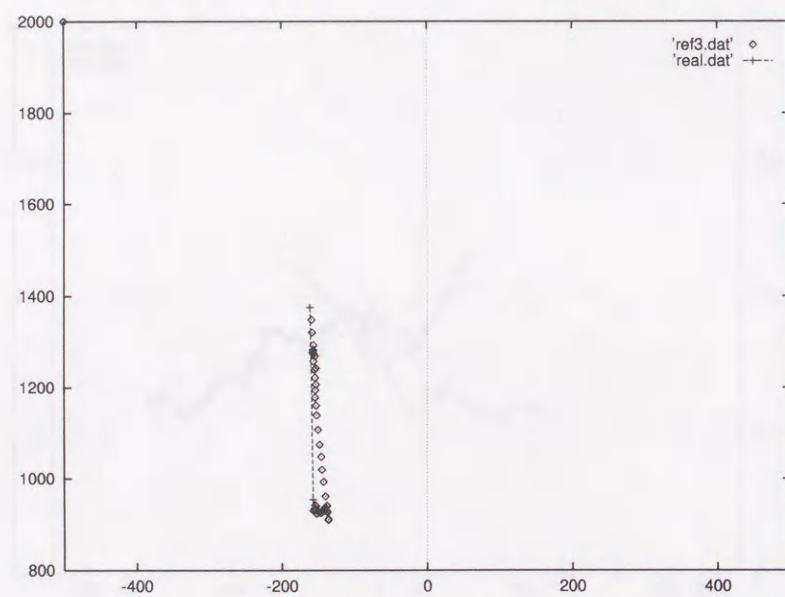
(a) frame 0

(b) frame 13



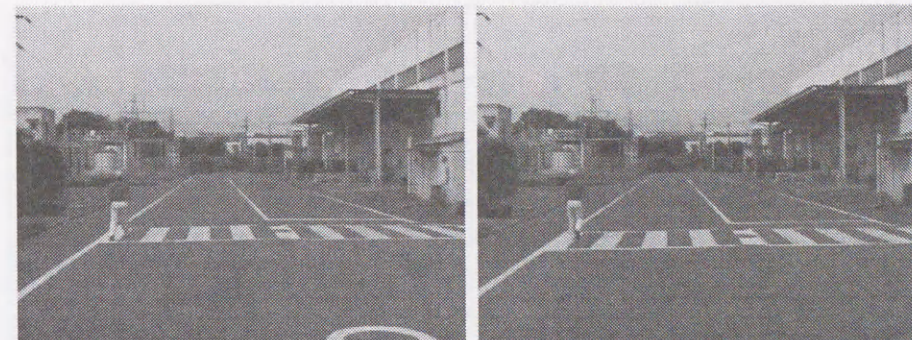
(c) frame 26

(d) frame 40



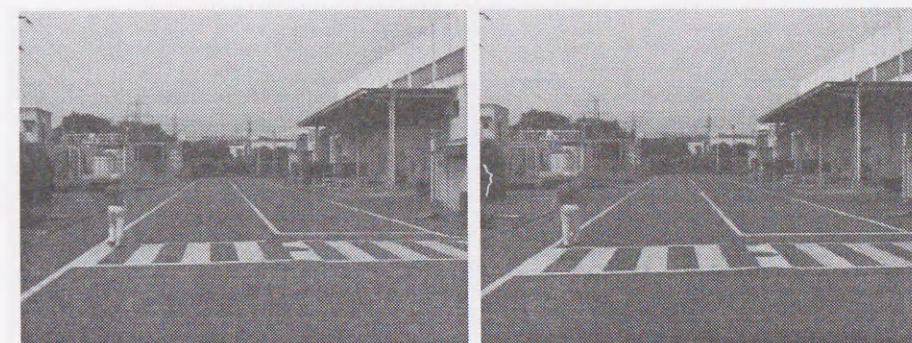
(e) Obstacle location (cm)

図 24: Scene 3 (Pedestrian)



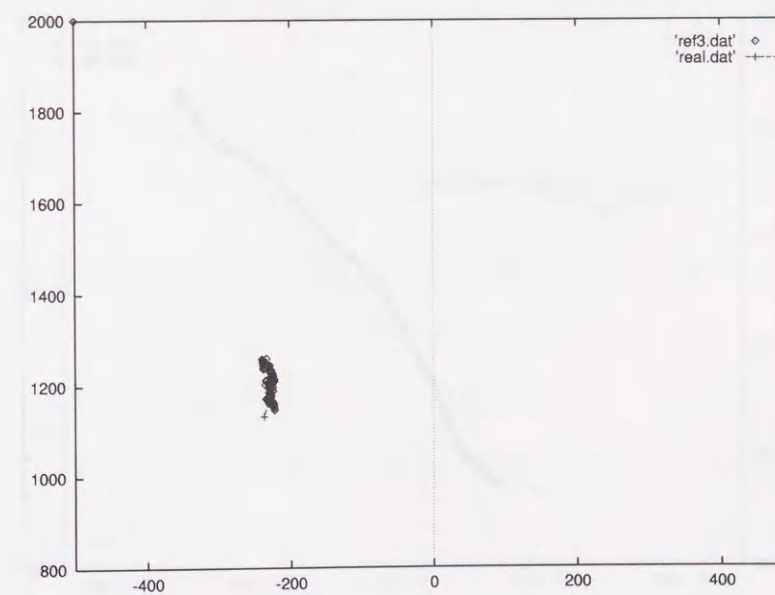
(a) frame 0

(b) frame 24



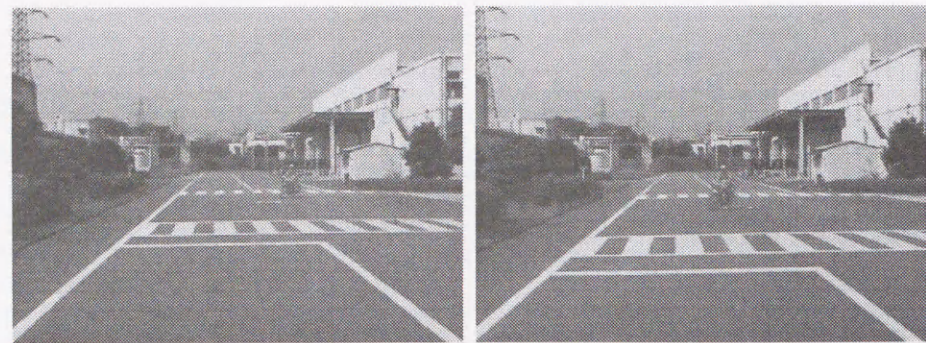
(c) frame 48

(d) frame 70



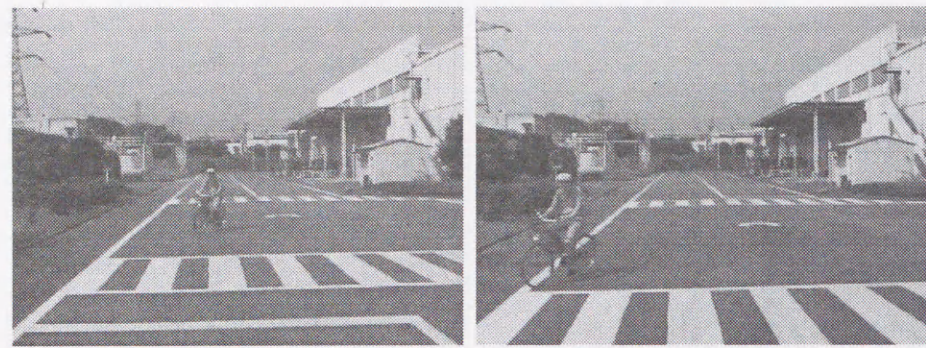
(e) Obstacle location (cm)

図 25: Scene 4 (Pedestrian)



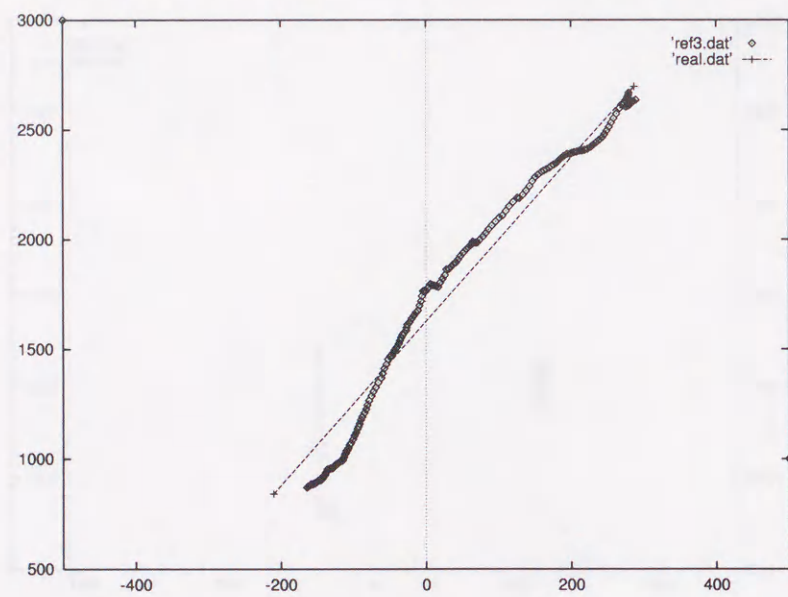
(a) frame 0

(b) frame 56



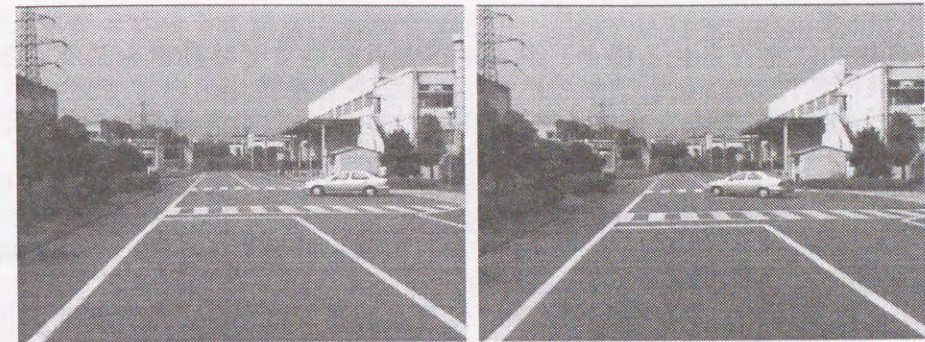
(c) frame 112

(d) frame 169



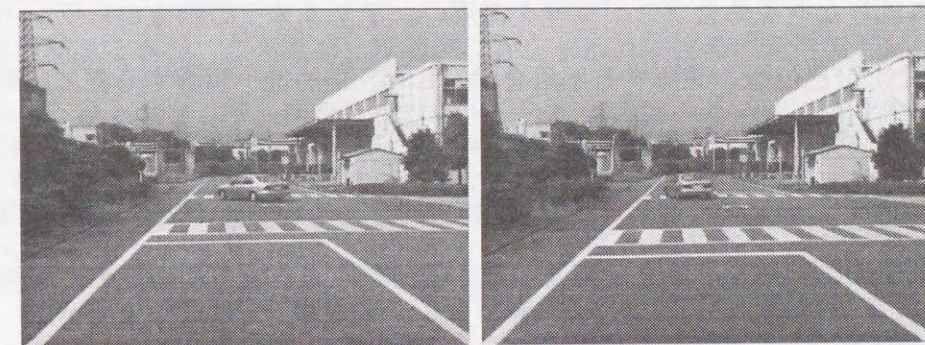
(e) Obstacle location (cm)

⊠ 26: Scene 5 (Bicycle)



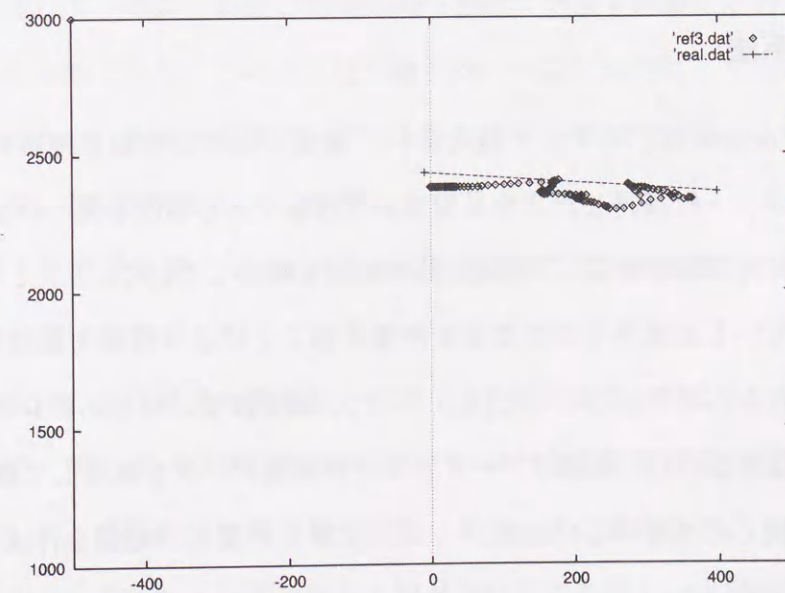
(a) frame 0

(b) frame 43



(c) frame 86

(d) frame 130



(e) Obstacle location (cm)

⊠ 27: Scene 6 (Automobile)

4 手続き型地図による誘導アルゴリズム

移動車を自律的に走行させる場合、到達すべき目的地まで移動車を誘導するための手法が必要となる。誘導に際し、ジャイロ等の内界センサからの情報のみを用いると誤差が蓄積されるため、外界センサからの情報が必要となる。しかし、誘導線を辿ったり、予め設置しておいた特殊なマークを検出することで自己位置を確認する手法は、誘導線やマークの敷設工事が必要なため、走行経路の追加・変更が容易ではない。走行経路の追加・変更に対応するためには、環境に手を加えることの無い誘導手法が必要となる。このため、環境内に既存な対象物をランドマークに用い、TVカメラの画像からランドマークを検出することで自己位置を確認する手法を検討した。目的地へ誘導する際に、周囲の環境に関する知識を持たないと、試行錯誤的に動き回って目的地を探索するため、移動効率が非常に悪くなる。このため、変化の少ない静的環境においては、周囲の環境に関する知識(地図)を積極的に利用することが望ましく、環境の地図を獲得するための研究が盛んに行なわれてきている。

4.1 従来手法

1970年代に Moravec は、ステレオ視を用い、環境の3次元情報を獲得する手法を提案した[2]。移動ロボットは獲得した3次元情報から移動できる場所を見つけ出し、そこへ移動して新たな3次元情報を獲得し、環境全体の地図を構築していく。また、Tsuji と Theng は床面から伸びている垂直エッジの位置を計測することにより移動可能な空間を検出し、環境の地図を作成する研究を行なった[3]。さらに、超音波センサとステレオ視の結果から環境の地図を作成する[31]、距離のデータとカメラ画像データを統合して環境を解釈する[4]、画像中で注視した点を中心に垂直エッジの位置を計測し、地図を作成する[5]といった研究も行なわれている。

しかし、これらの手法により得られる地図は、空いている領域の位置情報が単に記述されている、または、どの領域に何があるかが記述されているものであり、移動ロボット

の誘導に必要な情報が直接導き出せる形にはなっていない。例えば、環境内の自己位置がわかれば、移動ロボットの誘導に有利であるが、環境内に存在する対象物の位置情報のみを記述した地図から自己位置を算出するためには、地図作成の際に行なった環境計測と同じ処理により、周囲の対象物の位置情報をまず獲得する必要がある。そして、獲得された対象物の位置情報と地図に記載された対象物の位置情報とを照合し、どの場所に該当するか調べねばならない。移動ロボットを目的地へ誘導するための地図という観点に立った場合、誘導に必要な情報を使い易い形で記述した地図を用いる方が、空き領域、ドア、壁等の占める位置情報を記述した地図を用いるより、情報の切り出しが容易であり、誘導に要する処理が軽減できる。このため、環境内に存在する対象物の位置情報だけではなく、目的地へ向かう途上のどこで、何を行なうかという位置確認のための手続き情報を記述した地図(手続き型地図)を用い、移動ロボットを効率良く目的地へ誘導するアルゴリズムを提案する。

後年、同様な考え方に沿った Annotated Map[29]がCMUの自律移動車NAVLABに搭載された。この地図には、移動経路中のどこで、何の処理を起動するかという手続き的なデータが記述されているが、設定した目的地へ移動ロボットを導くために各処理要素を統合することが主目的であり、このような手続き的な地図を如何にして作成するかに関しては触れられていない。人間が制御プログラムを作成する要領で構築すると、多数の走行経路が存在する際、地図作成のコストが膨大となる。

本章で提案する位置確認アルゴリズム[64][65][66][67]は、位置確認に必要なデータ(地図)を簡便に作成する機能も有している。データの作成手順は以下の通りである。まず、自律走行する前に、走行環境の簡単なレイアウトを記述した地図から、目的地までの走行経路を設定する。そして、交差点や曲がり角といった自己位置同定が必要な位置確認地点を定める。次いで、遠隔操作で移動車や移動ロボットを位置確認地点まで誘導し、ステレオ画像を収集する。このステレオ画像を人間が対話的に処理し、位置確認に用いるランドマークの種類やその抽出手順等を記述した手続き型地図を作成する。位置確認地点でのステレオ画像が得られれば、位置確認のための処理手順を記述した手続き型地図が対話処理により簡単に作成できるため、地図の作成や走行経路の変更に伴う地図の変更が容易で

ある。

自律走行時は、各位置確認地点において手続き型地図に記述された処理手順に基づきランドマークを検出する。そして、検出されたランドマークと移動車との間の位置関係をステレオ視により求め、走行経路上の自己位置を算出する。

以下で、手続き型地図の詳細を述べるとともに、移動環境として原子力発電所内部を想定した際の手続き型地図の生成手法、及び原子力発電所内部を模擬した環境で行なった実験結果を示す。

4.2 手続き型地図

環境内の自分の位置が既知であれば目的地への誘導に有利であるが、常時、正確な位置を知る必要はないと思われる。人間が目的地まで道を歩く場合を考えると、代表的な建物や看板等で自己位置を推定し、曲がるべき交差点等を識別して目的地へ近付いて行く。つまり、道沿いに存在するすべての建築物等を見ながら常に自分の位置を認識し、歩いている訳ではなく、ポイントとなるランドマークを用いて要所要所（曲がり角や交差点等）で位置の確認を行なう。そして、その間は舗道に沿って歩いたり、目標物に向かって歩いたりしていると考えられる。手続き型地図はこの考えに基づいて構築されており、以下の3つの要素から構成されている。ここで、位置確認地点とは、曲がり角や交差点等正確な自己位置が必要な地点のことである。

- (1) 走行経路
- (2) 自己位置確認手順
- (3) 位置確認地点間の移動手法

(1) は走行経路上での出発地点から目標地点までの位置確認地点の系列で表す。

(2) は各位置確認地点で環境内の自己位置を計測するための手順を示している。この手順に従い、位置確認地点で撮像した画像から目標物を抽出し、自己位置を算出する。このため、目標物の種類やその抽出手順等が記述される。表6に代表的な目標物を示す。色や

表 6: Example of target

Outdoor	Signpost, Signboard, Intersection
Indoor	Door, Intersection, Signboard Equipments in Factory or Power Plant (Meter, Switch box, etc)

輪郭形状を用いてこれらの目標物を抽出する。

(3) は各位置確認地点間でどのような走行手法を採るかを示している。位置確認地点間が、道路や廊下等の通路である場合、通路端に沿って走行すれば、経路に沿って次の位置確認地点まで移動ロボットを誘導することができる。また、位置確認地点間が広場や広いフロアである場合には、次の位置確認地点周辺に設定した目標物へ向かって接近すれば、次の位置確認地点へ到達することができる。このため、

- 沿目標物走行
- 向目標物走行

という2つの走行手法を用意した。位置確認地点間が通路であれば、通路端に沿って走行する沿目標物走行を、また、広いエリアであれば目標物に向かって接近する向目標物走行を用いる。沿目標物走行を用いる際には、通路の幅及び、カメラと床面との間の距離等のカメラパラメータを付加しておく。また、向目標物走行を用いる際には、向かっていく目標物の抽出手順、及び画像中心等のカメラパラメータを付加しておく。

手続き型地図は(1)～(3)の情報をグラフにより表現している。このグラフは、ノードの位置及びノード間を結ぶエッジの接続状態で(1)の情報を表している。また、ノードの属性で(2)の情報を、さらにエッジの属性で(3)の情報を表している。例えば、図28の移動環境において、スタート地点から目的地であるドアの手前までの走行経路を矢印で示すように設定し、途中の交差点、曲り角及び目的地の手前を位置確認地点 a, b, c として定めたとする。この際、図29に示すような有向グラフとして手続き型地図は表現される。つ

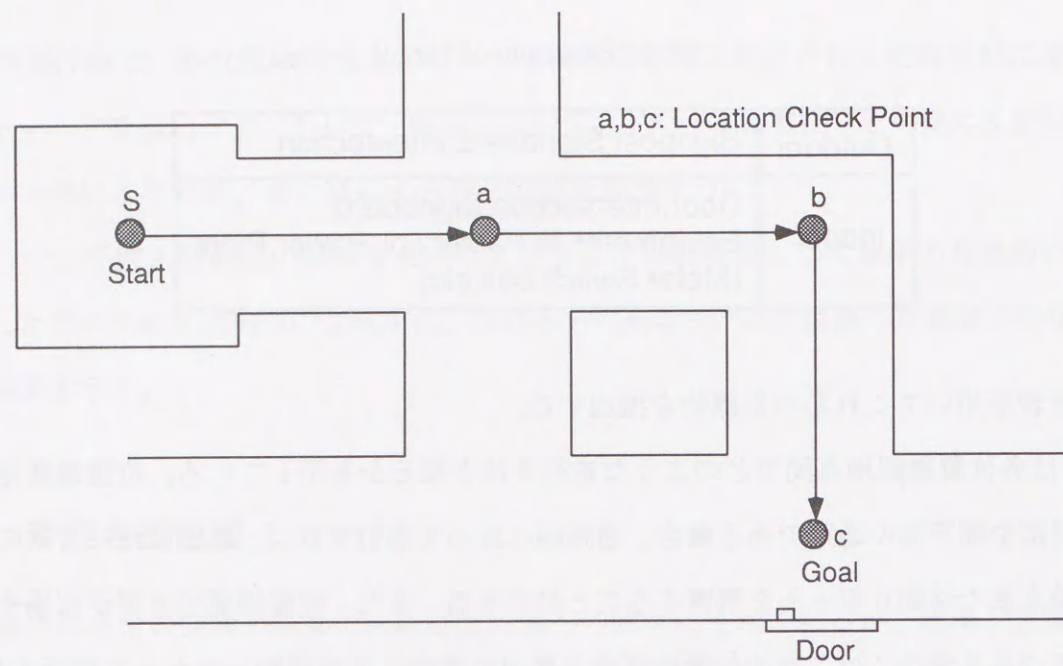


図 28: Example of environment

まり、環境内に設定した座標系に於けるスタート地点及び各位置確認地点 a,b,c の座標値 (x_s, y_s, z_s) , (x_a, y_a, z_a) , (x_b, y_b, z_b) , (x_c, y_c, z_c) と、 $s \rightarrow a \rightarrow b \rightarrow c$ なる接続関係が (1) の情報を示し、表 7 に示すノード属性及びエッジ属性が (2),(3) の情報を示す。

このグラフ表現された手続き型地図を、自律走行前、遠隔操作で移動ロボットを走行経路に沿って動かすことにより、作成しておく。自律走行時、移動ロボットは、各位置確認地点においてノードに記述された手順に従い自己位置を計測し、経路からのずれを算出する。次いで、次の位置確認地点に対応するノードの方向と移動ロボットが現在向いている方向から旋回角度を算出してターンする。そして、エッジに記述されている走行手法を用い、位置確認地点間に対応する距離だけ前進する。スタート地点から目的地までこの手法を繰り返し、移動ロボットを誘導する。

表 7: Graph attribute

	Attribution
Node	Self-location Measurement Procedure (Target Configuration, Target Extraction Method)
Edge	Movement Style (Path Following, Target Tracking)

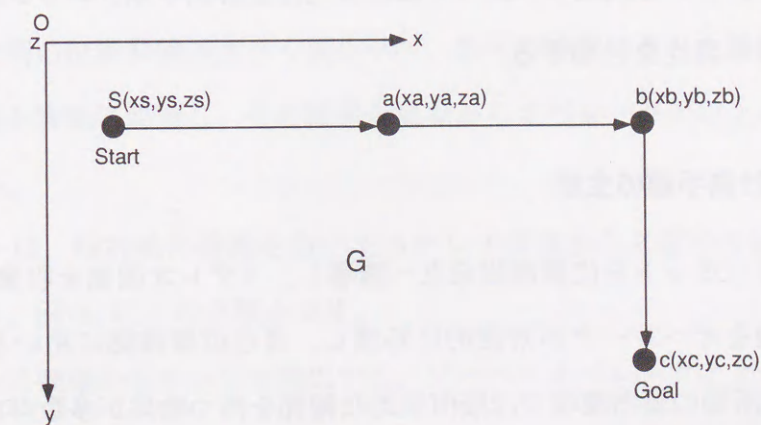


図 29: Graph structure

4.3 手続き型地図の作成手法

手続き型地図は、自律走行前に移動ロボットを走行経路に沿って遠隔操作で動かし、収集したステレオ画像から対話的に作成される。オペレータは位置確認地点で入力したステレオ画像を処理し、4.2章で述べた自己位置確認手順を作成する。また、位置確認地点間を誘導する間に入力した画像を見て、位置確認地点間の走行手法を決定する。

例えば、原子力発電所等の屋内環境を移動するための手続き型地図を作成する場合、位置確認地点で収集したステレオ画像を処理し、メータや配電盤といった目標物を設定する。また、位置確認地点間は通路で結ばれているため、沿目標物走行を位置確認地点間の走行手法として選択する。

自律走行時には、この手続き型地図に記述された手順を自動的に実行し、自己位置を確認する。

4.3.1 位置確認地点の設定

曲り角や交差点、及び段差の手前等に位置確認地点を設定する。そして、位置確認地点をノードとしたグラフを生成し、ノードには自己位置計測手順、エッジには位置確認地点間の走行手法等の属性を付加する。

4.3.2 自己位置計測手順の生成

遠隔操作で移動ロボットを位置確認地点へ誘導し、ステレオ画像を収集する。そして、このステレオ画像をオペレータが対話的に処理し、自己位置確認に用いる目標物を設定する。原子力発電所等の屋内環境では幾何学的な輪郭を持つ物体が多数存在するため、オペレータは楕円、円、四角形、平行線、角点といった特徴を、収集したステレオ画像から抽出し、これらの組合せで目標物を記述する。ただし、特徴の組合せを選択するに当たり以下の2点に留意する。第1点は、シーンの中で一意に特定できる特徴の組合せを選ぶことである。図30において、左側のメータを目標物として選択し、1つの円で記述すると、右側のメータの円を左側のメータとして誤って抽出する可能性がある。この問題を回避す

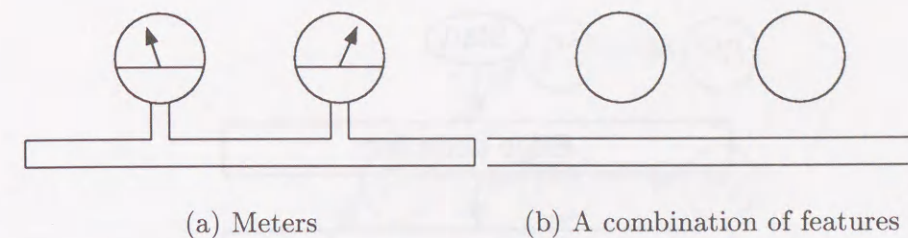


図 30: Target detection

るため、オペレータはシーン中で唯一の目標物となるよういくつかの特徴を組み合わせる。例えば、図30(a)の場合、図30(b)に示す2つの円と1つの平行線の組合せで目標物を記述する。このように、楕円、円、四角形、平行線、角点といった個々の特徴は、単体では画像中で多数検出される可能性が高いため、複数の組み合わせで1つの目標物を表現する。

第2点は、視点の変化に対し安定な特徴の組合せを選ぶということである。つまり、視点が変わっても目標物を構成する特徴間の位置関係及び特徴の形状があまり変化しないことが望まれる。この条件は、目標物が壁面上にあり、ロボットから十分離れていればほぼ満足される。例えば、図30に示すメータが前方正面の離れた壁面上にある場合、視点が変わっても特徴間の位置関係は大きく変化せず、各々の特徴の歪みも無視できる。収集したステレオ画像を実際に処理し、その結果を見ながらオペレータが以上の2点を満たす目標物を選択する。

移動ロボットは、輪郭線の情報を用いてステレオ画像から目標物を抽出し、自己位置の計測を行なう。図31にこの手順を示す。

まず、ステレオ画像からエッジを抽出する。ソーベルオペレータによる空間フィルタリング → 2値化 → 細線化 → 微小エッジ除去の順で処理を行う。オペレータは2値化しきい値や除去エッジの最大長等を対話的に繰り返し設定し、最適な画像処理パラメータを手続き型地図に記述する。

次いで、分岐点や屈曲点でエッジを分割する。これにより得られた各線分（ラインセグメント）に対し、端点の位置、長さ、端点間距離、端点を結ぶ直線の傾き、及びチェー

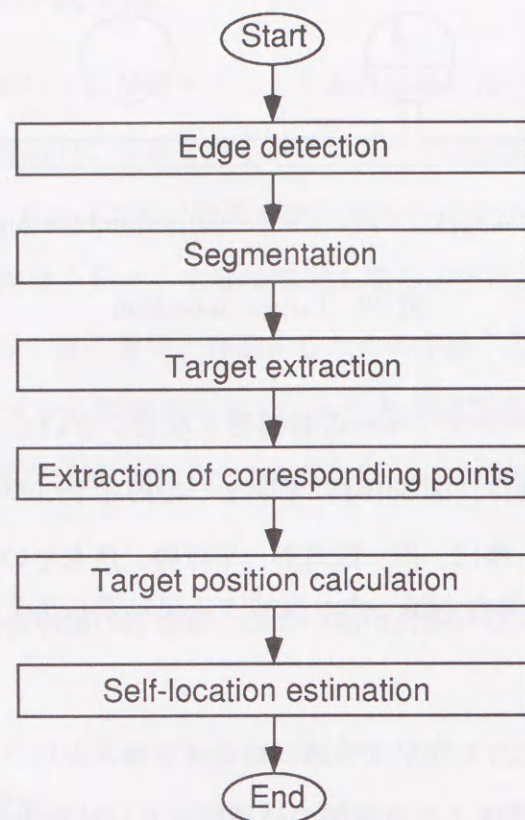
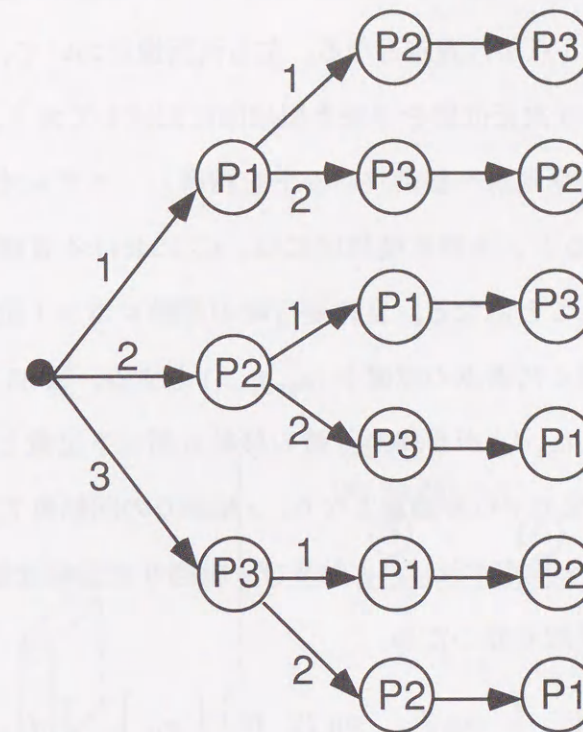


図 31: Self-location measurement blockdiagram

ンコードを算出する。これらの情報を用い、楕円、円、四角形、平行線、及び角点といった特徴を画像から抽出する。

複数の特徴の組み合わせからなる目標物を効率良く抽出するため、各特徴の形状パラメータ（画像中の位置やサイズのデータ。楕円の場合には、中心、傾き、長軸長、及び短軸長である）、及び特徴の抽出順序を示す手順木を手続き型地図に記述しておく。図 32 に手順木の例を示す。この例では、目標物は 3 つの特徴 (P_1, P_2, P_3) の組合せからなっており、手順木の各枝には、抽出順序が与えられている。もし、ノードの示す特徴が検出されたならば、抽出順位が一番高いエッジで結ばれている子ノードを次に抽出する特徴として選択する。親ノード及び選択した子ノードが示す特徴の形状パラメータには、特徴の画像中の位置と大きさが記述されているため、選択した特徴の探索領域を画像中で制限し、特徴探索に要する時間を短縮する。この探索領域の中で選択した特徴の形状パラメータと



P1,P2,P3 : Feature

図 32: Procedure tree

合致した特徴を抽出する。例えば、図 30 に示すメータにおいて、 P_1 を左の円、 P_2 を右の円、 P_3 を平行線に設定すると、左の円がまず抽出される。次いで、左の円と右の円の位置関係及びサイズから右の円の探索領域が設定され、この探索領域中で右の円が抽出される。そして、右の円と平行線の位置関係及びサイズから平行線の探索領域が設定され、その内部で平行線が抽出される。もしノードに示す特徴が検出されなければ、親の次の抽出順位のエッジを選択する。手順木は各特徴の重要度を例えば $P_1 \rightarrow P_2 \rightarrow P_3$ というようにオペレータが設定すると自動的に作成される。また、実際にステレオ画像から目標物を構成する特徴を対話的に抽出することにより各特徴の形状パラメータを算出する。

目標物を構成する各特徴の形状パラメータ、及び目標物抽出の手順木を求めた後、目標物中の代表点の 3 次元位置をステレオ視により求める。代表点としては、特徴毎に予め定めておいた点（四角形なら 4 頂点、楕円、円、平行線ならその重心、角点なら角点を構成

する直線の交点)を用いる。例えば、図30に示すメータの場合、左の円の重心、右の円の重心、平行線の重心の3点が代表点となる。左右両画像において、これら代表点をそれぞれ対応付け、算出した3次元位置を手続き型地図に記述しておく。

遠隔操作により位置確認地点へ移動ロボットを誘導し、ステレオ画像を撮影した際のカメラ座標系を C_t とすると、手続き型地図には、 C_t における目標物中の代表点の位置 (x_{ti}, y_{ti}, z_{ti}) が記述されたことになる。自律走行時の移動ロボット位置でのカメラ座標系を C とし、 C における同じ代表点の位置を (x_i, y_i, z_i) とする。図33に示すように、 C_t から C への平行移動量 (d_x, d_y, d_z) が自律走行時の移動ロボット位置と手続き型地図作成時の移動ロボット位置との間の平行移動量となり、 z 軸回りの回転角 T_z が回転量となる。平坦な場所に位置確認地点を設定すれば、 x 軸及び y 軸回りの回転は無視できる。 d_x, d_y, d_z 、及び T_z に関しては次式が成り立つため、

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \begin{bmatrix} \cos T_z & \sin T_z & 0 \\ -\sin T_z & \cos T_z & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_{ti} \\ y_{ti} \\ z_{ti} \end{bmatrix} + \begin{bmatrix} d_x \\ d_y \\ d_z \end{bmatrix} \quad (35)$$

目標物上に2点以上代表点が存在し、その3次元位置が得られれば、位置確認地点で手続き型地図を作成した位置からの平行移動量及び回転量が算出できる。目標物上に2点以上代表点を確保するため、楕円、円、平行線、角点等代表点が1点しか取れない特徴は、必ず複数個組み合わせて用いる。手続き型地図作成時には、この点を考慮し、対話的に目標物を構成する特徴を選択する。

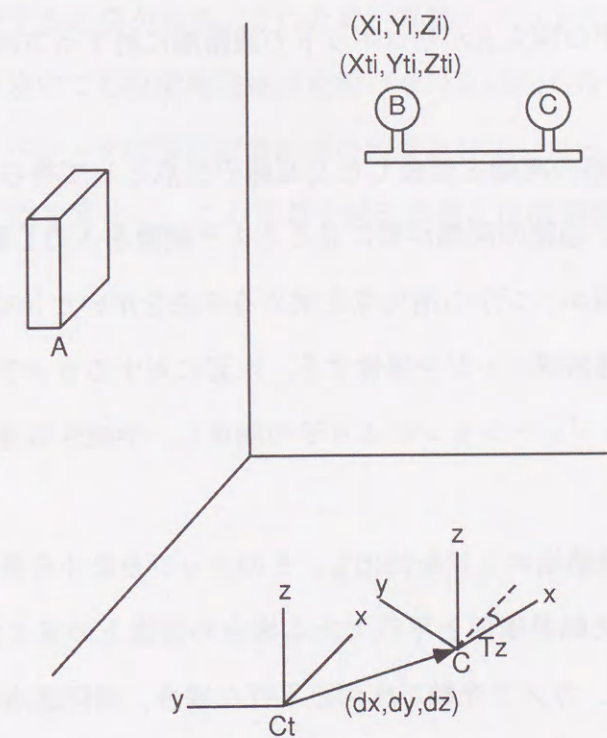


図 33: Self-location measurement

4.3.3 位置確認地点間の走行手法の設定

遠隔走行中に位置確認地点間で撮影した画像を見て、沿目標物走行手法、または、向目標物走行手法のいずれかを手続き型地図に設定する。原子力発電所等の屋内環境では、走行路が廊下等の通路である場合が多い。このため、沿目標物走行手法が位置確認地点間の走行手法として設定される。

以下で、沿目標物走行手法の詳細を説明する。通路端に対する移動ロボットの方向と距離が頻繁に得られれば、通路端に沿って走行するようステアリングを制御することができる。このため、画像中の消失点からロボットの通路端に対する方向と距離を算出する手法を提案する。

一般に、消失点は通路の両端を延長した2直線の交点として得られるが、通常のレンズでは画角が狭いため、通路の両端が常に見えるよう画像を入力し続けることが難しい。このため、一方の通路端エッジから消失点を求める手法を用いた [68]。カメラは下向きにし、できるだけ近くの通路端エッジを撮像する。床面に対するカメラのチルト角、ロール角、及び高さは、キャリブレーションにより予め測定し、手続き型地図のエッジ属性に記述しておく。

まず、入力画像から通路端エッジを抽出し、そのエッジを最小自乗法により直線近似する。次いで、カメラの光軸が床面と平行である場合の画像上の見え方に直線式を変換する。図 34 に示すように、カメラ光軸が床面と平行な場合、無限遠点は画像中心を通る水平線上に投影される。このため、変換後の直線と画像中心を通る水平線との交点が消失点となり (図 35 参照)、通路端に対するロボットの方向 TH と距離 X が次式より得られる。

$$TH = \tan^{-1} \frac{Q_x}{f} \quad (36)$$

$$X = L \frac{\cos TH}{\tan k} \quad (37)$$

ここで、 L はカメラの高さ、 f は焦点距離、 Q_x は画像上の消失点の x 座標値、 k は通路端エッジを表す直線と x 軸の成す角度である (図 34, 35 参照)。

手続き型地図のエッジの属性に通路端との間に保持する距離 D を記述しておき、上述

の手法で求めた TH 及び X を用いて $X = D$ 、 $TH = 0$ となるようロボットの方向を制御する。

本手法は、通路端からの距離と方向を毎計測時、独立に得るため、通路が伸びている方向に対し横方向の誤差は蓄積されない。このため、計測間隔を短くすれば、通路端との間の距離を一定に保って走行することができる。ただし、通路が伸びている方向に対する位置情報は本手法では得られず、移動ロボットのエンコーダから求める必要があるが、エンコーダ情報の誤差は蓄積されるため、位置確認地点における位置計測で一定以上、誤差が蓄積されるのを回避する必要がある。このため、移動ロボットのエンコーダの精度によっては、直線の通路の途中でも位置確認地点を設けなければならない場合が生じる。手続き型地図作成の際、オペレータは自己位置計測の精度及びエンコーダの精度から次に位置確認が必要な限界の距離を算出し、この距離を越える前に位置確認地点を設定する。

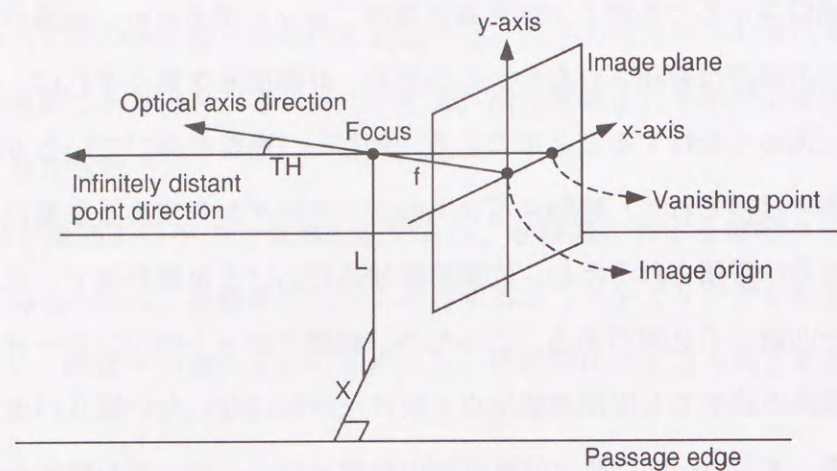


図 34: Infinately distant point projection to the image plane

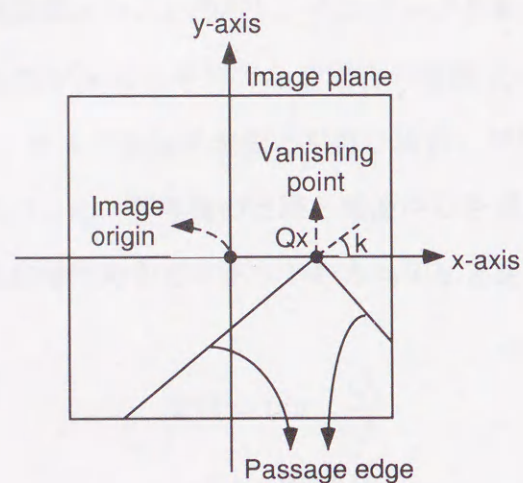


図 35: Passage edge direction

4.4 実験結果

4.4.1 位置確認地点における自己位置計測精度

表 8,9 に自己位置計測の精度を見積もるための実験結果を示す。

表 8 は 2m、表 9 は 4m 離れた目標物を用いている。手続き型地図を作成した位置の周辺 (4 地点) に移動ロボットを置き、手続き型地図作成位置からの位置ずれ量をそれぞれ計測した。目標物は 2 つの四角形の組であり、自己位置の算出にはそれらの角点 (合計 8 点) を用いた。また、位置ずれ量の真値はレーザ測距計により計測した。

4m 離れた位置の目標物を用いた場合の方が、2m の場合よりも誤差が大きくなっているが、これはステレオ視の位置計測誤差が距離に応じて大きくなるためである。図 36 に示すように、対応点が 1 画素左右にずれした場合に生じる誤差の領域は斜線部分に示す菱形の領域になるが、この領域は対象物までの距離が大きくなる程広がっていく。このため、自己位置計測に用いる各点の位置計測誤差が大きくなり、離れた目標物を用いる程、移動量に含まれる誤差が大きくなる。自己位置の算出には複数個の点を用いるが、移動ロボットからの位置の違いにより、1 画素のずれに対する誤差領域の大きさや形状はそれぞれ異なっている。このため、目標物までの距離と自己位置計測誤差との間の関係を一概に求めることはできないが、本実験で用いた点の配置では、1 画素対応付けがずれると、表 8、9 に示す最小値から最大値の範囲で自己位置計測値が変化する。実際の移動量計測結果は、すべてこの範囲に入っており、1 画素のずれにより生じるステレオ誤差の影響の範囲内に精度が収まっていることがわかる。本実験では、位置計測の誤差が x 方向、y 方向ともに目標物までの距離に対して 3% 程度であった。すなわち、2m 先の目標物を用いる場合は 6cm、4m 先の目標物を用いる場合には 12cm 程度の計測誤差が含まれる可能性がある。これは、移動のための位置確認としては一般に十分な精度である。要求される位置精度は位置確認地点により異なるが、高い位置精度を必要とする位置確認地点では、誤差の範囲を考慮してなるべく近い目標物を用いる必要がある。

表 8: Experimental result of self-location measurement accuracy (4m)

		Result value	Real value	Error	$\frac{\text{Error}}{\text{Distance}}$	Minimum	Maximum
Pos.1	X-Translation	367.4 mm	289.3 mm	78.0 mm	1.95 %	282.8 mm	464.2 mm
	Y-Translation	-95.1 mm	2.2 mm	97.3 mm	2.43 %	-617.4 mm	304.3 mm
	Z-Rotation	1.2 deg	0.1 deg	1.11 deg		-4.9 deg	9.3 deg
Pos.2	X-Translation	370.4 mm	307.7 mm	62.7 mm	1.57 %	270.7 mm	479.3 mm
	Y-Translation	-138.5 mm	-90.7 mm	47.8 mm	1.20 %	-735.1 mm	528.9 mm
	Z-Rotation	2.3 deg	-0.7 deg	3.0 deg		-7.9 deg	11.5 deg
Pos.3	X-Translation	117.0 mm	30.9 mm	86.1 mm	2.15 %	67.1 mm	271.3 mm
	Y-Translation	-102.3 mm	31.2 mm	133.5 mm	3.34 %	-931.2 mm	93.2 mm
	Z-Rotation	1.8 deg	0.24 deg	1.51 deg		-1.1 deg	13.9 deg
Pos.4	X-Translation	262.3 mm	197.5 mm	64.8 mm	1.62 %	197.5 mm	331.2 mm
	Y-Translation	128.0 mm	64.9 mm	63.1 mm	1.58 %	-258.7 mm	290.5 mm
	Z-Rotation	0.8 deg	0.4 deg	0.4 deg		-3.4 deg	5.0 deg

表 9: Experimental result of self-location measurement accuracy (2m)

		Result value	Real value	Error	$\frac{\text{Error}}{\text{Distance}}$	Minimum	Maximum
Pos.1	X-Translation	-300.2 mm	-305.9 mm	5.7 mm	0.3%	-335.5 mm	-258.0 mm
	Y-Translation	-33.6 mm	15.4 mm	49.0 mm	2.5 %	-334.2 mm	182.2 mm
	Z-Rotation	0.8 deg	0.6 deg	0.2 deg		-4.7 deg	8.4 deg
Pos.2	X-Translation	140.4 mm	140.9 mm	0.5 mm	0.04 %	115.0 mm	158.8 mm
	Y-Translation	49.1 mm	69.5 mm	20.4 mm	1.0 %	-130.0 mm	147.2 mm
	Z-Rotation	-2.2 deg	-2.5 deg	0.3 deg		-5.5 deg	3.5 deg
Pos.3	X-Translation	-345.6 mm	-370.0 mm	24.4 mm	1.2 %	-374.9 mm	-313.7 mm
	Y-Translation	1.0 mm	6.0 mm	5.0 mm	0.2 %	-198.0 mm	291.3 mm
	Z-Rotation	-0.4 deg	-0.1 deg	0.3 deg		-8.0 deg	4.8 deg
Pos.4	X-Translation	-579.7 mm	-604.1 mm	24.4 mm	1.2 %	-609.5 mm	-540.8 mm
	Y-Translation	-45.7 mm	3.3 mm	49.0 mm	2.45 %	-322.5 mm	353.7 mm
	Z-Rotation	-0.3 deg	-0.9 deg	0.6 deg		-1.0 deg	6.1 deg

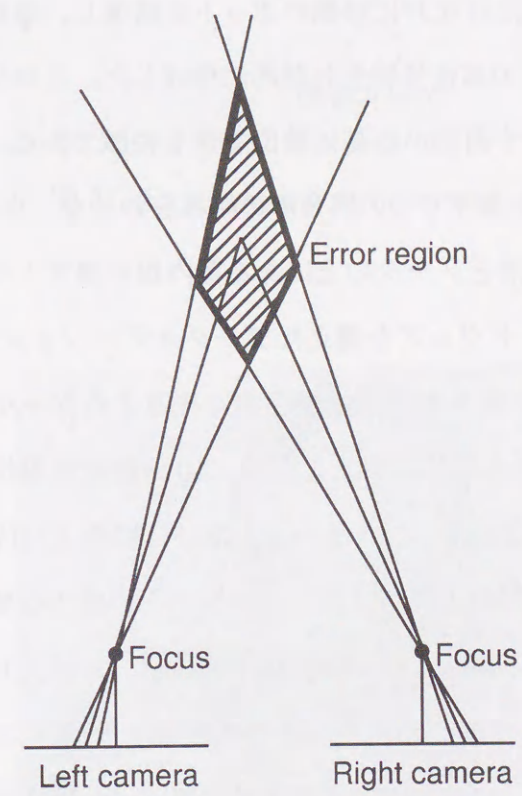


图 36: Error region by corresponding error

4.4.2 走行実験結果

原子力発電所を模擬した環境において、手続き型地図を用いた走行実験を行なった。図 37 に走行環境、図 38 に使用した知能移動ロボット AIMARS を示す [69]。AIMARS は、車体前面にステレオカメラ、側面のカメラ台上に通路端を撮像する単眼カメラを備えている。

スタート地点 P に位置確認地点を設定し、作業場所 Q へ到達するための手続き型地図を作成した。まず、位置確認地点 P に移動ロボットを誘導し、撮影したステレオ画像 (図 39) から P に対するノードの属性情報を対話的に作成した。このシーンでは配電ボックスを表す四角形とメータを表す円形が容易に検出できる特徴である。このため、第 1 の目標物 $T1$ として配電ボックスを表す 2 つの四角形の組み合わせを、また、第 2 の目標物 $T2$ としてメータを表す 2 つの円形とメータの上にある案内板を表す 1 つの四角形の組み合わせを設定した。画像処理ハードウェアを備えたワークステーション TOSHIBA AS4260HM (SUN-4 と同等) [70] 上に手続き型地図を対話的に作成するツールを構築し、 P に対するノードの属性情報を図 39 から実際に得たところ、10 分程度の操作で、画像処理パラメータ、目標物抽出手順、特徴の形状パラメータ、及び目標物上の代表点の 3 次元位置情報が獲得できた。手続き型地図を人間が制御プログラムを作成する要領で構築する場合、数日程度はかかると予想されるが、この対話型地図作成ツールにより、地図作成に要する時間を大幅に短縮できることが確認できた。また、手続き型地図はテキストファイルから成っているため、このシーンでのファイル容量は数百バイト程度にすぎない。環境全体の画像や形状データを用いる従来手法では、少なくとも数メガバイト程度のファイル容量が必要のため、手続き型地図に必要なファイル容量が大幅に少ないことがわかる。

ステレオ計測の結果、 P から $T1$ 、 $T2$ までの距離は約 5m であった。このため、 P における位置確認結果の前後方向の誤差は約 15cm となる。また、移動ロボットのエンコーダの誤差が約 1% であるため、 PQ 間 (4m) を走行する間に生じる誤差は約 4cm となる。このため、 P で位置計測した後、 Q に誘導した際の移動ロボットの前後方向の誤差は最大で ± 19 cm 程度となるが、 Q の周囲には半径 1m 程度の空間があるため、移動ロボットが危

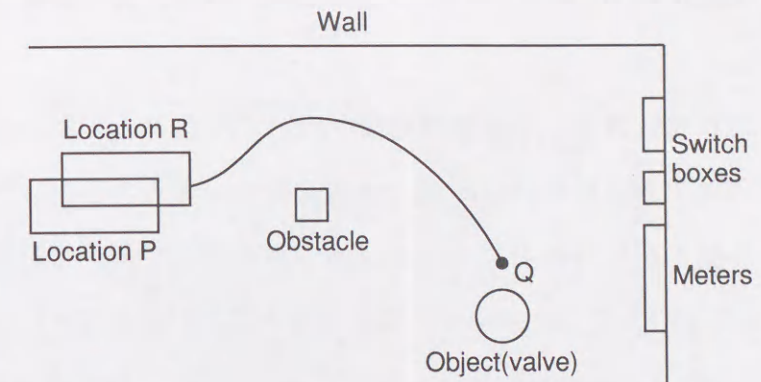


図 37: Experimental environment

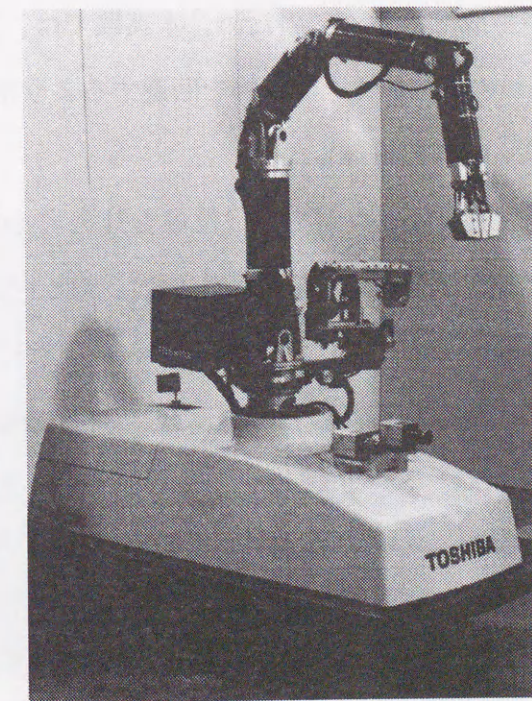
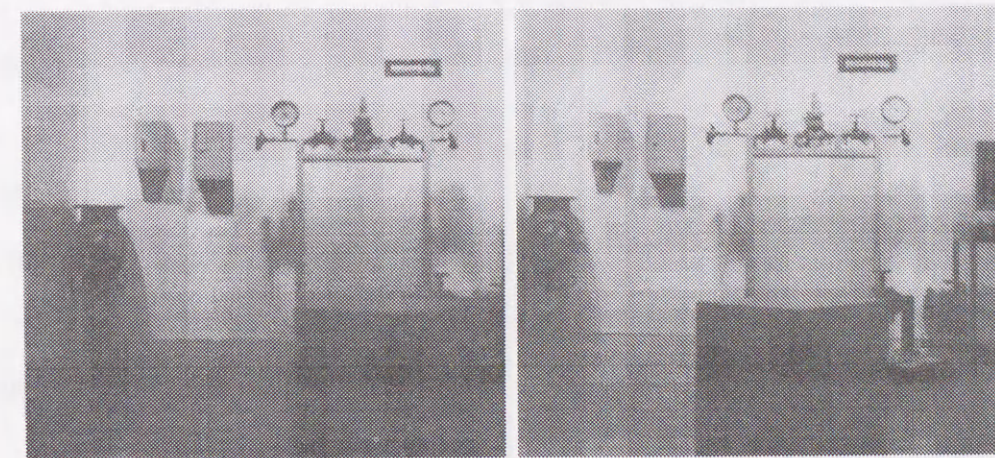


図 38: Mobile robot AIMARS

険な状態に陥る可能性はない。よって、図 37 の走行環境では PQ 間に他の位置確認地点を設定する必要はなく、手続き型地図に PQ を結ぶエッジを付加し、属性情報として、予め計測しておいた単眼カメラのパラメータ、 PQ 間距離 (4m)、及び壁面と Q までの距離 (2m) 等を記述した。

移動ロボットを地点 R に置き、手続き型地図を用いて地点 P との間の位置ずれを算出し、作業場所 Q へ誘導する実験を行なった。地点 R において入力したステレオ画像を図 40 に示す。また、手続き型地図の P に対するノード属性情報を用いて目標物 $T1$ 、 $T2$ を図 40 から抽出した結果を図 41、42 に示す。手続き型地図に記述されている特徴の組合せが正しく抽出されている。この結果を用いて位置ずれを修正し、作業場所まで移動ロボットを誘導するが、本実験で用いた移動ロボットシステムには障害物検出 [71] 及び経路生成 [72] の機能も付加されており、前方に置かれた障害物 ([71] の方法では、障害物の奥行きはわからないが、実験では障害物の奥行きは 1 m 以下と仮定して経路を生成した) を検出し、それを回避する経路が生成できる。このため、本実験では、 R から Q へ移動するに際し、壁から一定距離を保つのではなく、障害物を回避するよう生成された経路からのずれが小さくなるようステアリングを制御した。

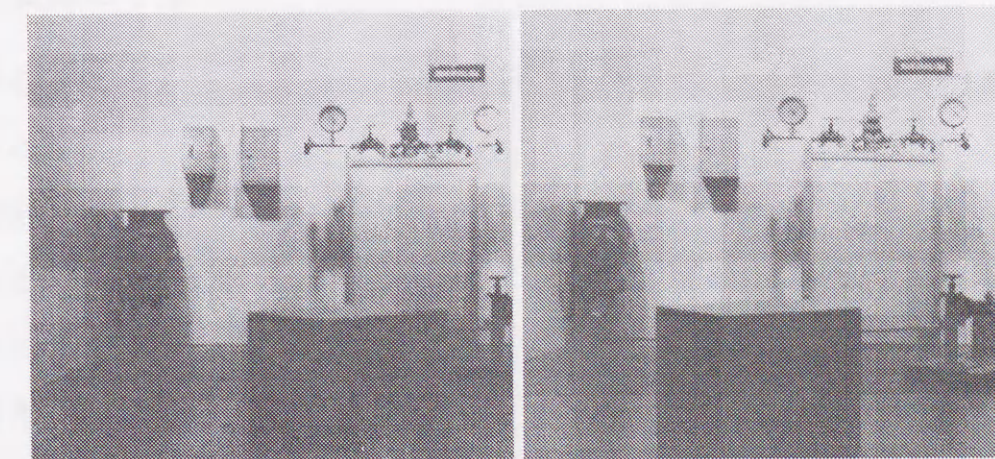
手続き型地図のノード属性を参照しながら自己位置を計測するのに要する処理時間は、目標物の数や検出する特徴に依存するが、本実験の場合、手続き型地図を作成したワークステーション上で約 20 秒かかった。このため、各位置確認地点においてロボットは数十秒間停止しなければならないが、この時間のかかる処理は離散的な位置確認地点だけで行われ、それ以外は 800ms 間隔でステアリングにフィードバックをかける沿目標物走行を用いるため、0.3km/h の速度で位置確認地点間を移動することができた。走行環境を変更し、同様の走行実験を 10 通り行なったが、それぞれ手続き型地図は 10 分程度で作成でき、かつ本実験と同等の処理速度が得られた。



(a) Left image

(b) Right image

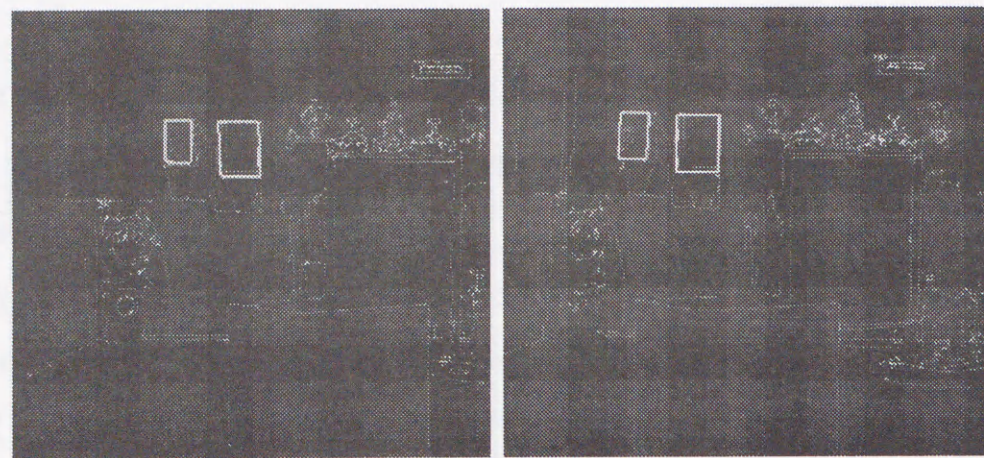
図 39: Stereo images at location P



(a) Left image

(b) Right image

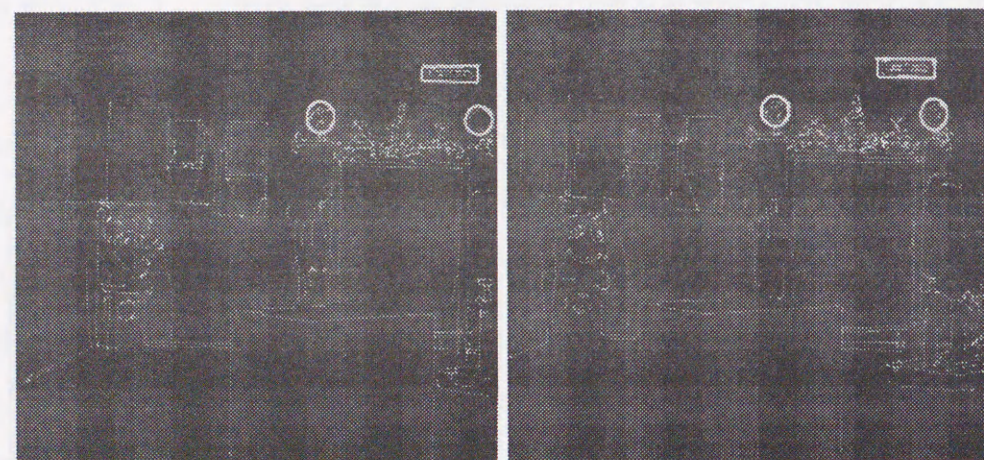
図 40: Stereo images at location R



(a) Left image

(b) Right image

図 41: Target T1 extraction result



(a) Left image

(b) Right image

図 42: Target T2 extraction result

5 行動指向型誘導アルゴリズム

病院、オフィス内等のように、人間等の移動障害物や椅子等といった可動物体が多数存在する動的環境においては、ランドマークが障害物により隠され、検出できない事態が頻発する。このため、経路に沿ってランドマークを検出し、自己位置を順次確認していくことが困難である。

動的環境においては、障害物の位置が特定できないため、走行経路を設定することが難しく、走行手順の作成も困難である。このため、移動車の動きを障害物回避、徘徊、目標物探索・追跡等といったいくつかの行動に分類し、これらの行動を適宜切り替えることにより、障害物を回避しつつ目標物を探索し、目的地へ接近する行動指向型誘導アルゴリズムを提案する。本アルゴリズムは静的環境においても適用可能であるが、行動の組み合わせで移動車を制御するため、与えられた経路に沿って正確に誘導することが難しく、効率の良い移動は実現できない。このため、静的環境で移動車を設定した経路に沿って常時走行させるには、手続き型地図による誘導アルゴリズムが有効である。

5.1 従来の手法

環境の変動が大きく、観測が安定に行えない動的環境においては、目標物を見失う等といった処理の失敗が頻発すると考えられる。このような場合においても、ロボットが制御不能に陥らないよう、障害物検出、目標物追跡等ロボットが必要とする基本的な行動をそれぞれ独立のプロセスとして構成する。そして、これらのプロセスを並列に動作させることにより、基本行動の一部が失敗しても、ロボットが制御不能に陥らない行動指向型の考え方が提案されている。この考え方では、複数のプロセスの非同期な出力からロボットが採るべき行動を決定するメカニズムがポイントとなる。Brooksの提案した包含(subsumption)構成[30]では、基本行動を、障害物回避のような反射的かつ頑強なプロセス(下位)から目標物追跡のような目的達成的かつ安定動作が困難なプロセス(上位)へと階層化し、各層の間に固定的な結合を設け、上位の出力が下位の出力を抑制すること

で、行動を一意に決定している。

この包含構成の課題は、下位の反射行動と上位の目的達成行動のギャップをどう埋めるかという点である。例えば、人間の往来が多い環境においては、障害物回避等のために目標物追跡等の目的達成行動（上位レベル）は失敗し易く、徘徊行動（障害物に衝突しないよう動き回る行動）等下位レベルの行動が頻繁に発現し、同じ障害物と障害物の間を往復する等、同一経路を繰り返し動き回り、目的地へ到達できなくなるといったデッドロックに陥る可能性が高くなる。また、デッドロックから復帰するための機能が無いため、一旦デッドロックに陥ると、永久に抜け出せなくなる可能性がある。

本論文では、基本行動の一部が失敗したことにより発生するこのようなデッドロック状態から効率的に回復することが可能な群化マルチエージェント構成を提案する。基本的な移動行動を独立のプロセス（以下、エージェントと呼ぶ）として作成し、行動決定を行うプロセスに直接データを出力できるようにした。そして、従来の反射行動エージェント群（障害物回避）と目的達成行動エージェント群（目標物追跡）に加え、デッドロック状態等を回避するためのヒューリスティクスを基本行動として出力するエージェントから成る群を新たに準備し、全てのエージェントをこれら3つの群に割り当てた。

このようなマルチエージェント制御方式としては、機能、制御、計画の3つの階層を設け、ロボットの行動目的に応じて制御層を機能層のエージェントで構築できるよう可変性を持たせたもの[73]、各エージェントの間に抑制結合を持たせ、失敗が発生した場合に速やかに回復できるようにしたもの[74]、基本行動のネットワークを準備し、現在の状況と行動目的の両方から活性・抑制制御を伝播させることにより行動決定を行うもの[75]等、目的達成行動と反射行動との間のギャップを埋めようとしたものが、従来から提案されている。しかし、複数に階層化し計画層を設けたり、エージェント間に直接結合を持たせた場合、エージェント相互の干渉が発生し、行動指向型本来の目的であった「一部の基本行動の失敗に対する安定性」は必ずしも保証されなくなる。また、デッドロックから有効に回復する手法も考えられていない。

5.2 設計の基本方針

動的環境においては、目標物の失認、デッドロック等様々な処理の失敗が随時発生する可能性がある。これに対処するため、まず対処すべき失敗を抽出した後、この失敗から回復するためのヒューリスティクスを新たに基本行動（以下、適応行動と呼ぶ）として準備した。ここでは、動的環境としてオフィス等の屋内環境を想定し、1台の移動ロボットが目的地まで移動する場合を説明する。

目標物を追跡するエージェントにおける追跡の失敗や、目標物を探索するエージェントにおける目標物の不検出等は、各エージェント自体で検知可能な失敗である。また、障害物に接触したため安全装置が働き、ロボットが停止した状態が継続する状況や、近接した複数の障害物に挟まれて抜け出せなくなるデッドロック状態もロボットの動きから検知可能な失敗である。このような検出可能な失敗に対し、陥った状況から回復するための基本行動を準備することで移動効率の低下を防止する。また、障害物検出は、複数種類のセンサを用いる独立のエージェントを準備することで、障害物を失認する危険性を下げる。

さらに、エージェントを階層化するのではなく、各エージェントを反射行動、目的達成行動、及び失敗を回復するための適応行動に分類し、独立の群にまとめた群化マルチエージェント構成[76]を新たに考案した。

オフィス等の動的環境で自律移動するシステムを設計する場合、移動中に発生する様々な失敗からできるだけ効率的に回復できる行動選択メカニズムが重要である。移動中に個々のエージェントが失敗しないならば、必要最小限の反射行動エージェントと目的達成行動エージェントを準備し、これらの出力を直接切り替えることにより行動を選択すれば良い。

しかし、安定性を高めるために反射行動エージェントを複数用意したり、失敗に対処するエージェントを付加したりすることでエージェント数が増加すると、各エージェントの出力から行動を決定する手法が非常に複雑になり、エージェントの一部が失敗しても制御不能に陥らないという行動指向型本来の利点が失われる危険性がある。このため、同種のエージェントをまとめた群を構成し、群単位で行動選択を行うことで、効率良く行動決

定を行えるようにした。この構成は、失敗への対処行動を新たに追加する等といった拡張性を考えた場合においても、群に新しいエージェントを追加するだけで済むため有利である。

障害物検知、回避等安全性を確保する行動を出力するエージェントをまとめた反射行動エージェント群 (Reflexive behavior agent group)、目標物検出・追跡等移動経路を効率良く進むための行動を出力するエージェントをまとめた目的達成行動エージェント群 (Purposive navigation behavior agent group)、及びこれらの失敗を回復するエージェントをまとめた適応行動エージェント群 (Adaptive behavior agent group) からなる群化マルチエージェント構成を構築した。群の数は、ロボットのタスクに応じて最適値が決定されるべきものであるが、ここでは、目的地へ向かう移動ロボットシステムを想定し、3に設定した。

図 43 にシステムの全体構成を示す。3つの群の外部に、群化されたエージェントからの行動出力を統合しロボットの移動行動を決定するための行動決定部 (Motion Executer) を設けた。行動決定部からの出力は制御部 (Robot Controller) に送信され、移動アクチュエータを制御する。

個々のエージェントは、現時点での外界観測結果及び前時点までの行動計算結果に基づき、現在の位置から次に進行すべき位置までの移動ベクトルを出力する。反射行動エージェントは、障害物が近くに存在した場合のみ移動ベクトルを出力するよう設計する。そして、各々の移動ベクトル (静止障害物検出エージェントの非常停止のための0ベクトルを含む) が出力された場合に活性化されると判定する。目的達成行動エージェントは、エージェント内部で目標物検出の失敗が検知された場合に、0ベクトルを移動ベクトルとして出力する。適応行動エージェントは、センサデータが供給されている限り、常時移動ベクトルを出力する。従って、適応行動エージェントに対する活性化判定は、出力した移動ベクトルの絶対値を算出し、予め定めた閾値と比較することにより行う。

3つの群の間の優先順位は、失敗が検知されていない通常状態においては、ロボットの安全確保が第一義に優先されるため、反射行動エージェント群が最優先となる。活性化された反射行動エージェント群が無い場合は、できるだけ効率良く目的地へ到達するために目

的達成行動エージェント群が優先される。目的達成行動エージェント群に属したエージェントの失敗が検知された場合、5.4章で述べる選択規則に基づき適応行動エージェント群に属するエージェントを優先し、デッドロックに陥らないよう誘導する。

各々の群に対する行動選択方式について述べる。反射行動エージェント群に対しては、センサの種類により予め優先順位を定め、複数のエージェントが同時に活性化した場合の選択を行う。目的達成行動エージェント群に対しては、移動経路に沿って予め設定しておいた目標物の系列の順番に基づき予め優先順位を定めておく。適応行動エージェント群に対しては、失敗の種類に基づき、目標物の失認等目的達成行動エージェントの失敗、あるいは、デッドロック等の状況から回復するための行動を出力するエージェントを選択する。

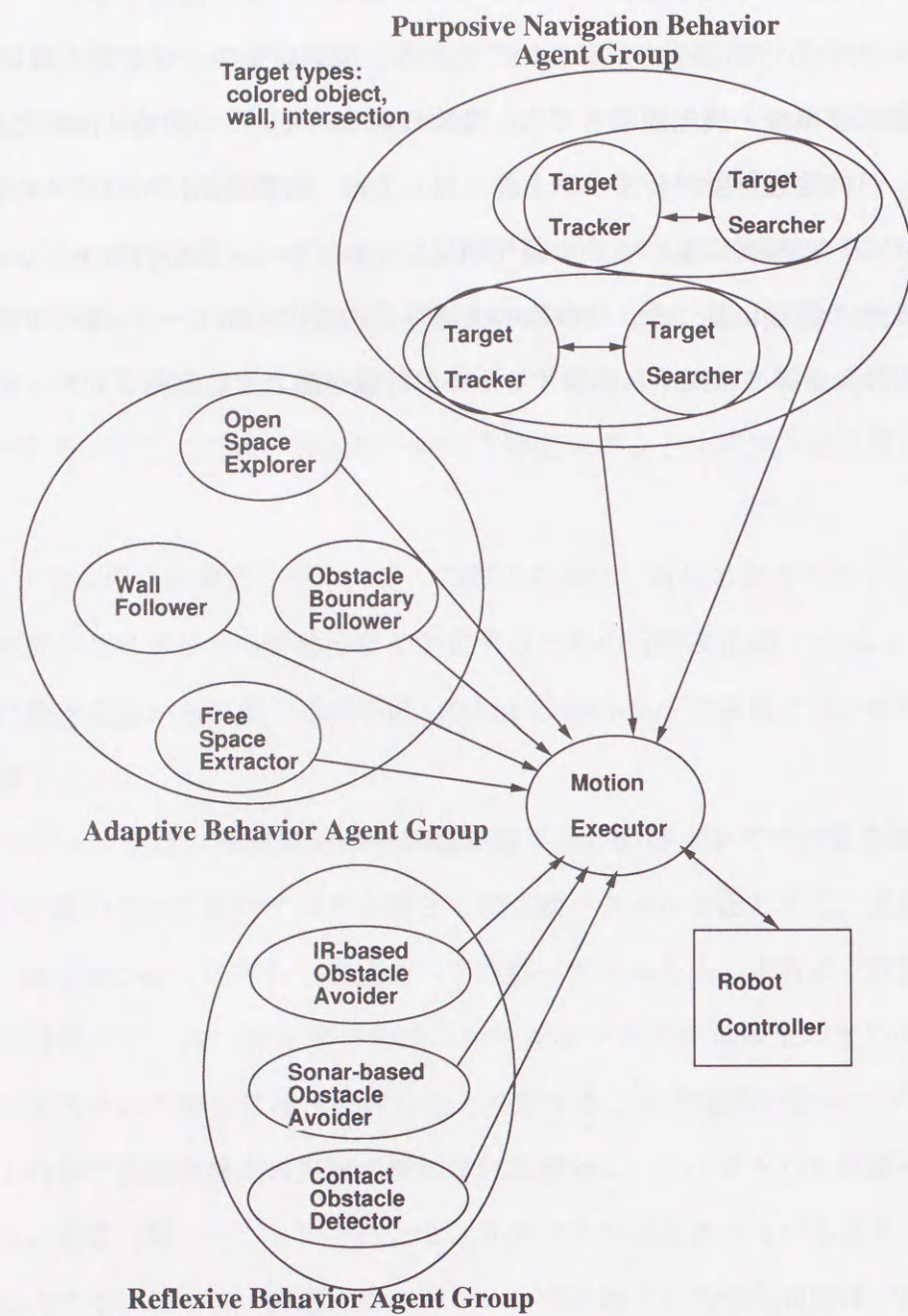


図 43: System configuration

5.3 各エージェントの詳細

5.2章で述べた基本方針に基づき、移動制御システムを計算機上に構築した。各エージェントは独立のプロセスとして作成され、プロセス間通信や共有メモリを介して行動決定部に行動出力を送信している。以下、各群に含まれるエージェントの詳細について説明する。

5.3.1 反射行動エージェント群

異種センサを用いたエージェントを複数用意することにより障害物を見落とす危険性を軽減している。触覚センサを用いた静止障害物検知エージェント、超音波センサから得られる距離情報を用いた障害物回避エージェント、人間を検知するための赤外線センサを用いた近接障害物検知エージェントを作成した。

静止障害物検知エージェントは接触センサが働いた時点で0ベクトル（非常停止）を出力し、停止する。近接障害物検知エージェントも、赤外線センサが働いた時点で0ベクトル（非常停止）を出力し、停止する。これに対し、障害物回避エージェントは、ロボット周囲の障害物までの距離情報を用い、最も周囲の障害物から離れた場所を求める。そして、そこへ移動するための移動ベクトルを出力する。

5.3.2 目的達成行動エージェント群

移動経路上の途中通過点、及び最終目的地に設定した目標物を、TVカメラの画像を用いて探索するエージェント (Target Searcher) と追跡するエージェント (Target Tracker) の組を基本単位として構成した。目標物探索エージェントと追跡エージェントを組にしたのは、目標物が検出されたら即座に追跡行動が発現し、追跡が失敗したら即座に目標物探索行動が発現するよう構成することにより、障害物回避等による目標物追跡の失敗から効率良く復帰することを図ったためである。本群は、色領域探索・追跡エージェント、壁面探索・追跡エージェント、及び交差点探索・追跡エージェントの3つから構成されている。

色領域探索・追跡エージェント

特定な色に着色されている目標物を探索・追跡するエージェントであり、マーク等が設置できる環境において有効である。カラー画像中から、目標物と同一な色相を持つ領域を抽出することにより、目標物を探索・追跡する [79][81]。図 44 に示すように、抽出された領域の重心と画像中心との間の距離を x 、カメラの焦点距離を f とすると、カメラの光軸方向に対し、

$$\theta = \tan^{-1} \frac{x}{f} \quad (38)$$

の方向に目標物が存在するため、 $\theta = 0$ となるよう移動ロボットを制御し、目標物に接近させる。移動ロボットが目標物に正しく接近している場合、画像中で目標物は、画像中心を通り、上下方法に動く軌跡を描くため、目標物の追跡結果は画像中心を通る垂直な軌跡 (図 45) を描く。このため、追跡結果が異なる軌跡を示した時、追跡処理が失敗したと判定し、目標物探索エージェントを活性化させる。

壁面探索・追跡エージェント

壁面や通路に沿って移動する際、有効なエージェントである [80]。一般的に、壁面や通路は平行な 3 次元直線の集合で記述される。画像中では、これら平行な 3 次元直線は 1 点 (消失点) に交わるため、以下の手順で画像中の消失点位置 (x_v, y_v) を求める。

- (1) エッジ画像からある程度長い線分 L_i を求める
- (2) 各 L_i を延長し、その方向と垂直方向に沿って輝度の分布が 1 次元ガウス分布となるような幅を持たせる (図 46)。ノイズや直線抽出に含まれる誤差の影響で、平行な 3 次元直線は、画像中で正確には 1 点に交わらない。このため、この処理により直線に幅を持たせ、交点が得られるようにしている。
- (3) 1 次元ガウス分布を持たせた各直線を入力画像と同一サイズの投票用画像に足し込み、輝度値がピークとなる位置を消失点位置 (x_v, y_v) とする。

図 48 に図 47 のエッジ画像、図 49 に 1 次元ガウス分布の輝度の幅を直線に持たせ足し込んだ投票結果、図 50 に消失点の抽出結果を示す。

壁面や通路が伸びる方向に対するカメラのパン角を θ 、床面に対するチルト角を α とすると、消失点位置は、

$$(x_v, y_v) = (f \frac{\tan \alpha}{\cos \theta}, f \tan \theta) \quad (39)$$

となる。よって、

$$\theta = \tan^{-1} \frac{y_v}{f} \quad (40)$$

となり、 $\theta = 0$ となるようロボットを制御することで、壁面や通路に沿ってロボットを誘導することができる。ここで、 f はカメラの焦点距離である。

交差点探索・追跡エージェント

室内環境を移動する場合、通路 (廊下) が交差する交差点の情報は位置を同定する上で重要である。このため、交差点を目標物として探索・追跡し、交差点へと誘導するエージェントを作成した [79]。

本エージェントは、「交差点は壁面の末端に存在する」という単純なヒューリスティクスを用い、画像中から交差点を検出している。以下に検出手順を示す。

- (1) 壁面探索・追跡エージェントと同様の手法により、消失点位置を求める。
- (2) 消失点を通る床面上の 2 本の直線を通路端エッジの組として求める。
- (3) 通路端の直線に下端が接している垂直線の中で、ロボットからほぼ等距離に位置するものを、左と右の通路端直線からそれぞれ 1 本ずつ求める。

図 50 に通路端エッジの抽出結果を示す。カメラの光軸方向が壁面と平行であるなら、ロボットから等距離にある垂直エッジと通路端エッジの交点は画像中で高さが同一となる。このため、消失点位置から求めたカメラのパン角を 0 にするようロボットの方向を制御した後、この条件を満足する 2 本の垂直エッジの組を求める。

図 51 に、図 50 から垂直エッジのみを選択した結果を示す。この垂直エッジの中から、本条件を満たす垂直エッジの組を求めた結果を図 52 に示す。図 52 では、垂直エッジの組の上に同じ長さの平行線を重ね書きし、強調表示している。交差点探索エージェントがこの垂直エッジの組を見つけたならば、交差点追跡エージェントが活性化され、SAD (Sum

of Absolute Difference) 法により垂直エッジの追跡を行う。そして、2本の垂直エッジの中央位置に向かうよう、ロボットを制御する。ロボットのステアリング角 θ_i は、

$$\tan \theta_i = \frac{C_{avr} - C_i}{f} \quad (41)$$

で得られる。ここで、 C_{avr} は画像中で検出された2本の垂直エッジの x 座標値の平均値、 C_i は画像中心の x 座標値、 f は焦点距離である。

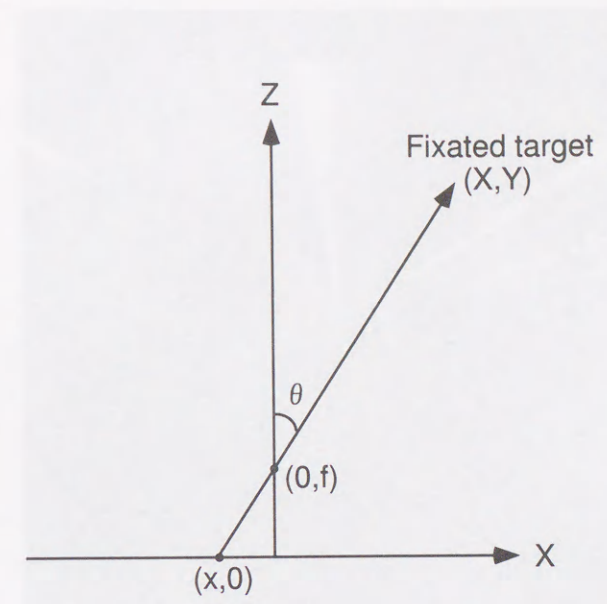


図 44: Movement toward fixated target

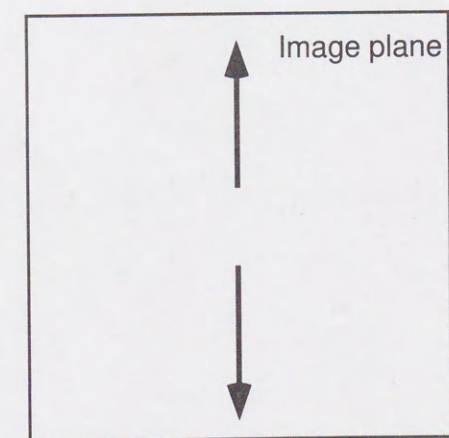


図 45: Result of tracking one fixated target

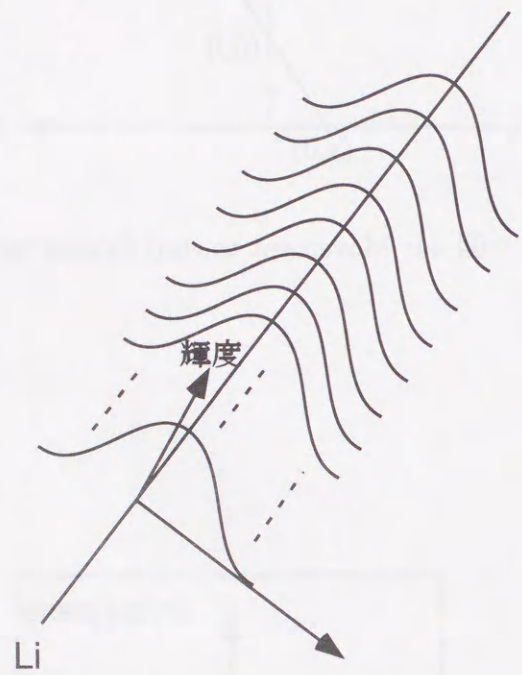


図 46: Probability distribution function of vanishing points along edge segments

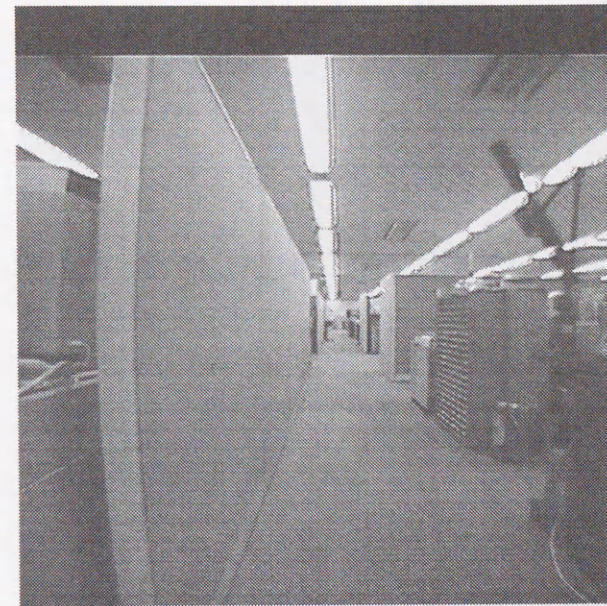


図 47: Input image

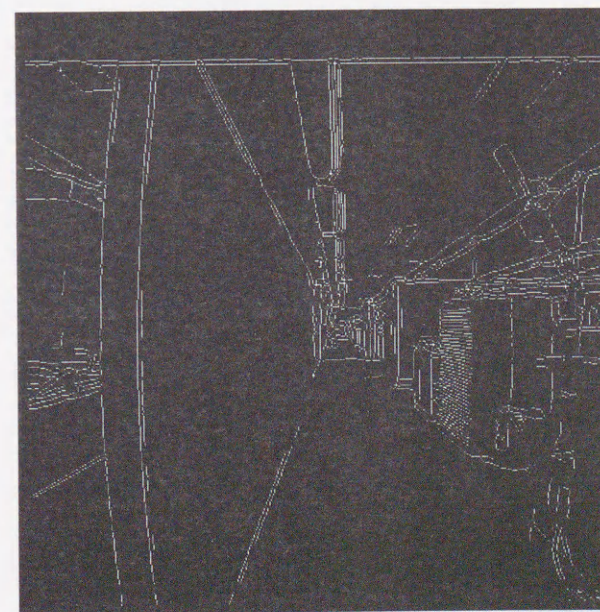


図 48: Extracted edges

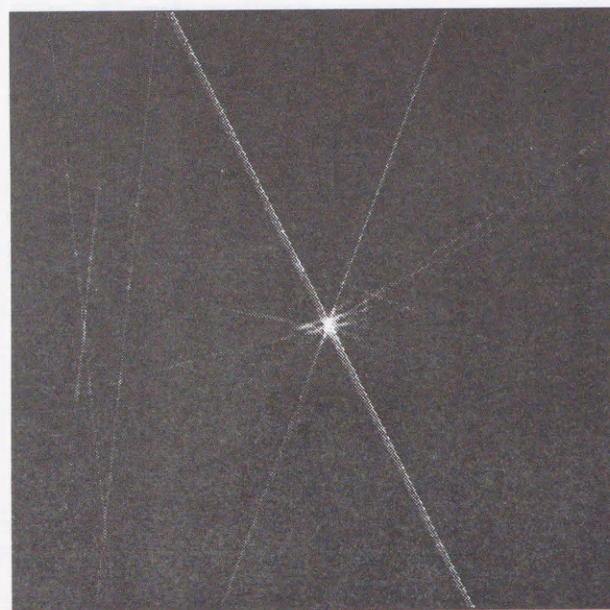


图 49: Voting result

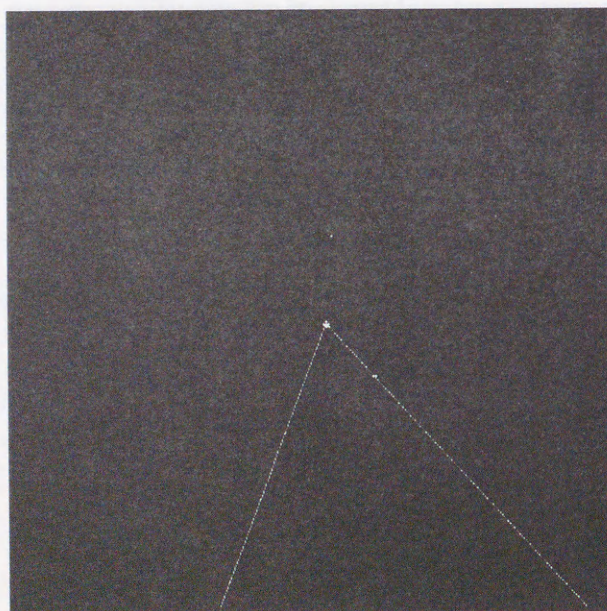


图 50: Extracted vanishing point

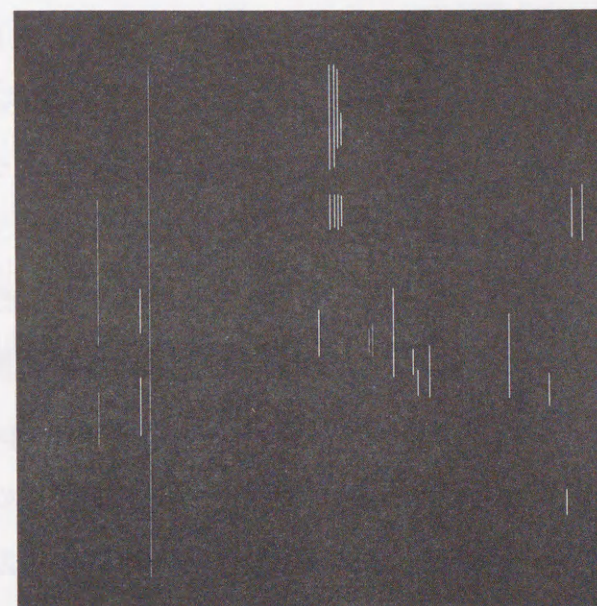


图 51: Candidate vertical edges

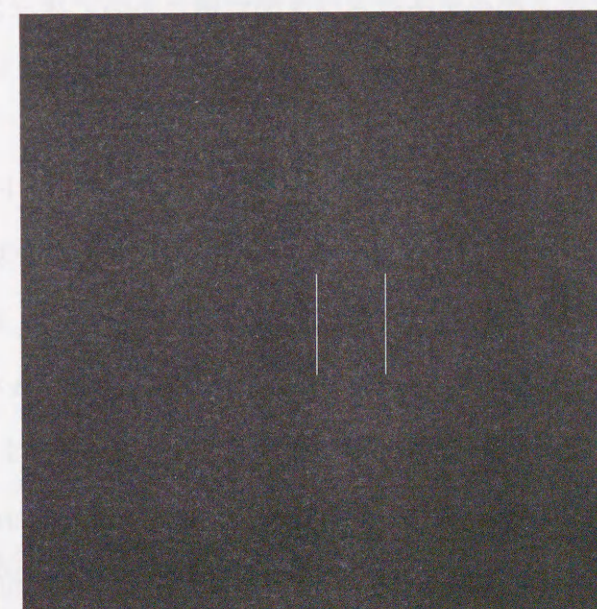


图 52: Extracted intersection

5.3.3 適応行動エージェント群

自己の周囲の障害物が存在しない空間（以後、自由空間と呼ぶ）を検出し、最も広い自由空間へ移動する行動を出力する自由空間探索エージェント (Free Space Explorer)、ドアの外のように開放された空間へ進行する行動を出力する開空間探索エージェント (Open Space Explorer)、障害物に沿って移動する行動を出力する沿障害物走行エージェント (Obstacle Boundary Follower) の3つから構成される。これらはすべて超音波センサをベルト状に複数並べた超音波アレイを用いている。反射行動エージェント群に属する障害物回避エージェント (Obstacle Avoider) と自由空間探索エージェント (Free Space Explorer) の違いは、前者が現時点でのセンサ出力のみを用いて回避行動を決定するのに対し、後者は前時点の行動出力結果も用いることにより、自由空間への円滑な行動出力を得る点にある。これらのエージェントを導入することにより、従来の移動システムで用いられた単純徘徊行動より目的に合った行動の選択が実現できる。

センサから得られる自己の周囲の距離情報と前時点の行動からエネルギー最小化原理に基づく計算により次に進むべき位置を求める行動決定方式を開発した。以下で、本方式を説明するとともに、上記3つのエージェントの詳細について述べる。

エネルギー最小化原理に基づく行動決定方式

超音波センサは超音波の伝播時間を計測することにより、ロボット周囲の距離情報を安価なシステムで獲得することができる。このため、複数の超音波センサ（超音波アレイ）を用い、エネルギー最小化原理に基づく行動決定手法を提案した [77]。本手法は、Snakes [78] として知られるエネルギー最小化原理に基づく輪郭抽出モデルを用い、複数のセンサデータからロボットが進むべき最良の位置と方向を決定する。Snakes はエッジの強度等から定義される External force と輪郭の滑らかさ等から定義される Internal force との釣り合いをとることにより輪郭の形状を求めるが、この考えをロボットの位置・方向を算出することに適用した。つまり、全ての External force 及び Internal force の釣り合いが維持される位置にロボットを誘導する。

External force としては、各超音波センサから得られる障害物までの距離が均一となる

位置へロボットを押し出す Range force f_i と、目標物へ引っ張る力を与える Target force を設定した。また、Internal forces としては、ロボットの移動軌跡ができるだけ滑らかになるような力を与える Smooth force f_s と、ロボットの走行メカニズムから決まる Dumping force f_d を設定した。

ロボットに働く External force と Internal force の例を図 53 に示す。ここで、ロボットは 2 次元平面上を移動すると仮定している。図 54 を用いて、エネルギー最小化原理に基づく位置決定手法を説明する。ロボットの位置は環境内に設定した 2 次元座標系 (X-Y) におけるベクトル

$$v_t = (x_t, y_t) \quad (42)$$

で記述される。ここで、 (x_t, y_t) は時刻 t でのロボットの位置を示す。Smooth force f_s は、

$$f_s(v) = \alpha (v_t^+ - v_t) \quad (43)$$

で求まる。ここで、 v_t^+ は滑らかさの拘束条件より算出される時刻 t でのロボットの位置であり、 v_t はロボットの現在位置である。また、 α はロボットの移動軌跡の滑らかさの度合いを制御するパラメータである。滑らかさの拘束条件を満たすロボット位置 v_t^+ は、単純な三角法を用い、以下の式から求めることができる。

$$v_t^+ = (x_t^+, y_t^+) \quad (44)$$

$$x_t^+ = 2x_{t-1} - x_{t-2} - 2(y_{t-1} - y_{t-2}) \tan\left(\frac{\Delta\theta}{2}\right) \quad (45)$$

$$y_t^+ = 2y_{t-1} - y_{t-2} - 2(x_{t-1} - x_{t-2}) \tan\left(\frac{\Delta\theta}{2}\right) \quad (46)$$

ここで、 $\Delta\theta$ は、図 54 に示す 2 直線 $v_{t-2}v_{t-1}$ 、 $v_{t-1}v_t^+$ の成す角度である。また、ロボットは等速で移動するため、 $\overline{v_{t-2}v_{t-1}} = \overline{v_{t-1}v_t^+}$ である。

ロボットの移動機構 (dynamics) から設定される Dumping force $f_d(v)$ は、

$$f_d(v) = d(v_t - v_{t-1}) \quad (47)$$

で与えられる。ここで、 v_t は時刻 t (現在)のロボット位置であり、 v_{t-1} は時刻 $t-1$ のロボット位置である。また、 d はロボットのdynamicsの影響を制御するパラメータである。

複数のセンサから得られる Range force f_t^j は、

$$f_t^j = k(s_j - v_t) \quad (48)$$

より算出される。ここで、 $s_j = (x_s^j, y_s^j)$ は、デカルト座標系における j 番目の距離センサの出力値であり、 k は Range force の影響を制御するためのパラメータである。

f_s 、 f_d 、及び複数センサからの f_t^j の間で平衡状態を満足する位置が、ロボットの次に進むべき位置 v_{t+1} であり、式 43,47,48 より算出することができる。

$$v_{t+1} = v_t + \sum_{i=0}^n f_t^i + f_d + f_s \quad (49)$$

ここで、 n は超音波センサの数である。次に進むべき位置 v_{t+1} は、Smooth force、Dumping force、及び Range force の持つ各パラメータ α 、 d 、 k により制御される。

自由空間探索エージェント

自由空間探索エージェントは、ロボットにとって最も安全な場所として、最も開けた空間へロボットを誘導する。エネルギー最小化原理に基づく行動決定手法を用い、移動すべき場所を算出する。Range force を制御するパラメータ k を大きな値に設定することにより、Range force の影響力を Smooth force 及び Dumping force より強くする。この操作により、ロボットは各センサから得られる Range force が釣り合う位置、すなわち周囲の障害物から最も離れた位置へ移動する。

沿障害物走行エージェント

近接する障害物までの距離計測結果から障害物表面に対する接線方向の力 (Tangential range force) を求め、障害物表面に対し接線方向に移動するようロボットを誘導する。式 48 中の $f_t^j(s, v)$ に代えて、Tangential range force

$$f_t^j = \begin{cases} k(s_j - s_{j-1}) & \text{if } \rho_j \leq \rho_{th} \text{ and } \rho_{j-1} \leq \rho_{th} \\ 0 & \text{otherwise} \end{cases} \quad (50)$$

を用いる。 s_j は j 番目の距離センサの距離計測結果 ρ_j のデカルト座標系における値である。また、 s_{j-1} は $j-1$ 番目の距離センサの距離計測結果 ρ_{j-1} のデカルト座標系における値である。 ρ_{th} は対象とする障害物の近さを設定する閾値である。本エージェントは、近くの障害物表面に対し、常に接線方向へロボットを誘導するため、ロボットが障害物と衝突することはない。また、壁面を障害物と考えると本エージェントは壁面に沿ってロボットを移動させることもできる。

開空間探索エージェント

室内環境では、部屋のドアのような出口を見つける行動がデッドロックを回避する上で重要である。この開空間探索エージェントを自由空間探索エージェントを基に作成した。図 55 に示すように、ロボットの背面に仮想的な壁面を設定し、ロボットの背後から得られる超音波センサの読みをこの壁面からの距離に置き換える。そして、これらの距離データに対し自由空間探索エージェントと同じアルゴリズムを適用し、ロボットを誘導する。ドア等の開空間がロボット前方に存在するなら、その開空間方向へロボットは引っ張られるため、本エージェントにより、ロボットはドアなどの狭まった空間を通過することができる。

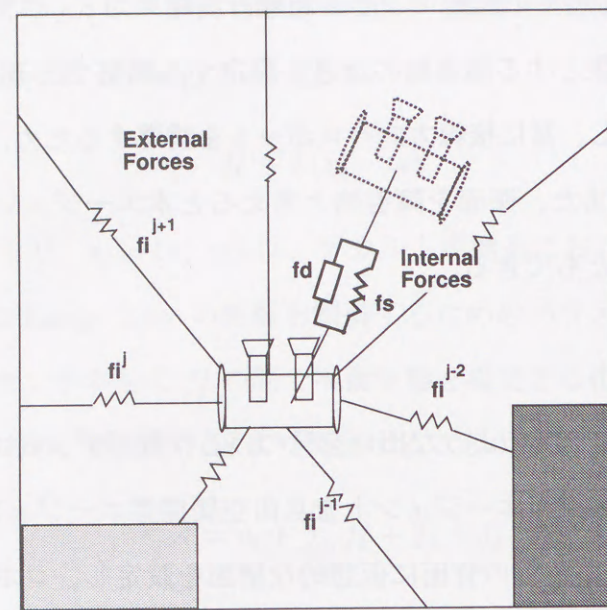


图 53: Energy-based motion decision algorithm

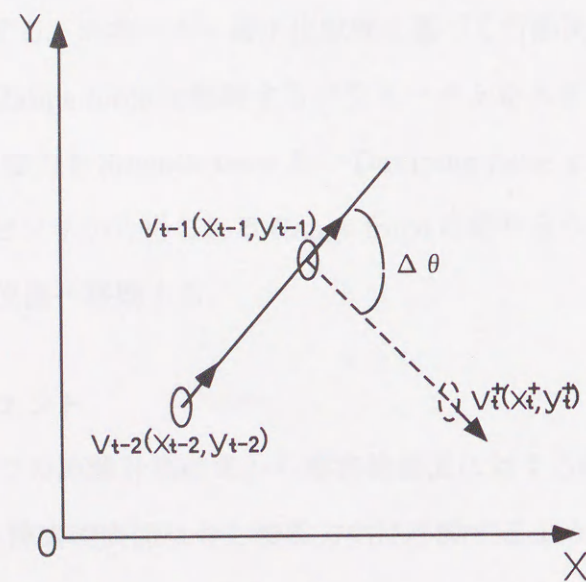


图 54: Smoothness constraint for robot's locations

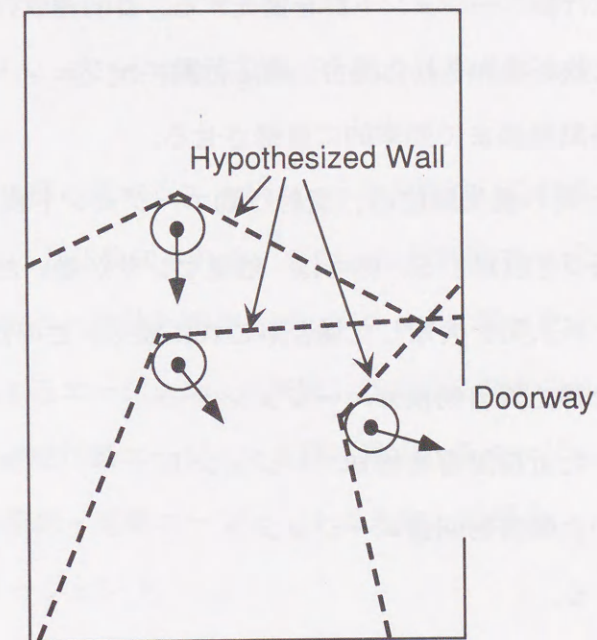


图 55: Open-space-explorer using free-space-explore

5.4 行動の選択

各エージェントがそれぞれ独立に出力してくる行動の中から、現在実行すべき1つの行動を選択する必要がある、図43中の行動決定部 (Motion Executer) がこの行動選択を行う。各群間及び各群中のエージェント間に優先順位を設定し、この優先順位に従ってロボットが現在実行すべき行動を選択する。

3つの群の間の優先関係は、ロボットの安全確保が第一義に優先されるため、反射行動エージェント群が最優先となる。活性化された反射行動エージェント群が無い場合は、危険が無いとして目的達成行動エージェント群を優先する。目的達成行動エージェント群に属するエージェントの失敗が検知された場合、適応行動エージェント群に属するエージェントの行動を選択し、移動経路まで効率的に復帰させる。

各群中のエージェント間の優先順位は、反射行動エージェント群の場合、使用しているセンサの優先順位に基づき設定する。例えば、触覚センサが働いた場合、最も危険であり、次いで赤外線センサが人間を検知した場合がこれに続く。このため、

- ①触覚センサを用いた静止障害物検知エージェント
- ②赤外線センサを用いた近接障害物検知エージェント
- ③超音波センサを用いた障害物回避エージェント

の順で優先順位を設定する。

目的達成行動エージェント群の場合、移動目的に応じて優先順位が設定される。例えば、オフィス等の代表的な屋内環境では、壁面追跡や交差点追跡を組み合わせ、目的地までロボットを誘導する。図56に示す環境では、壁面追跡 → 交差点追跡 → 壁面追跡の順で行動を選択することにより、地点Aから地点Gへの移動が実現できる。このため、この行動選択のシーケンスに従って各行動の優先順位を設定する。つまり、壁面追跡行動を選択すべき場所では、壁面探索・追跡エージェントを優先させる。また、交差点追跡行動を選択すべき場所では、交差点探索・追跡エージェントを優先させる。

適応行動エージェント群に含まれる各エージェントは、目的達成行動エージェント群に属するエージェントが失敗した場合及びデッドロックに陥った場合に選択される。目的

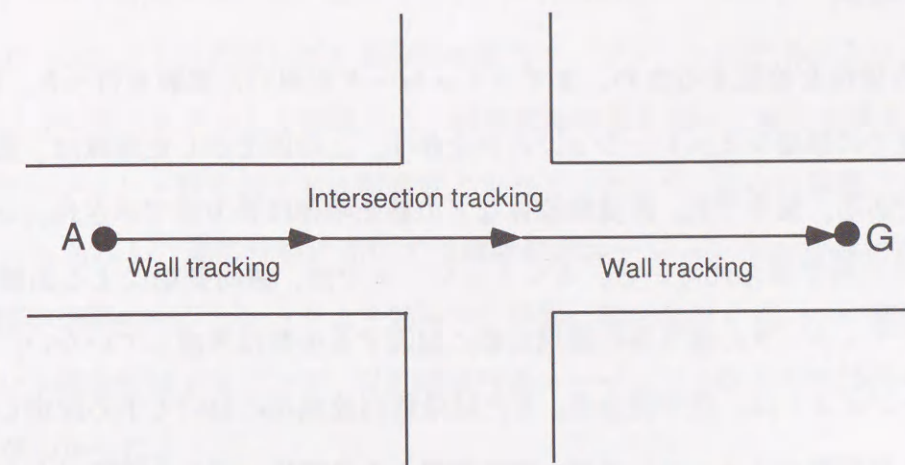


図 56: Example of indoor environment.

達成行動エージェント群に属するエージェントの失敗は、これまで成功していた壁面追跡や交差点追跡の失敗により検知される。また、反射行動エージェント群に含まれる同一のエージェントが選択され続けた場合、デッドロックに陥ったと判断する。目的達成行動エージェント群に属するエージェントが失敗した場合に選択される適応行動エージェントの優先順位は、目的達成行動エージェント群に属する各エージェント毎に予め設定しておく。例えば、交差点探索・追跡エージェントが失敗した場合、

- ①沿障害物走行エージェント
- ②開空間探索エージェント
- ③自由空間探索エージェント

の順で適応行動エージェント群に含まれるエージェントが選択される。この優先順位により、ロボットはまず壁面に沿って移動するが、交差点に接近し壁が途切れると沿障害物走行エージェントの出力が弱まるため、開空間探索エージェントが選択され、交差点中央にロボットが誘導される。開空間探索エージェントの出力が弱い場合、自由空間探索エージェントが選択され、徘徊行動を行う。デッドロックに陥っても、壁面に接近すると沿障害物走行エージェントの出力が強まるため、壁面に沿って移動する行動が選択され、デッドロック状態から回復できる。

5.5 実験結果

本構成の有効性を検証するため、まずシミュレータを用いた実験を行った。図 57、58 に、室内環境での移動シミュレーションの例を示す。この例で示した環境は、実験室を模擬したものである。図中で机、計算機筐体などの静止物体は長方形で示され、ロボットの移動軌跡は円の列で示されている。本シミュレータでは、照明変動による画像処理の失敗、移動アクチュエータの滑り等の環境変動に起因する失敗は考慮していない。目標物探索・追跡エージェントは、途中通過点、または最終到達地点において予め設定しておいた目標物がカメラ視野内に入っていれば、現在位置から目標物へ向かう移動ベクトルを出力する。一方、障害物により目標物が遮られ目標物探索が失敗すると 0 ベクトルが出力される。赤外線センサ及び超音波アレイを用いたエージェントは、センサの検知範囲内の環境を調べ行動を出力する。

図 57 は、目的達成行動エージェント群と反射行動エージェント群だけで動作させた場合のロボットの移動結果である。目標物として設定した部屋の出口の両端 A、B は、壁に遮られて移動開始位置からは直接見えないため、目標物探索エージェントは失敗する。適応行動エージェント群がないため、ロボットは徘徊行動を採り、前方の自由空間に向かって直進する。そして、反対側の壁に近づくと障害物回避エージェントが働き、ロボットの進行方向が切り替わる。以後、ロボットはこの行動を繰り返しデッドロックに陥る。ロボットの移動開始位置・方向を変え、部屋の出口の両端 A、B が視野内に入るよう設定しても、壁に接近した際、障害物回避エージェントが働くため、反射行動と目的達成行動の優先関係から出口に接近する行動は選択されない。この様に、目的達成行動と反射行動のみから構成される従来手法では、いつまでも目的地へ到達できない状況が発生する。

これに対し、自由空間探索エージェント、開空間探索エージェント、沿障害物走行エージェントからなる適応行動エージェント群を付加すると、図 58 に示すように、ロボットが出口から外に出ることが可能となる。この場合、出口目標の探索が失敗すると、沿障害物走行エージェント、開空間探索エージェントが順に選択され、ロボットを出口付近まで誘導する。出口の両端点 A、B が視野内に入ると目標物探索が成功し、目標物追跡エージェ

ントがロボットを誘導する。壁に接近し過ぎ、障害物回避エージェントの行動出力で、目標物追跡エージェントが失敗しても沿障害物走行エージェントの出力により、壁に沿って進行し、出口の外へロボットを誘導する。移動開始位置を様々に変え実験を行った結果、適応行動エージェント群を加えた 3 群構成とすることにより、出口へ到達できることが確認できた。もちろん、適応行動の途中で目標物探索エージェントが目標物抽出に成功すれば、目標物追跡エージェントがより効率的な移動行動を出力する。この様に、適応行動エージェント群を付加することで、目的達成行動エージェント群の失敗に効率良く対処できることがわかった。

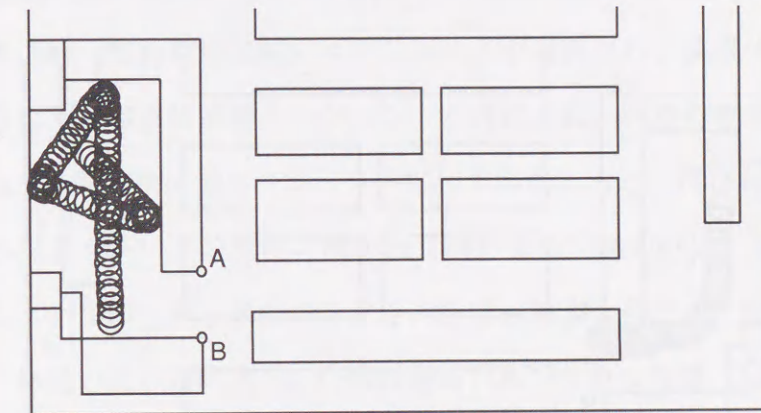


図 57: Motion sequence without Adaptive Behavior agents

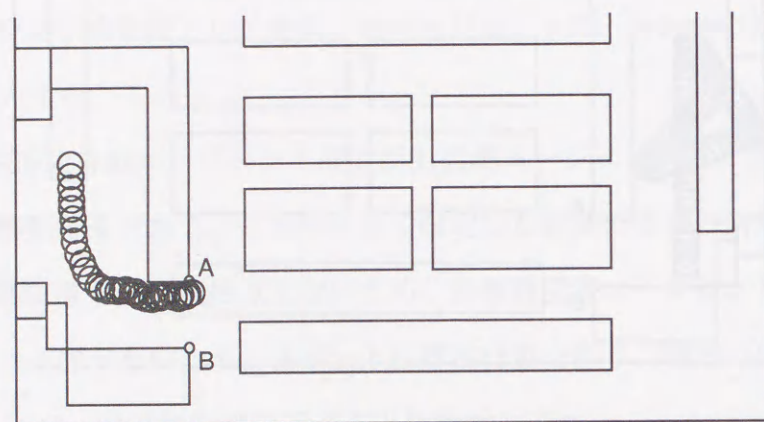


図 58: Motion sequence with Adaptive Behavior agents

実際のロボットを用いた実験結果を次に示す。図 59 にロボットの全景を示す。外界センサとして、2 台の TV カメラ、反射行動エージェント用にロボット上部に 4 個の超音波センサ、及び適応行動エージェント用にロボット中央部に 18 個の超音波センサアレイを備えている。また、18 個の赤外線センサアレイ及び導電ゴムによる接触バンパも備えている。移動は 2 つの駆動車輪と 2 つの自由車輪により行う。電源は車下部に装着された 12 V バッテリ 2 個から供給され、画像信号のみ有線で送信される。センサ及び車輪の制御はロボット搭載の CPU (MC68030) で行われ、外部の計算機ネットワーク上に構築された移動制御システムと無線で交信しながらロボットを移動させる。

図 60 に、目標物 (消火器) に向かって進行中、人間が出現した場合の行動選択例を示す。最初は、目標物追跡 (赤色領域追跡) エージェントが選択され、直進するが (図 60(a))、障害物が検知されると障害物回避エージェントが選択され、回避行動を採る (図 60(b))。障害物が無くなり、障害物回避エージェントの出力が弱まると、再び目標物探索エージェントが優先され探索を行い、目標物に向かって進行する (図 60(c))。

シミュレーション実験と同一構成のオフィス内で移動実験を行った結果を図 61 に示す。目標物として、部屋の出口の両端 (A,B) 及び通路の曲がり角の両端 (C,D) を設定し、交差点探索・追跡エージェントを優先するシーケンスを与えた。

最初、出口目標物 (A,B) が視野外にあるため、交差点探索・追跡エージェントが失敗する。周囲に接近した障害物が無く、沿障害物走行エージェントの行動出力が弱いため、開空間探索エージェントの行動が適応行動として選択され、ロボットを出口付近まで誘導する。出口の両端 (A,B) が視野に入り、交差点探索エージェントが成功すると、交差点追跡エージェントがロボットを出口まで誘導する。これは、シミュレーションと同じ行動選択である。通路に出ると、まず次の目標物 (C,D) を検出し、これに向かって直進し始める。通路壁に近づき過ぎ、障害物回避エージェントの出力が強くなると、回避行動が採られ、目標物を見失う。この時、適応行動エージェント群の優先順位に従い、沿障害物走行エージェントの出力が選択され、ロボットを通路壁に沿って (C,D) の付近まで誘導する。曲がり角の両端 (C,D) が視野に入り、交差点探索エージェントが成功すると、交差点追跡エージェントがロボットを曲がり角まで誘導する。図 62 に、曲がり角 (C,D) に向かってロ

ロボットが移動している様子を示す。

以上述べたように、シミュレーション及び移動ロボットを用いた実験により、行動指向型誘導アルゴリズムがデッドロックや障害物出現後の目標物失認等、移動中に発生する様々な状況に有効に対処できることを確認した。

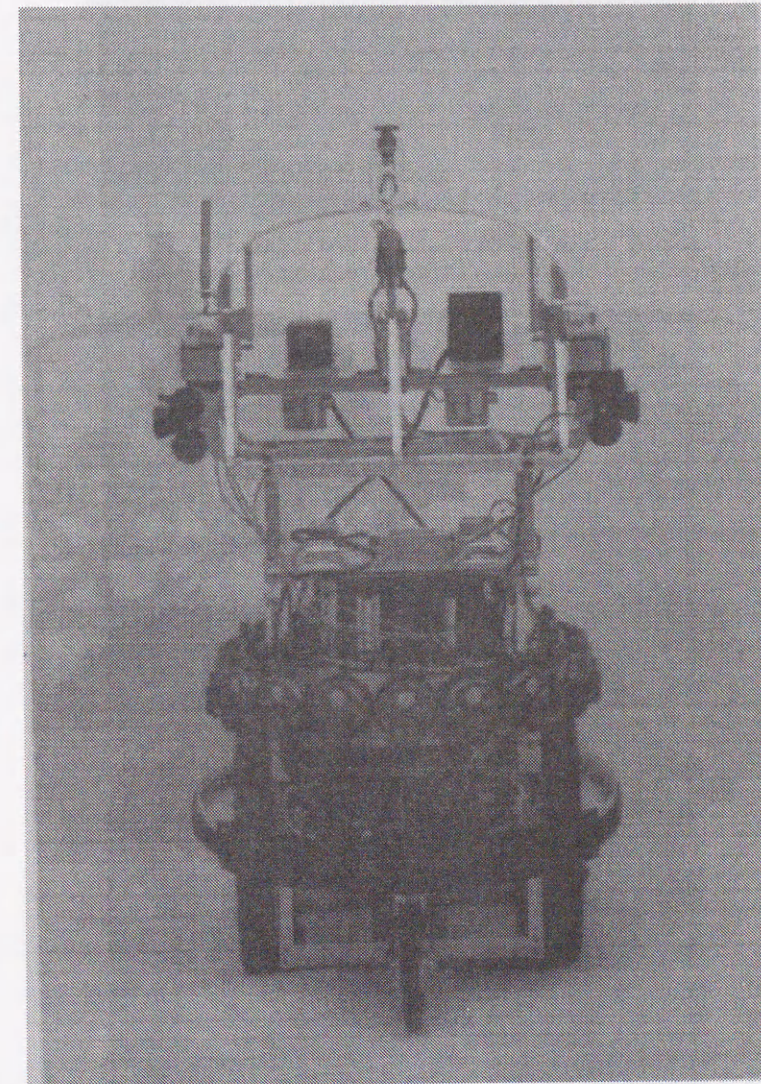
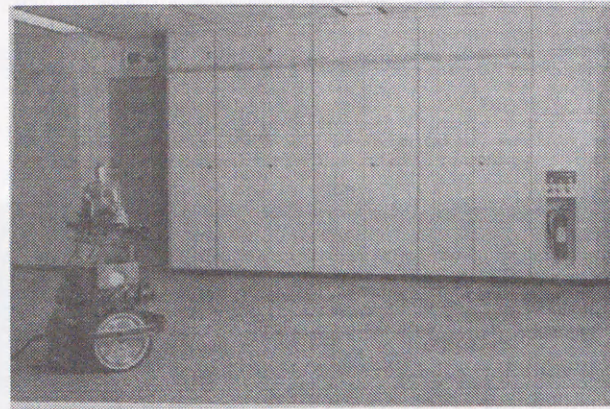
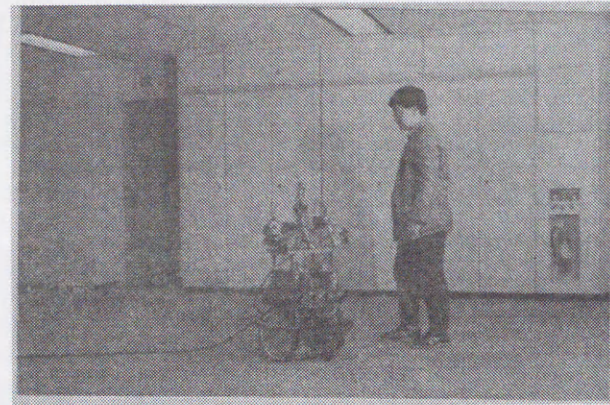


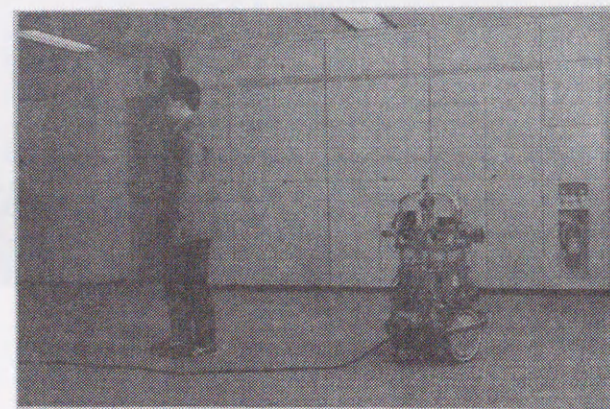
図 59: Birdie overview



(a) Target tracking

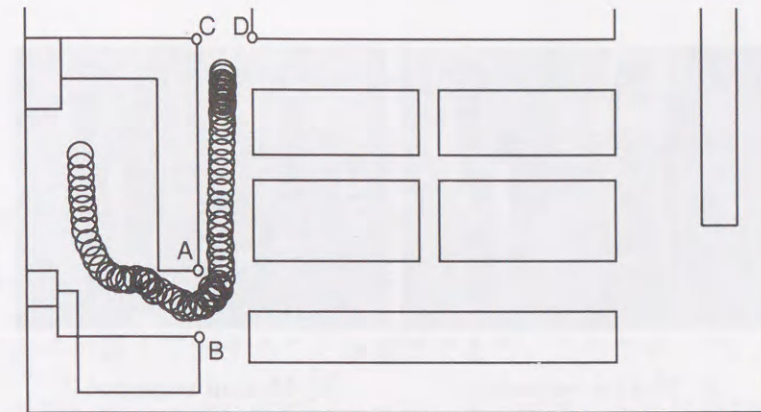


(b) Obstacle avoidnce

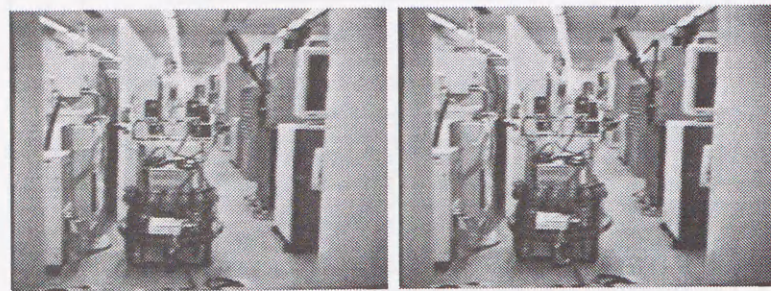


(c) Target tracking

☒ 60: Robot motion sequences of obstacle avoidnce

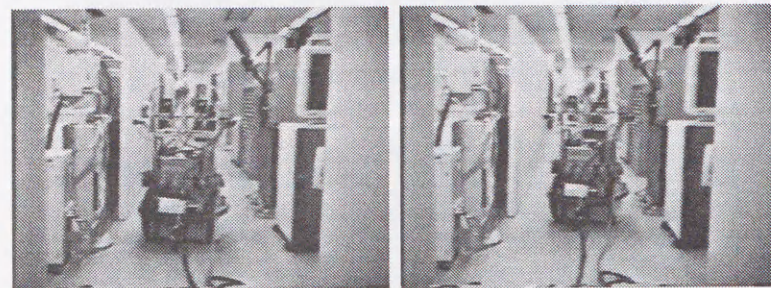


☒ 61: Total motion sequence



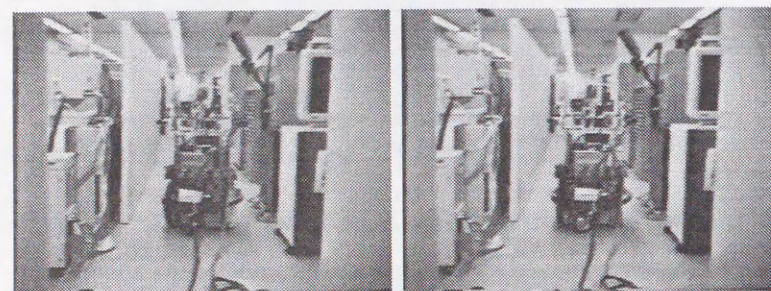
(a) Motion sequence 1

(b) Motion sequence 2



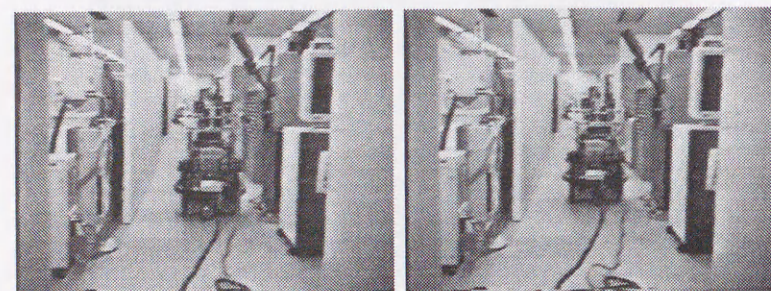
(c) Motion sequence 3

(d) Motion sequence 4



(e) Motion sequence 5

(f) Motion sequence 6



(g) Motion sequence 7

(h) Motion sequence 8

図 62: Robot motion sequences toward the intersection (C,D)

6 結論

本論文では、実環境において自律移動車や移動ロボットが安全かつ効率的に目的地へ移動するための空間把握と誘導のアルゴリズムを与えた。

道路等屋外環境においても適用できる空間把握アルゴリズムを実現するためには、広い範囲で障害物が検出できるとともに、歩行者や走行中の車等を移動障害物として切り出し、その進行方向や位置を予測することが重要である。このため、TV カメラをセンサとして用い、広い空間において密な距離マップが一度に獲得できる移動可能空間検出アルゴリズム、及び移動物体と静止物体を切り分け、移動物体の位置を算出する移動障害物位置検出アルゴリズムを提案した。

まず、移動可能空間検出アルゴリズムとして平面投影ステレオ法を提案し、実際の道路シーンによる実験結果を示した。平面への逆投影モデルを用いる手法は、設定平面の方程式が持つ誤差に弱く、道路面上のテクスチャ周辺に障害物として誤抽出される領域が多数現れるという欠点を持っていたが、平面投影ステレオ法は、平面投影差分の持つ特性に基づき、放射状走査で障害物の形状を調べることで、これらの誤抽出領域を除去することが可能である。このため、走行時に車体が多少揺れても安定に移動可能領域が抽出できる。実験結果を定量評価した結果、真の道路領域の89% (平均) が正しく抽出されていることがわかった。また、道路として抽出された領域中の3% (平均) が道路以外の誤抽出領域であった。安全性を考慮し、道路の幅が移動車の2倍以上あると仮定すると、道路端付近の10%程度の領域が未抽出であっても、移動車が走行できる十分な領域が残る。このため、本アルゴリズムにより移動車を安全に制御できることがこの定量評価結果から確認できた。

次いで、移動する車から画像を用いて歩行者、自転車、車等の移動障害物を検出し、車に対する位置を安定に算出する移動障害物位置検出アルゴリズムを提案した。本アルゴリズムは、まず、平面投影ステレオ法を用い、前方の移動可能な空間マップを作成する。次いで、オプティカルフローから動きの消失点位置 (FOE) を推定する際に得られる残差から画像中の移動障害物領域を抽出する。そして、この移動障害物領域を移動可能空間マッ

プ上に投影することで移動障害物と移動車との間の位置関係を求める。本アルゴリズムでは、フレーム間の追跡やステレオ画像間の対応探索が不要なため、画像中で見え方やサイズが大きく変化する対象物や、形状が常に変形する非剛体に対しても、安定にその位置を求めることができる。また、高さの情報を用いて障害物領域を求めるため、道路面上に現われる移動体の影を誤って抽出することがない。このため、歩行者、自転車、車といった一般道路上に通常存在する移動障害物の位置検出に有効である。

誘導アルゴリズムに関しては、ランドマークが安定に検出できる静的環境において効率の良い手続き型地図による誘導アルゴリズムと、人間等の移動障害物や椅子等といった可動物体が多数存在する動的環境において適用可能な行動指向型誘導アルゴリズムを提案した。

手続き型地図による誘導アルゴリズムは、走行経路上で自己位置を確認する地点を予め設定しておき、検出すべきランドマークの種類やその抽出手順等、位置確認のための手続きを記述した地図を作成しておく。移動ロボットはこの手続き型地図を参照しながら自己位置を確認し、目的地へ近づいて行く。この様に、要所要所の情報のみを使い易い形で記述した手続き型地図を用い、自己位置を同定するため、走行環境全体の画像や形状データ等を保存する従来手法と比べ、データ量を大幅に減らすことができ、屋外等のような広い走行環境にも適用することができる。手続き型地図は走行環境において実際に入力した画像を対話的に処理し作成するため、抽出が容易な目標物が選択でき、安定に自己位置が計測できる。また、位置確認地点間は通路に沿う等高速に処理できる手法を用いるため、円滑な移動が実現できる。原子力発電所内部を模擬した環境において誘導実験を行い、アルゴリズムの有効性を示した。

病院やオフィス内等のような環境では、人間等の移動障害物及び椅子等といった可動物体が多数存在し、位置確認のために設定したランドマークが隠されてしまう可能性が高い。このため、ランドマーク検出の失敗が頻発するような動的環境においても、目的地へ向かって走行できる3群から成るマルチエージェント構成の行動指向型誘導アルゴリズムを提案し、シミュレーション及び移動ロボットによるオフィス内移動実験で有効性を確認した。本構成は、障害物回避等の反射行動及び目標物探索・追跡等の目的達成行動に対

し、新たにデッドロック状態から抜け出すための適応行動（周囲の全ての障害物から最も離れた位置へ移動する自由空間探索行動、ドアの外といった開けた空間へ移動する開空間探索行動、及び障害物に沿って移動する沿障害物走行行動）を付け加えることで、目的達成行動の失敗及びデッドロック状況から有効に脱出することが可能である。

本論文で提案したアルゴリズムは、自律移動車及び自律移動ロボットが目的地まで安全かつ円滑に移動するためのシステムを構築する上で有用である。実用化のためには、処理速度や信頼性をより一層向上させることが必要であるが、計算機の演算性能は年々向上しており、また、TVカメラのフレームレート、ダイナミックレンジ及び解像度等の進歩も著しいため、本論文で提案したアルゴリズムが将来的には、市販車の衝突回避用安全装置や自動運転等の実現に役立つものと考えられる。

謝辞

本論文の執筆にあたり、ご指導、ご鞭撻を賜りました、東北大学大学院情報科学研究科の西関隆夫教授に心から感謝いたします。また、貴重なご意見を賜りました、東北大学大学院情報科学研究科の根元義章教授、東北大学大学院工学研究科の阿部健一教授に深く感謝いたします。

本研究は株式会社東芝研究開発センターにおいて実施したものであります。研究を進める上でご指導、ご助言いただきました木戸出正継博士、麻田治男博士、恒川尚氏、星野弘氏、岡崎彰夫博士、浮田輝彦博士、前田賢一氏に深く感謝申し上げます。特に、前田賢一氏においては、本研究を論文化するにあたり、多大なご支援、ご配慮をいただきました。厚く御礼申し上げます。

本研究を共同で行った渡辺睦博士、久野義徳博士、In So Kweon 博士、岡本恭一氏、石川実氏、武田信之氏等多くの人々に厚く御礼申し上げます。特に、4章、5章は、渡辺睦博士の貢献に負うところが多く、このような研究成果が得られたことに深く感謝いたします。また、4章、5章の研究を進めるにあたり、多くの技術的アドバイスをいただいた久野義徳博士、5章の研究において多大な貢献をいただいた In So Kweon 博士、4章で述べたアルゴリズムを実証するにあたり、基礎となるプログラムを作成していただいた岡本恭一氏、3.2.2章で提案した移動体抽出アルゴリズムの開発に貢献していただいた武田信之氏に感謝いたします。

最後に、長年の間、企業での研究生生活を支えてくれた妻の潤子に最大限の感謝を捧げます。

参考文献

- [1] N.J.Nilsson : "Shakey the robot", Technical Report No.23, SRI AI Center, 1984.
- [2] H.P.Moravec : "The Stanford Cart and the CMU Rover", in Proceedings of the IEEE, Vol.71, No.7, pp. 872-884, 1983.
- [3] S.Tsujii and J.Y.Zheng : "Visual Path Planning by a Mobile Robot", in Proceedings of 10th International Joint Conference on Artificial Intelligence(IJCAI'87), pp. 1127-1130, 1987.
- [4] 浅田稔: "センサ統合と環境モデルの構築", 信学技報 PRU89-101, 1989.
- [5] 石黒浩, パトリック ステルマジック, 辻三郎: "注視制御による局所地図の獲得", 信学論 (D-II), J74-D-II, 7, pp. 926-932, 1991.
- [6] K.Onoguchi, M.Watanabe, Y.Okamoto and H.Asada: "Visual Navigation System for a Mobile Robot", in Proceedings of International Workshop on Intelligent Robots and Systems(IROS'89), pp.590-597, 1989.
- [7] 森英雄, 小谷信司, 石黒浩, 安富敏, 茅野之雄: "構内道路を走行する自律移動ロボット (移動ロボットの一戦略 - 晴信4号機)", 日本ロボット学会誌, Vol.5, No.5, pp29-41, 1987.
- [8] 松本勉, 油田信一: "経路地図に従った移動ロボットの自律走行システム", 日本ロボット学会誌, Vol.5, No.5, pp.19-27, 1987.
- [9] S.Tsugawa, T.Yatabe, T.Hirose and S.Matsumoto: "An Automobile with Artificial Intelligence", in Proceedings of 6th International Joint Conference on Artificial Intelligence(IJCAI'79), pp.893-895, 1979.

- [10] J.Huber and V.Graefe: "Feature-based Motion Stereo Vision for Mobile Robots", in Proceedings of International Conference on Advanced Mechatronics(ICAM'93), pp.704-709, 1993.
- [11] M.Ohzora, T.Ozaki, S.Sasaki, M.Yoshida and Y.Hiratsuka: "Video-Rate Image Processing System for an Autonomous Personal Vehicle System", in Proceedings of IAPR Workshop on Machine Vision and Applications(MVA'90), pp.389-392, 1990.
- [12] K.Yamazawa, Y.Yagi and M.Yachida: "Obstacle Detection with Omnidirectional Image Sensor HyperOmni Vision", in Proceedings of 1995 International Conference on Robotics and Automation(ICRA'95), pp.1062-1067, 1995.
- [13] K.Storjohann, Th.Zielke, H.A.Mallot and W.Seelen: "Visual obstacle detection for automatically guided vehicles", in Proceedings of 1990 International Conference on Robotics and Automation (ICRA'90), pp.761-766, 1990.
- [14] P.Jasiobedzki: "Detecting Driveable Floor Regions", in Proceedings of International Conference on Intelligent Robots and Systems(IROS'95), pp.264-270, 1995.
- [15] C.E.Thorpe, M.H.Herbert and T.Kanade, "Vision and navigation for the Carnegie-Mellon Navlab", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.10, No.3, pp.362-373, 1988.
- [16] "運輸省 先進安全自動車計画書", 1998.
- [17] 実吉敬二, 植松憲爾, 木下昌裕, 埜圭二, 喜瀬勝之: "三次元画像認識技術を用いた運転支援システム", 自動車技術会学術講演前刷集, pp.169-172, 1992.
- [18] V.Graefe: "Dynamic Vision Systems for Autonomous Mobile Robot", in Proceedings of International Workshop on Intelligent Robots and Systems(IROS'89), pp.12-23, 1989.

- [19] A.M.Waxman, J.J.Lemoigne, L.S.Davis, B.Srinivasan, T.R.Kushner, E.Liang and T.Siddalingaiah: "A visual Navigation System for Autonomous Land Vehicle", IEEE Journal on Robotics and Automation, Vol.RA-3, No.2, pp.124-142, 1987.
- [20] 鈴木敏彦, 青木啓二, 橘彰英, 森部弘, 井上博允: "コンピュータビジョンによる自動運転システムー走行誘導用白線の認識ー", 自動車技術会学術講演前刷集, pp.161-164, 1992.
- [21] 農宗千典: "高速道路走行画像からの消失点推定に基づく実時間白線検出", 電学論(C), Vo.113, No.2, pp.139-148, 1993.
- [22] A.Kutami, Y.Maruya, H.Takahashi and A.Okuno: "Visual Navigation of Autonomous On-Road Vehicle", in Proceedings of International Workshop on Intelligent Robots and Systems(IROS'90), pp.175-180, 1990.
- [23] H.Ishiguro, T.Miyashita and S.Tsuji: "T-Net for Navigating a Vision-Guided Robot in a Real World", in Proceedings of 1995 International Conference on Robotics and Automation(ICRA'95), pp.1068-1073, 1995.
- [24] 松本吉央, 稲葉雅幸, 井上博允: "視野画像列の記憶に基づく移動ロボットのナビゲーションー廊下環境における一姿勢および障害物の検出ー", 第13回日本ロボット学会学術講演会予稿集, pp.313-314, 1995.
- [25] 前田佐嘉志, 久野義徳, 白井良明: "固有空間解析に基づく移動ロボットの位置認識", 電子情報通信学会論文誌(DII), Vol.J80-D-II, No.6, pp.1502-1511, 1997.
- [26] T.Ohno, A.Ohya and S.Yuta: "Autonomous Navigation for Mobile Robots Referring Pre-recorded Image Sequence", in Proceedings of International Conference on Intelligent Robots and Systems(IROS'96), pp.672-679, 1996.

- [27] J.Hong, X.Tan, B.Pinette and R.Weiss: "Image-based Homing", in Proceedings of 1991 International Conference on Robotics and Automation(ICRA'91), pp.620-625, 1991.
- [28] 亀島鉦二, 小川優理子, 中野善之: "再帰型画像処理機構を用いた移動ロボットの視覚誘導", 日本ロボット学会誌, Vol.5, No.5, pp.11-18, 1987.
- [29] C. Thorpe and J. Gowdy : "Annotated Maps for Autonomous Land Vehicles", in Proceedings of the Roundtable Discussion on Vision-Based Vehicle Guidance '90, pp. 4-1 - 4-7, 1990.
- [30] R.A.Brooks: "A Robust Layered Control System for a Mobile Robot", IEEE Journal of Robotics and Automation, Vol.2, No.1, pp14-23, 1986.
- [31] A. Elfse: "Using Occupancy Grids for Mobile Robot Perception and Navigation", Computer, Vol. 22, No. 6, pp 46-57, 1989.
- [32] 久保寺豊, 金山裕, 油田信一: "自立ロボットのための超音波視覚モジュール", 第1回日本ロボット学会学術講演会予稿集, pp.63-66, 1983.
- [33] M.P.Turchan and A.K.C.Wong: "Low level Learning for a Mobile Robots : Environment Model Acquisition", in Proceedings of 2nd AI Application, pp.156-161, 1985.
- [34] U.R.Dhond and J.K.Aggarwal: "Structure from Stereo - A Review", IEEE Trans. on System, Man, and Cybernetics, Vol.19, No.6, pp.1489-1510, 1989,
- [35] T.Kanade, H.Kano and S. Kimura: "Development of a Video-Rate Stereo Machine", in Proceedings of International Conference on Intelligent Robots and Systems(IROS'95), pp.95-100, 1995.
- [36] B.K.P.Horn: "Reflectance Map:Photometric Stereo", Chapter 10 in Robot Vision,The MIT Press, pp.202-242, 1986.

- [37] B.K.P.Horn: "Reflectance Map:Shape from Shading", Chapter 11 in Robot Vision,The MIT Press, pp.243-277, 1986.
- [38] E.Krotkov: "Focusing", International Journal of Computer Vision, Vol.1, pp.223-237, 1987.
- [39] T.Hwang, J.J.Clark and A.L.Yuille: "A Depth Recovery Algorithm Using Defocus Information", in Proceedings of Computer Vision and Pattern Recognition(CVPR'89), pp.476-482, 1989.
- [40] 金出武雄, コンラッド ポールマン, 森田俊彦: "因子分解法による物体形状とカメラ運動の復元", 信学論 (DII), Vol.J76-D-II, No.8, pp.1497-1505, 1993.
- [41] A.Maki, M.Watanabe and C.Wiles: "Geotensity:Combining Motion and Lighting for 3D Surface Reconstruction", in Proceedings of 6th International Conference on Computer Vision(ICC'98), pp.1053-1060, 1998.
- [42] P.Fua: "A Parallel Stereo Algorithm that Produce Dense Depth Maps and Preserves Image Features", Machine Vision and Applications, Vol.6, No.1, pp.35-49, 1993.
- [43] N.Ayache and B.Faverjon: "Efficient Registration of Stereo Images by Matching Graph Descriptions of Edge Segments", International Journal of Computer Vision, pp.107-131, 1987.
- [44] H.S. Lim and T.O.Binford: "Structural Correspondence in Stereo Vision", in Proceedings of Image Understanding Workshop, Vol.2, pp.794-808, 1988.
- [45] Y.Ohta and T.Kanade: "Stereo by intra and inter-scanline search", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.7, No.2, pp.139-154, 1985.
- [46] 山口証, 白井良明, 中山収文, 浅田稔: "信頼性の高い対応を優先した多段階ステレオ法", 信学論 (D-II), VolJ74-D-II, No.7, pp.918-925, 1991.

- [47] T.Q.Luong, J.Weber, D.Koller and J.Malik: "An integrated stereo-based approach to automatic vehicle guidance", in Proceedings of 5th International Conference on Computer Vision (ICCV'95), pp.52-57, 1995.
- [48] M.Xie: "Matching free stereovision for detecting obstacles on a ground plane", Machine Vision and Applications, Vol. 9, No. 1, pp.9-13, 1996.
- [49] Y.Yakimovsky: "A stereo for extracting three-dimensional measurement from stereo pair of TV cameras", Computer Graphics and Image Processing, No.7, pp.195-210, 1978.
- [50] K.Onoguchi, N.Takeda and M.Watanabe: "Planar projection stereopsis method for road extraction", in Proceedings of International Conference on Intelligent Robots and Systems (IROS'95), pp. 249-256, 1995.
- [51] K.Onoguchi, N.Takeda and M.Watanabe: "Planar projection stereopsis method for road extraction", IEICE Transaction on Information and Systems, Vol.E81-D, No.9, pp. 1006-1018, 1998.
- [52] M. Yachida, M. Asada and S. Tsuji: "Automatic analysis of moving image", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.3, No.1, pp.12-20, 1981.
- [53] N. Amamoto and K. Matsumoto: "Obstruction Detector by Environmental Adaptive Background Image Updating", in Proceedings of 4th World Congress on Intelligent Transport Systems, 1997.
- [54] T. Y. Tian and M. Shah: "Recovering 3D Motion of Multiple Objects Using Adaptive Hough Transform", in Proceedings of 5th International Conference on Computer Vision (ICCV'95), pp.284-289, 1995.

- [55] N. Ohta: "Structure from Motion with Confidence Measure and Its Application for Moving Object Detection", IEICE Transaction, Vol.J76-D-II, pp.1566-1571, 1993.
- [56] 寺久保敏, 横矢直和, 岩佐英彦, 竹村治雄: "移動カメラで撮影した動画像からの移動物体の抽出", 情報処理学会第51回全国大会講演論文集, Vol.2, pp.2-237 - 2-238, 1995.
- [57] N. Takeda, M. Watanabe and K. Onoguchi: "Moving Obstacle Detection using Residual Error of FOE Estimation", in Proceedings of International Conference on Intelligent Robots and Systems (IROS'96), pp.1642-1647, 1997.
- [58] M. Inaba, T. Hara and H. Inoue: "A Stereo Viewer Based on a Single Camera with View-Control Mechanisms", in Proceedings of International Conference on Intelligent Robots and Systems (IROS'93), pp.1857-1865, 1993.
- [59] H. Inoue, T. Tachikawa and M. Inaba: "Robot Vision System with a Correlation Chip for Real-time Tracking, Optical Flow and Depth Map Generation", in Proceedings of 1992 International Conference on Robotics and Automation (ICRA'92), pp.1621-1626, 1992.
- [60] M. Ohzora, T. Ozaki, M. Yoshida and Y. Hiratsuka: "Video-Rate Image Processing System for an Autonomous Personal Vehicle System", in Proceedings of IAPR Workshop on Machine Vision and Application (MVA'90), pp.389-392, 1990.
- [61] S. M. Smith, "ASSET-2: Real-Time Motion Segmentation and Shape Tracking", in Proceedings of 5th International Conference on Computer Vision (ICCV'95), pp.237-244, 1995.
- [62] K. Onoguchi, N. Takeda and M. Watanabe: "Obstacle Location Estimation Using Planar Projection Stereopsis Method", in Proceedings of 1997 International Conference on Robotics and Automation (ICRA'97), pp.2533-2558, 1997.

- [63] 小野口一則, 武田信之, 渡辺睦: “ステレオ画像の平面投影による移動障害物位置検出”, 信学論 (DII), Vol.J81-D-II, No.8, pp.1895-1903, 1998.
- [64] 小野口一則, 渡辺睦, 岡本恭一, 久野義徳: “移動視覚のための多重情報地図”, 日本ロボット学会学会誌, Vol.11, No.3, pp.401-409, 1993.
- [65] K. Onoguchi, M. Watanabe, Y. Okamoto, Y. Kuno and H. Asada: “A Visual Navigation System Using a Multi-Information Local Map”, in Proceedings of the 1990 International Conference on Robotics and Automation(ICRA'90), pp.767-774, 1990.
- [66] 小野口一則, 渡辺睦, 麻田治男: “移動ロボットの位置確認システム”, 情処 CV 研資, 57-3, 1988.
- [67] 渡辺睦, 小野口一則, 星野弘: “ステレオ視を用いた移動視覚システム—実験システムの試作”, 情処 CV 研資, 42-2, 1986.
- [68] 小野口一則, 穂積久士, 岡田敏, 中山良一, 渡邊睦: “視覚を用いた移動ロボットの走行制御”, 第6回日本ロボット学会学術講演会資料, pp. 543-546, 1988.
- [69] R. Nakayama, K. Sato, S. Okada, H. Hozumi, A. Abe and H. Okano: “Development of Mobile Maintenance Robot System AIMARS”, in Proceedings of USA-JAPAN Symposium on Flexible Automation, pp. 645-650, 1988.
- [70] 石川実, 星野弘: “ロボット視覚用イメージプロセッサ”, 電子情報通信学会全国大会予稿集, D-195, 1988.
- [71] 渡辺睦, 小野口一則, 久野義徳, 麻田治男: “視差予測ステレオ法を用いた障害物検出”, 信学論 (D-II), Vol.J73-D-II, No.6, pp. 862-870, 1990.
- [72] 穂積久士, 岡田敏, 中山良一, 小野口一則, 渡邊睦, 岡本恭一: “視覚情報を用いた移動経路生成および移動実験”, 第7回日本ロボット学会学術講演会資料, pp. 611-612, 1989.

- [73] F.R.Noreils and R.Prajoux: “From Planning to Execution Monitoring Control for Indoor mobile Robots”, in Proceedings of 1991 International Conference on Robotics and Automation(ICRA'91), pp.1510-1517, 1991.
- [74] R.Hartley and F.Pipitone: “Experiments with the Subsumption Architecture”, in Proceedings of 1991 International Conference on Robotics and Automation(ICRA'91), pp.1652-1658, 1991.
- [75] P.Maes: “The Dynamics of Action Selection”, in Proceedings of 11th International Joint Conference on Artificial Intelligence(IJCAI'89), pp.991-997, 1989.
- [76] 渡辺睦, 小野口一則, 久野義徳, In So Kweon: “群化マルチエージェント構成による移動ロボット制御”, 日本ロボット学会誌, Vol.13, No.3, pp.375-382, 1995.
- [77] I.S.Kweon, Y. Kuno, M. Watanabe and K. Onoguchi: “Sonar-Based Behaviors for a Behavior-based Mobile Robot”, IEICE Transaction on Information and Systems, Vol.E76-D, No.4, pp.479-485, 1993.
- [78] M. Kass, A. Witkin and D. Terzopoulos: “Snakes: Active Contour Models”, International Journal of Computer Vision, Vol.1, No.4, pp.321-331, 1988.
- [79] K. Onoguchi and M. Watanabe: “Qualitative Navigation Using Target Tracking”, in Proceedings of 1993 International Conference on Advanced Mechatronics(ICAM'93), pp.866-871, 1993.
- [80] I. S. Kweon, Y. Kuno, M. Watanabe and K. Onoguchi: “Behavior-based Mobile Robot Using Active Sensor Fusion”, in Proceedings of 1992 International Conference on Robotics and Automation(ICRA'92), pp.1675-1682, 1992.
- [81] 小野口一則, 石川実, 渡辺睦: “自律移動車を用いた環境理解の研究—色情報を用いた目標物追跡”, 情報処理学会第41回全国大会予稿集, Vol.2, pp.184-185.

著者論文リスト

(a) 雑誌

- (1) 小野口一則, 渡辺睦, 岡本恭一, 久野義徳: “移動視覚のための多重情報地図”, 日本ロボット学会誌, Vol.11, No.3, pp.99-107, 1993.
- (2) 小野口一則, 武田信之, 渡辺睦: “ステレオ画像の平面投影による移動障害物位置検出”, 信学論 (DII), Vol.J81-D-II, No.8, pp.1895-1903, 1998.
- (3) K.Onoguchi, N.Takeda and M.Watanabe: “Planar projection stereopsis method for road extraction”, IEICE Transaction on Information and Systems, Vol.E81-D, No.9, pp.1006-1018, 1998.
- (4) 渡辺睦, 小野口一則, 久野義徳, 麻田治男: “視差予測ステレオ法を用いた障害物検出”, 信学論 (DII), Vol.J73-D-II, No.6, pp.862-869, 1990.
- (5) I. S. Kweon, Y. Kuno, M. Watanabe and K. Onoguchi: “Sonar-Based Behaviors for a Behavior-Based Mobile Robot”, IEICE Transaction on Information and Systems, Vol.E76-D, No.4, pp.479-485, 1993.
- (6) 渡辺睦, 小野口一則, 久野義徳, I. S. Kweon: “群化マルチエージェント構成による移動ロボット制御”, 日本ロボット学会誌, Vol.13, No.3, pp.375-382, 1995.
- (7) M.Watanabe, N.Takeda and K. Onoguchi: “Moving Obstacle Detection and Recognition by Optical Flow Pattern Analysis for Mobile Robots,” Advanced Robotics, Vol.12, No.8, pp.791-816, 1999.

(b) 国際会議予稿集

- (1) K. Onoguchi, M. Watanabe, Y. Okamoto, Y. Kuno and H. Asada: “A Visual Navigation System Using a Multi-Information Local Map”, in Proceedings of 1990 International Conference on Robotics and Automation(ICRA'90), pp.767-774, 1990.
- (2) K. Onoguchi, N. Takeda and M. Watanabe: “Planar Projection Stereopsis Method for Road Extraction”, in Proceedings of International Conference on Intelligent Robots

and Systems(IROS'95), pp.249-256, 1995.

- (3) K. Onoguchi, N. Takeda and M. Watanabe: “Obstacle Location Estimation Using Planar Projection Stereopsis Method”, in Proceedings of 1997 International Conference on Robotics and Automation(ICRA'97), pp.2553-2558, 1997.
- (4) K. Onoguchi: “Shadow Elimination Method for Moving Object Detection”, in Proceedings of International Conference on Pattern Recognition(ICPR'98), pp.583-587, 1998.
- (5) M. Watanabe, K. Onoguchi, Y. Kuno, H. Hoshino and S. Tsunekawa: “Obstacle Detection Method for Mobile Robots with Stereo Vision”, in Proceedings of 5th Scandinavian Conference on Image Analysis(SCIA'87), pp.325-334, 1987.
- (6) M. Watanabe, K. Onoguchi, I. S. Kweon and Y. Kuno: “Architecture of Behavior-based Mobile Robot in Dynamic Environment”, in Proceedings of 1992 International Conference on Robotics and Automation(ICRA'92), pp.2711-2718, 1992.
- (7) I. S. Kweon, Y. Kuno, M. Watanabe and K. Onoguchi: “Behavior-based Mobile Robot Using Active Sensor Fusion”, in Proceedings of 1992 International Conference on Robotics and Automation(ICRA'92), pp.1675-1682, 1992.
- (8) I. S. Kweon, Y. Kuno, M. Watanabe and K. Onoguchi: “Behavior-based Intelligent in Dynamic Indoor Environment”, in Proceedings of International Conference on Intelligent Robots and Systems(IROS'92), pp.1339-1346, 1992.
- (9) M. Watanabe, N. Takeda and K. Onoguchi: “A Moving Object Recognition Method by Optical Flow Analysis”, in Proceedings of International Conference on Pattern Recognition(ICPR'96), pp.528-533, 1996.
- (10) N. Takeda, M. Watanabe and K. Onoguchi: “Moving Obstacle Detection using Residual Error of FOE Estimation”, in Proceedings of International Conference on Intelligent Robots and Systems(IROS'96), pp.1642-1547, 1996.
- (11) K. Onoguchi and M. Watanabe: “Qualitative Navigation Using Target Tracking”, in Proceedings of International Conference on Advanced Mechatronics(ICAM'93),

pp.866-871, 1993.

- (12) K. Onoguchi, M. Watanabe, Y. Okamoto and H. Asada: "Visual Navigation System for a Mobile Robot", International Workshop on Intelligent Robots and Systems(IROS'89), pp.590-597, 1989.

(c) 解説

- (1) 小野口一則: "見つめて動く…注視点追跡による移動制御", 画像ラボ, Vol.3, No. 4, pp.47-50, 1992.
- (2) 小野口一則: "極限作業ロボット用情報処理システム", 東芝レビュー「保全点検ロボット/グローバルネットワークシステム」特集, Vol.46, No.7, pp.557-560, 1991.
- (3) 小野口一則, 武田信之, 渡辺睦: "車載画像認識技術—ステレオ画像の平面投影による移動障害物位置検出", 画像ラボ, Vol.10, No. 4, pp.33-37, 1999.

各章と主要論文の関係

- 1章:
 - 雑誌 (1),(2),(3),(5),(6)
 - 国際会議予稿集 (1),(2),(3),(6),(7),(8),(10),(12)

- 2章:
 - 雑誌 (3)
 - 国際会議予稿集 (2)

- 3章:
 - 雑誌 (2)
 - 国際会議予稿集 (3)
 - 解説 (3)

- 4章:
 - 雑誌 (1)
 - 国際会議予稿集 (1),(11)
 - 解説 (2)

- 5章:
 - 雑誌 (5),(6)
 - 国際会議予稿集 (6),(7),(8),(11)
 - 解説 (1)

- 6章:
 - 雑誌 (1),(2),(3),(5),(6)
 - 国際会議予稿集 (1),(2),(3),(6),(7),(8),(10),(12)

