

## 分散視覚システムによる移動ロボットの誘導

十河 卓司\* 木元 克美\* 石黒 浩\* 石田 亨\*

### Mobile Robot Navigation by a Distributed Vision System

Takushi Sogo\*, Katsumi Kimoto\*, Hiroshi Ishiguro\* and Toru Ishida\*

This paper proposes a general infrastructure for robot navigation in an outdoor environment. The infrastructure, called a *Distributed Vision System*, consists of vision agents connected with a computer network, monitors the environment, maintains the environment models, and provides various information for robots by organizing communication between the vision agents. As the first step of our research, we have developed a prototype of the distributed vision system for navigating mobile robots. The experimental results show that the system has an ability to navigate the robots in a complex environment.

**Key Words:** Distributed Vision System, Vision Agent, Mobile Robot Navigation

#### 1. はじめに

これまで、視覚情報に基づいて実世界で自律的に移動するロボットに関して数多くの研究がなされており、研究室や工場など、屋内の限られた環境のもとで行動するロボットはいくつかが開発されている。しかし、環境の変化が激しい屋外などで自律的に移動するロボットを実現するのは依然として困難である。

能動視覚の研究[1]で議論されているように、視覚情報に基づいて行動するロボットの問題は注意制御にある。移動可能な領域や障害物の発見など、様々な視覚タスクを複雑な環境下で同時に遂行するために、ロボットは様々な事象に対して適切に注意を切り替え情報を収集する注意制御の機構を持たなければならない。単一の視覚しか持たない自律移動ロボットが複数の視覚タスクを並列に実行するには、実行するタスクを時間的に切り替える必要がある(時間的注意制御)。また、自律移動ロボットが実行する必要がある様々な視覚タスクを遂行するためには、視覚タスクに応じた情報を得るために最も適した視点を選択する必要がある(空間的注意制御)。

しかし、このような注意制御を実現するのは、現在のロボットでは困難である。その原因としては、次のものが挙げられる。

- 従来のロボットは視覚センサが台車に固定されており、適切な視点からの視覚情報を得るのが難しい。
- 単体のロボットでは、広範囲に渡る動的な環境の首尾一貫したモデルを維持するのが難しい。

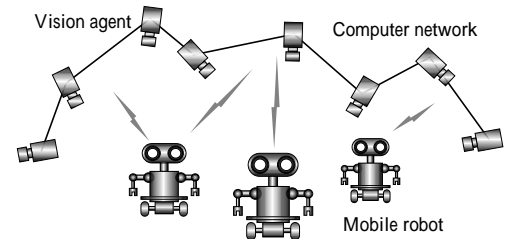


Fig.1 Distributed Vision System

本論文では、これらの問題を解決し、屋外のように複雑で動的な実環境における移動ロボットの誘導を可能にする分散視覚システム(DVS: Distributed Vision System)を提案する。DVSは、ネットワークによって接続された、実環境に埋め込まれた多数の視覚エージェント(VA: Vision Agent)によって構成される(Fig.1)。各VAはその周囲の環境を観測し、ネットワークを通じて相互に情報を交換することで、ロボットが移動するのに必要な情報を提供する。

本論文で提案するDVSは、

- 注意制御の問題を多数のVAの選択という形で解決し、屋外などの複雑な環境下で視覚情報に基づいたロボットの誘導を可能にする
- 冗長な数の視覚エージェントによって構成される、ロバストかつ柔軟な分散システム
- 実世界で行動するロボットや人間を支援するための新しい情報基盤となる可能性を持つ

という特徴を持つ。DVSを開発する上では、

原稿受付 1998年X月XX日

\*京都大学大学院情報学研究科社会情報学専攻

\*Department of Social Informatics, Kyoto University

(1) 各 VA によるロボットの誘導方法  
(2) VA 間のコミュニケーション  
(3) VA のネットワーク上での環境モデルの管理  
等が重要となる点となるが、ここでは特に(1)を中心にシステムの試作について議論する。以下では、分散視覚の概念、試作した DVS のアーキテクチャ、ロボットの誘導手法、および屋内に構築した街のモデルを用いて行ったロボットの誘導実験について述べる。

## 2. 分散視覚システム

### 2.1 分散視覚の概念

視覚情報に基づいて移動するロボットは、前述のように移動可能な領域や障害物の発見など様々な視覚タスクを同時に遂行するために注意制御の機構を持たなければならないが、現在のロボットでこれを実現するのは困難である。

この問題を解決する一つの方法は、ネットワークで結ばれた多数の VA を環境の随所に配置し、それら多数の VA によって環境の様々な側面を同時に観測することである。各 VA はそれぞれが独立に周囲の環境を観測し、ネットワークを介して互いに通信し、ロボットに情報を提供する。例えば DVS がロボットをある目的地へ誘導する場合、VA はそれぞれの視点からロボットの周囲の環境(ロボット自身、道路や障害物など)を観測し、互いに通信を行うことによってロボットの誘導経路の計画などを行い、適切な情報が得られる VA を次々と切り替えながらロボットを誘導する。多数の人、車やロボットなどが行き交う交差点などではロボットを誘導するためには多くの視覚情報が必要となるが、必要な数の VA を必要な場所に設置することにより、ロボットはタスクを達成するのに十分な視覚情報を得ることができる。また、VA は環境に固定されているため、静止した視点から動的に変化する環境を安定に観測することができ、各 VA がその周囲の状況を観測することで、環境の首尾一貫したモデルを維持、管理することができる。このように、DVS は VA の選択という形で注意制御の問題を解決し、屋外などの複雑で動的な実環境における移動ロボットの誘導を可能にする。このような多数の視覚を用いた視覚システムを、分散視覚と呼ぶ。分散視覚に関しては[2]において詳細に議論されている。

### 2.2 関連研究

近年、ロボットの分野で多数のセンサを用いた研究が行われている。例えば、Robotic Room[3]は部屋にセンサやロボットを設置しておき、人間の活動を支援するものである。この研究では、決められたタスクを実行するための機構やセンサシステムを開発することを目的としている。一方、本研究では、様々なセンサを用い、人やロボットに情報を提供する情報基盤としての柔軟なセンサシステムを提案することである。

多数のカメラを用いた研究は、特にマルチメディアの分野でなされている。Immersive Video[4]は、正確な位置が計測された複数のカメラを用いて屋外環境における人などの正確な幾何モデルを復元するシステムである。また、TV番組の放送において、スタジオに取りつけられた様々なカメラによって場面を認識し、適切な視点のカメラを選択して放送するシステムの研究[5]もある。これらの研究の問題点は、正確な位置が計測されたカメラや

幾何的なモデルを用いているため、システムのロバスト性や柔軟性を欠くことである。この問題を解決するため、本研究では後述のように動的かつ局所的にカメラパラメータを推定し、ロボットのタスクを直接表現する手法をとる。

また、分散人工知能の分野では、複数のセンサを用いたシステムに関する基礎的な研究が行われている。Distributed Vehicle Monitoring Testbet (DVMT)[6]では、分散センサネットワークによる乗り物の監視システムが分散問題解決の例として扱われている。Partial Global Planning[7]では、複数のエージェントから得られる信号を大域的に分析するためのプランニング手法が提案されている。DVS はこのような分散センシングシステムの一つとみなすことができるが、これらのシステムがプランニングアルゴリズムの問題を扱っているのに対し、DVS ではノイズの多い視覚センサ情報をもとに、いかに VA が協調してロボットを誘導するかという、センサ利用の問題を扱っているところが異なる。

### 2.3 設計方針

DVS は、

ロボットのタスクは局所的な環境と密接に関わるという考え方に基づいて設計される。例えば、ロボットがある目標地点に近接するというタスクを実行する際、そのタスクは目標付近の局所的な環境に密接に関係している。DVS においては、複雑な環境内でロボットが行動するために必要な情報は環境に埋め込まれた VA が環境を観測することによって提供されるが、個々の VA は固定されているため、例えば複雑な背景からの物体の切り出しや、日照変化による物体の色の変化に対する処理など、その局所的な環境に特化した処理を行うことによって、観測対象を容易に、かつロバストに認識することができる。

Immersive Video[4]など複数のカメラを用いる研究のほとんどは、非常に正確に測定されたカメラの位置を基に情報処理を行っている。一方 DVS では、VA の正確な位置はあらかじめ計測せず、個々の VA が環境を繰り返し観測することによって、各 VA に局所的な座標系における位置を推定する。このように VA の正確な位置をあらかじめ計測しないのは、システムに柔軟性を持たせるためである。例えば、DVS では VA の配置は重要な問題の1つであるが、あらかじめ必要十分な配置を行うことは困難であり、必要に応じて VA を配置しなおすことのできる柔軟性が求められる。しかし、カメラの正確な位置が既知であるという前提を設けることは、DVS においてはシステムの柔軟性を大きく欠くことになる。従って、DVS では VA の位置はあらかじめ計測されていないという前提でシステムを設計する。

ロボットを誘導するためにシステムに地図を与えておくという手法は一般的である。しかし、このような幾何モデルに基づく手法は、一般に画像情報と結び付けることが難しく、例えば屋外環境のように複雑な環境では道路を認識することも難しい。そこで、DVS では地図を与えるのではなく、ロボットの行動をタスクとして直接教示し、それに基づいてロボットを誘導する。すなわち、DVS によるロボットの誘導はタスクの教示とロボットの誘導の2段階で行われる。タスクの教示においては、人間がロボットを動かして見せたり、システムが環境中を移動する物体を観測することにより、環境中の様々な場所に設置された VA はロ

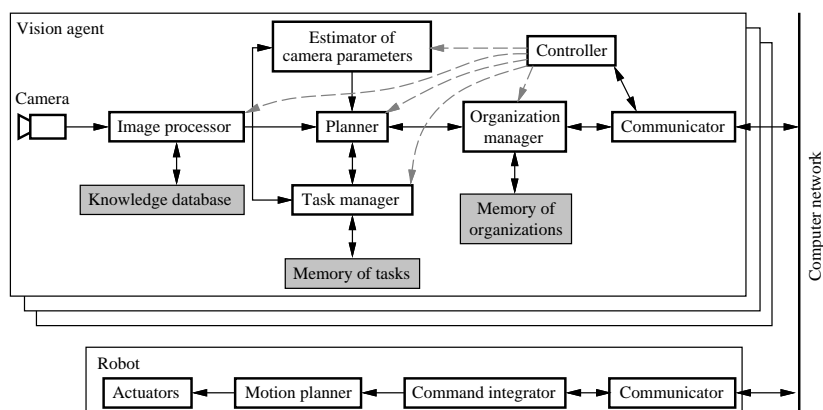


Fig. 2 The architecture of the DVS

ロボットを誘導するためのタスクとしてその運動を記憶する。このとき VA の正確な位置は計測されていないため、各 VA は画像上の移動軌跡を記憶する。ロボットの誘導の段階では、VA は画像上の軌跡として記憶されたタスクをもとに互いに通信することによってロボットの誘導計画を生成する。

DVS では VA が環境を繰り返し観測することによってその位置を推定するが、視覚センサの観測誤差のために正確な位置を推定することは難しく、ロボットの誘導にも影響を与えかねない。この誤差の問題は、上記のような簡単なモデルをロボットの誘導に用い、多くの VA の中から適切な視点でロボットを観測している VA のみを（複数）選択して用いることで解決される。

また、多数の VA による観測や誘導を行う際には VA 間の同期をとる必要がある。このため、例えば各 VA が正確に同期した時計を持つと仮定することが考えられるが、これはシステムの柔軟性を損うことになる。DVS ではこのような仮定をせず、VA 同士の、ネットワークを介したメッセージ交換によって同期をとるものとする。

#### 2.4 アーキテクチャ

ロボットを誘導するための DVS 全体のアーキテクチャを Fig.2 に示す。システムは複数の VA (Vision agent) とロボット (Robot)、それらを接続するコンピュータネットワーク (Computer network) によって構成される。

Image processor はロボットやその他の物体を検出する。このとき、Knowledge database に記憶された背景画像、道路領域やロボットの色などの知識を用いると同時に、これらを適宜更新する。Estimator of camera parameters は、観測されるロボットの動きをもとにカメラパラメータを推定する。Task manager は、ロボットなどの移動軌跡を教示されたタスクとして Memory of tasks に記憶する。また、ロボットから提示されたタスクを達成するために、記憶されたタスクの中から適切なものを選択する。Planner は、記憶されたタスク、観測されたロボットの運動やカメラパラメータをもとに、ロボットが提示したタスクを達成するための誘導計画を生成する。Organization manager は他の VA と通信し、教示段階や誘導計画の生成においてその VA を用いるのが適切かどうかを判断し、他の VA とともに組織を構成する。また、その組織を Memory of organizations に記憶す

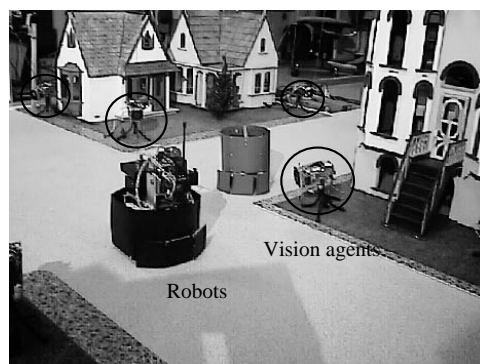


Fig. 3 A prototype of the DVS

る。Communicator は VA と他の VA、あるいは VA とロボットの間の通信を担う。Controller はシステムの状態（教示、誘導）やロボットの誘導要求に応じて、各モジュールを適切に制御する。

ロボット側では、ロボットがシステムにタスクを提示することで各 VA によって生成された複数の誘導計画が Communicator を介して受信され、さらに Command integrator によって適切に取捨選択、統合される。その結果をもとに Motion planner がアクチュエータコマンドを生成し、Actuator を制御する。

### 3. システムの試作

前述のアーキテクチャに基づいて、室内に分散視覚システムを試作した (Fig.3)。システムは 16 台の VA によって構成され、2 台のロボットを同時に誘導することができる。このシステムについて、教示、誘導の各段階ごとにその詳細を述べる。

#### 3.1 ロボットのタスクと教示

ロボットの誘導においては、

- (1) 移動可能な道路領域上を移動する。
- (2) 他のロボットや移動物体との衝突を回避する。
- (3) 目的地に向かう。

という機能が必要となる。分散視覚システムではこれらすべてを、環境に設置された複数の VA からの情報を選択的に利用する

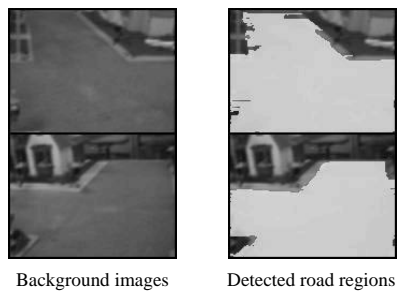


Fig. 4 Detecting road regions

ことによって実現する。(1)については、ロボットが移動可能な道路領域をVAが認識し、ロボットがその情報を利用することで、ロボットは例えば道路からはみ出さないように移動することができる。この道路領域の知識は、VAが環境を観測することによって得ることができる。すなわち、移動物体が通るところが道路であると仮定し、環境中の様々な移動物体を長時間観測すれば、その通過した領域を道路であるとみなすことができる。試作したシステムでは、後述のタスクの教示の際にロボットが通過した部分と同じ色の領域を道路領域として認識している(Fig.4)。

(2)については、ロボットや他の障害物同士の衝突の可能性をVAが判断し、衝突しそうな場合には回避するように誘導する。今回の実験では、VAによって検出されたロボットの進行方向などをもとにロボット同士が衝突するかどうかチェックし、その可能性がある場合には後述(3.2節)のように誘導目標を一時的に修正することによって衝突を回避する。しかし、ロボット同士が非常に接近した場合などには、適切な視覚情報が得られない可能性があるため、ロボット自身が持つ接触センサなども用いて衝突を回避する。(3)については、誘導に必要な経路の知識をオペレータがシステムに教示し、その知識を利用してロボットを誘導する。以下では、この目的地に向かうというタスクの教示について述べる。

オペレータの指示によってシステムは教示段階に移り、タスクを記憶する。各VAはオペレータが操作するロボットを追跡し、画像上の軌跡を教示されたタスクとして記憶する。このときVAは、ロボットを誘導するために、教示されたタスクの意味付けを行う必要がある。この意味付けは、オペレータが明示的に与える場合と、システムがタスクの構造を解析して獲得する場合がある。例えば、家の前、ポストなどという、特定の場所の名前に関する知識等はオペレータが明示的に与える必要がある。ロボットがそのような地点を通過するとき、オペレータがロボットを観測しているVAに明示的に通知する。一方、システムがタスクの構造を解析して得られる知識は、誘導に直接かかわる知識、すなわち、経路の構造等である。今回の実験では、交差点に関する情報がそれに相当する。教示段階においてVAがある地点でロボットが複数の方向に誘導されていることを観測したとき、VAはそこに交差点があると判断する。システムはこの情報を基に、ロボットのタスクを交差点間の移動というサブタスクに分解し、それらサブタスクを組み合わせることによって(経路計画)、ロボットを自由に誘導することができる。もっとも、今回の実験で対象と

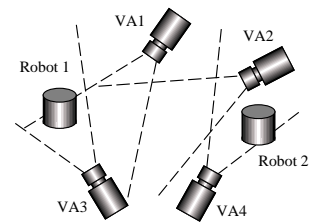


Fig. 5 Overlaps of visual fields of VAs

した環境には、2つの交差点しか存在しない。またロボットを連続的に誘導し、システムのロバスト性を検証するという目的から、ロボットの全体タスクを、それらの交差点をランダムに選択して、交差点間の移動を繰り返すというものに設定している。また、タスクの教示においても、交差点間の移動を教示することによって上記のサブタスクを直接与えている。2地点間の移動を交差点間の移動というサブタスクに分解し、それらを組み合わせて経路計画を行うという部分に関しては、今後より広い環境においてその機能を検証する予定である。

次に、タスクの教示の具体的な処理について述べる。まず各VAは視覚センサ画像からロボットを検出する。VAは固定されているので、物体が存在しないときの画像を記憶しておき、その画像と現在の入力画像との差分を取ることで環境中を移動する物体を容易に検出できる。そして、検出した物体がどのロボットなのか、あるいはロボット以外の物体なのかを認識する。今回試作したシステムでは色によってロボットの認識を行っている。但し、ロボットの色は個々の視覚センサの特性や照明条件などによって微妙に異なる色となって観測されるため、各VAの周囲の環境に合わせて調整する必要がある。今回の実験では、2台のロボットの誘導を行うため、あらかじめ2台のロボットをVAに見せ、VAごとに閾値などのパラメータを決定する(基本的には誘導するロボットの台数に制限はない)。

このとき、各VAの視野の重なりを求めておけば、VAがロボットでない物体を誤ってロボットと認識した場合などに、それを除去することができる。ここで言う視野の重なりとは、厳密に測定したカメラの配置などに基づくものではない。例えばFig.5の場合、VA1とVA4、およびVA3とVA4は視野が重なっていないが、もし全てのVAで同じ特徴のロボットが観測された場合は、VA4の観測したロボットが他とは異なるものである可能性が高いと判断できる。試作したシステムでは、タスクの教示の際に環境中にロボットは1台のみ存在すると仮定してVAの視野の重なりを求め、誘導の際にそれを用いて誤認識したと思われるロボットを除去している。さらに、環境の知識を導入することによって、よりロバストにロボットを認識することができる。例えば、Fig.4のようなVAの視覚センサ画像における道路領域の知識を用いれば、ロボットは道路上にのみ存在するという仮定に基づいて、検出された物体の下端が道路領域にあるもの以外はロボットとして認識しないようにすることができる。

以上のようにしてロボットが検出されると、各VAはそのロボットを追跡し、画像上の軌跡(実際は点列)を教示されたタス

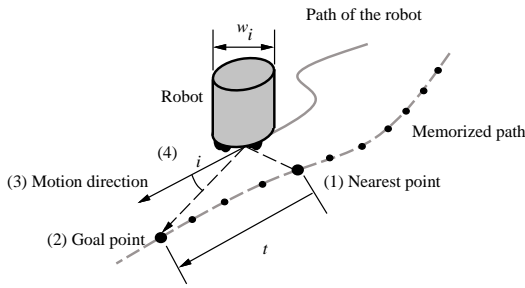


Fig. 6 Generating a navigation plan

クとして記憶する．特定の地点をロボットが通過する時には，オペレータがそれをシステムに通知することで，場所の意味付けを行う．また，システムは教示された経路を交差点間の移動というサブタスクに分解する．但し今回の実験では，前述のようにサブタスクを直接教示している．

### 3.2 誘導

タスクの教示を行った後，DVS は次の手順を繰り返すことによりロボットを誘導する．

- (1) ロボットがシステムにタスク（ここでは目的地）を提示する．
- (2) 各 VA は画像上のロボットの位置をもとに誘導目標を設定する．さらに誘導計画を生成し，それをロボットに通知する．
- (3) ロボットは複数の VA から受け取った誘導計画を選択，統合し，移動する．

処理の詳細を以下に述べる．

システムがタスクを提示したロボットの誘導計画を立てるためには，まずそのロボットが画像上のどのロボットであるかを同定しなければならない．ここでは，先に述べた教示段階と同様にロボットの同定を行う．

次に各 VA は誘導計画を生成する．提示された目的地にロボットを誘導するために，まずシステムは教示によって得られた経路（サブタスク）のうち適切なものを選択し，組み合わせる．次に，現在のロボットの位置から最も近い，選択された経路上の位置を推定する．ここでは，教示された経路は点列として記憶されており，それらのうちロボットに最も近い点を推定する（Fig.6 (1)）．次に，その地点からある一定の時間  $\Delta t$  だけ先の地点を誘導目標に設定し（Fig.6 (2)），その目標地点と現在のロボットの進行方向（Fig.6 (3)）との間の見かけの角度  $\theta_i$ （Fig.6 (4)）を求め，さらに次式によってロボットの移動平面における角度  $\theta_i^*$  を計算する．

$$\theta_i^* = \frac{\theta_i}{\sin \alpha_i} \quad (1)$$

ここで， $\alpha_i$  は VA  $i$  の視覚センサの俯角（視線と水平面とのなす角）である．VA はこの  $\theta_i^*$ ，すなわち現在の移動方向と目標地点のずれ角を誘導計画としてロボットに送る．

ロボットの目標地点を設定するとき，その進路にあたる道路領域上に障害物が検出された場合には，それを回避するように目標地点を修正する．ここでは誘導を行うロボット以外のロボットも障害物とみなすことで，障害物と同様にして衝突を回避する．

次にロボットは複数の VA がそれぞれ生成した誘導計画，すなわち現在の移動方向と目標地点のずれ角  $\theta_i^*$  を統合し，行動する．このとき  $\theta_i^*$  の誤差を推定し，これを考慮して  $\theta_i^*$  を統合することで，適切な位置からロボットを観測している VA の生成した誘導計画のみを選択することができる． $\theta_i^*$  の誤差には，ロボットの移動方向の観測誤差と視覚センサの俯角  $\alpha_i$  の誤差が含まれるが，ロボットの移動方向の観測誤差はロボットの見かけの大きさ  $w_i$  に反比例すると仮定し， $\alpha_i$  の誤差による  $\theta_i^*$  の誤差  $\Delta \theta_i^*$  は， $\alpha_i$  の推定誤差  $\Delta \alpha_i$ （3.3.2節）と式（1）をもとに次式により求める．

$$\Delta \theta_i^* = \frac{\theta_i}{\sin(\alpha_i + \Delta \alpha_i)} - \frac{\theta_i}{\sin \alpha_i} \quad (2)$$

ロボットは，まず VA が生成した誘導計画のうち誤差を多く含むと思われるものを除外する．ここでは，VA  $i$  によって観測されたロボットの見かけの大きさ  $w_i$  がそれらの最大値の  $2/3$  未満である場合に，その VA  $i$  の生成した誘導計画  $\theta_i^*$  は他のものと比べて誤差が相対的に大きいとみなし，除外する．さらに，残った誘導計画のうち  $\Delta \theta_i^*$  がそれらの最小値の 2 倍以上である VA  $i$  の生成した誘導計画も除外する．

次に，残った誘導計画を統合する．複数の VA が生成した誘導計画は，ロボットの移動方向を基準とした共通の座標系で表現されているため，その平均を求めることで統合することができる．ここでは，誘導計画  $\theta_i^*$  の誤差の大きさに応じた重みつき平均  $\theta^*$  を求め，統合された誘導計画とする．具体的には，次式のようにロボットの見かけの大きさ  $w_i$  と， $\alpha_i$  の推定誤差  $\Delta \alpha_i$  による  $\theta_i^*$  の誤差  $\Delta \theta_i^*$  を用いて求める．

$$\theta^* = \frac{\sum_i k_i \theta_i^*}{\sum_i k_i}, \quad k_i = \frac{w_i}{|\Delta \theta_i^*|} \quad (3)$$

この  $\theta^*$  をもとにして，ロボットのアクチュエータコマンドを生成する．

### 3.3 観測によるカメラパラメータの推定

一般に，視覚センサの位置は回転および平行移動の計 6 つのパラメータで表現される．細田らはこれらのパラメータを視覚フィードバックによって求める手法 [8] を提案しているが，本研究でも同様に環境中を移動するロボットを観測することで視覚センサのパラメータを推定する．しかし DVS では，画像上のロボットの下端をロボットの位置とみなしており，その計測は非常に不安定であるため，6 つのパラメータ全てを推定することは難しい．そこで，平行射影を仮定した時に必要となるパラメータのみ（Fig.7 の  $\alpha_i$ ,  $\beta_i$ ,  $\gamma_i$ ）をオンラインで推定する．これらのパラメータを用いることで，前述のように VA の視覚センサの画像平面における表現（ $\theta_i$ ）をロボットの移動平面において表現された誘導計画（ $\theta_i^*$ ）に変換することができる．

#### 3.3.1 推定手法

Fig.7 のように， $x$ ,  $y$  をある瞬間のロボットの進行方向とした直交座標系， $\alpha_i$  を VA  $i$  の視覚センサの俯角， $\beta_i$  を視覚センサと  $y$  軸との間の角度， $\gamma_i$  を視覚センサの視軸周りの回転角度とする．平行射影を仮定すれば，ロボットの速度ベクトル  $V$  は，次のように VA  $i$  の画像上のベクトル  $V_i = (u_i, v_i)^T$  に投影される．

$$V_i = S_i T_i R_i V \quad (4)$$

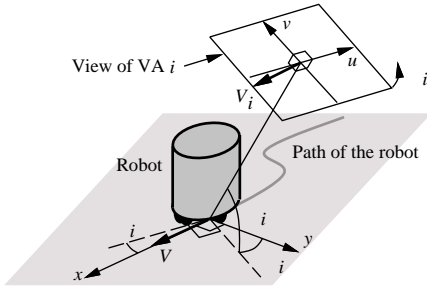


Fig.7 Estimating camera parameters

ここで,  $R_i$ ,  $S_i$  はそれぞれ角度  $\beta_i$ ,  $-\gamma_i$  の回転,  $T_i$  は平行射影を表す行列であり, 次のように表される.

$$R_i = \begin{pmatrix} \cos \beta_i & -\sin \beta_i \\ \sin \beta_i & \cos \beta_i \end{pmatrix} \quad (5)$$

$$S_i = \begin{pmatrix} \cos \gamma_i & \sin \gamma_i \\ -\sin \gamma_i & \cos \gamma_i \end{pmatrix} \quad (6)$$

$$T_i = \begin{pmatrix} 1 & 0 \\ 0 & \sin \alpha_i \end{pmatrix} \quad (7)$$

これから速度ベクトル  $V$  は  $V_i = (u_i, v_i)^T$  を用いて次のように表される.

$$V = \begin{pmatrix} u'_i \cos \beta_i + \frac{v'_i}{\sin \alpha_i} \sin \beta_i \\ -u'_i \sin \beta_i + \frac{v'_i}{\sin \alpha_i} \cos \beta_i \end{pmatrix} \quad (8)$$

但し,

$$u'_i = u_i \cos \gamma_i - v_i \sin \gamma_i \quad (9)$$

$$v'_i = u_i \sin \gamma_i + v_i \cos \gamma_i \quad (10)$$

とする. これらから,

$$V^2 = u_i'^2 + \left( \frac{v'_i}{\sin \alpha_i} \right)^2 \quad (11)$$

となる. オペレータがロボットを一定速度で操作する場合は  $|V|$  が既知であるから, 上式より

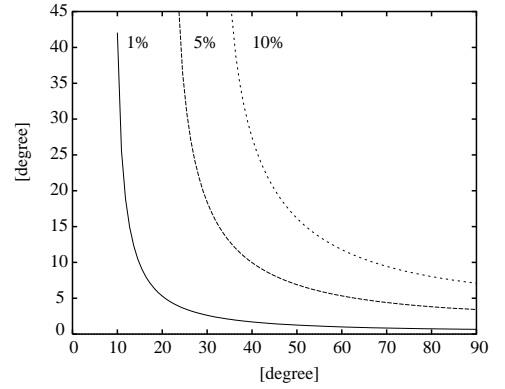
$$\sin \alpha_i = \sqrt{\frac{v_i'^2}{V^2 - u_i'^2}} \quad (v'_i \neq 0) \quad (12)$$

となる. さらに, 座標軸の取り方から  $V$  の  $y$  成分は常に 0 であるから, 式 (8) より

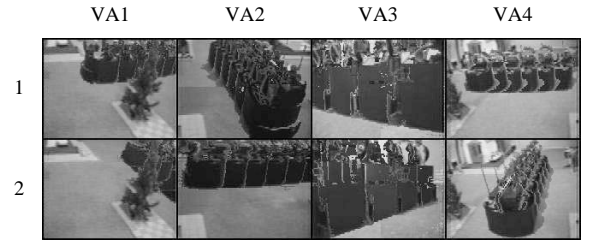
$$u'_i \sin \beta_i - \frac{v'_i}{\sin \alpha_i} \cos \beta_i = 0 \quad (13)$$

となる. 式 (9), (10), (12), (13) から, ロボットの速度を 2 回観測する (2 つの異なる  $V_i$  を観測する) ことによって,  $\alpha_i$ , (2 つの異なる)  $\beta_i$ ,  $\gamma_i$  を求めることができる. 但し, 今回の実験では簡単のため  $\gamma_i = 0$ , すなわちカメラは地面と平行であると仮定している. これにより, 1 回の観測によって式 (12) から  $\alpha_i$  を, 式 (13) から  $\beta_i$  を求めることができる.

なお, ロボットの見かけの速度  $V_i$  として, 実際には観測された速度をそのまま用いるのではなく, 画像上のロボットの見かけの大きさ  $w_i$  で正規化したものを用いる.

Fig.8 Error range of  $\alpha_i$ Table 1  $\alpha_i$  of vision sensors acquired by observation

VA	VA1	VA2	VA3	VA4
Actual	30	31	9	28
Observation 1	21.7*	35.8	30.4*	6.49*
Observation 2	24.9	8.08*	16.9*	34.4

Fig.9 Images used for estimating  $\alpha_i$ 

### 3.3.2 観測誤差

ロボットの速度の観測誤差  $\Delta u_i$ ,  $\Delta v_i$  とカメラの俯角の誤差  $\Delta \alpha_i$  の関係は, 式 (12) と仮定  $\gamma_i = 0$  より次のようになる.

$$\Delta \alpha_i = \sin^{-1} \left\{ \sqrt{\frac{(v_i + \Delta v_i)^2}{V^2 - (u_i + \Delta u_i)^2}} \right\} - \alpha_i \quad (14)$$

$\alpha_i = 30^\circ$  とし,  $\Delta u_i$ ,  $\Delta v_i$  を  $|V|$  の 1%, 5%, 10% としたときの誤差  $\Delta \alpha_i$  を示したのが Fig.8 である.  $u_i$ ,  $v_i$  は式 (11) で決まるため, グラフの横軸は  $\beta_i$  で示している. このように  $\beta_i$  が 0 に近いほど, すなわちロボットの移動方向が視覚センサの画像上で水平に近いほど, 誤差が大きくなるのが分かる. なお, この  $\Delta \alpha_i$  は, 式 (2), (3) に示した, 複数の VA によって生成されたロボットの誘導計画の統合において用いる.

表 1 は, 以上の手法を用いて Fig.9 の 2 通りのロボットの運動を観測して求めた視覚センサの俯角  $\alpha$  [deg] である. 上記のようにロボットの移動軌跡が水平に近いほど, 実際の値 (表の Actual) と比較して誤差が大きくなっている (表に \* で示した値) のがわかる. それ以外の場合でもある程度の誤差があるが, 式 (1) で示されるように VA が生成する誘導計画はある時点のロボットの移動方向と目標地点のずれ角で表現されており,  $\alpha_i$  の誤差が大きい場合でも誘導すべき方向 (左右) に影響はなく, ロボットは多少



Fig. 10 Robots navigated by the DVS

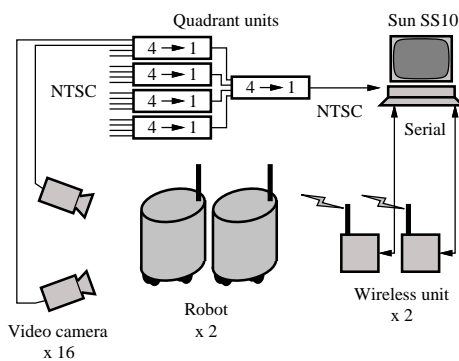


Fig. 11 Hardware system

蛇行するものの、そのような場合でも誘導は可能である。また、複数のVAが生成した誘導計画を統合し同時に用いることで、誤差の影響を少なくすることができる。

#### 4. 誘導実験

実際に屋外にDVSを構築しロボットの誘導を行うにはさまざまな障害があるため、ここではまずDVSのプロトタイプとして屋内の模型上にシステムを構築し、ロボットの誘導実験を行った (Fig.10)。この模型は Fig.3, Fig.10 に見られるように、複雑な背景や、日向や日陰、影といった照明条件の違いなど、複雑な屋外環境を忠実に再現したものである。この模型の大きさはおよそ4[m]×3[m]であり、ここに16台のカメラを設置してDVSを構築し、2台の小型ロボットの誘導を行った。システムのハードウェア構成を Fig.11 に示す。16台のカメラで撮影された画像は5台の4画面ユニット (Quadrant unit) によって1枚の画像にまとめられ、ホストコンピュータ (Sun SS10) に搭載した画像入力ボード (S2200) に入力される。画像入力ボードの解像度は640×480ピクセルであるので、VA1つあたりの解像度は160×120ピクセルとなる。ホストコンピュータでは16個のVAモジュールがそれぞれ対応する入力画像を参照して処理を行い、無線シリアル通信ユニットを介して2台のロボットと通信を行う。VAの画像処理速度は毎秒平均5フレーム程度である。2台のロボットの衝突回避は、3.2節に述べたように基本的にVAに

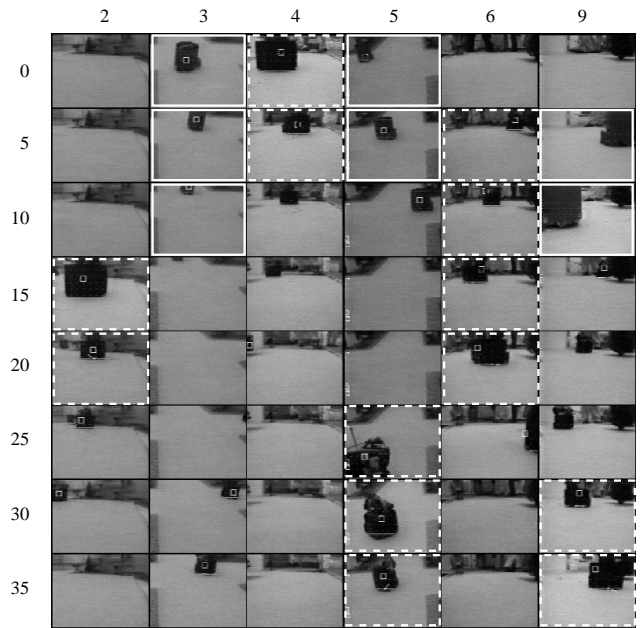


Fig. 12 Images processed by VAs

よって得られた情報に基づいて行うが、最終的にはロボットが持つ接触センサによって衝突を検知し、検知された場合には一定距離だけ後退して方向転換をするようにしている。

この実験ではまず、2台のロボットを区別するためにその色 (赤と黒) をVAに記憶させる。次に、複数の経路の教示を行うが、これら2台のロボットはその大きさや行動能力が似ているため、これらのうち1台のロボットを用いて複数の経路の教示を行う。最後に、2台のロボットを異なる経路に沿って誘導する。Fig.12は誘導段階におけるVAの画像処理の様子で、横軸はVAの番号、縦軸は時間 (単位は秒) を示している。実線の白枠は赤いロボット、破線の白枠は黒いロボットを誘導するために選択されているVAを示している。図のように、誘導タスクに応じて用いられるVAは次々に切り替わり、適切な視点からロボットを観測することによって誘導を行っている。Fig.10はDVSによるロボットの誘導の様子であり、DVSを国際会議IJCAI-97における学術展示として出展した際の模様である。16台のカメラは死角ができないように設置しているが、厳密な位置の測定は行っていない。3日間にわたる展示期間中、問題なくロボットを誘導し続けた。単純な画像処理、柔軟な誘導戦略、冗長な視覚情報といったDVSの持つ特徴が、このような複雑な環境におけるロボットの誘導を実現していると考えられる。

#### 5. おわりに

本論文では分散視覚システム (DVS) を提案した。DVSは環境に埋め込まれた多数の視覚エージェント (VA) によって構成されるシステムである。分散視覚という、これまでの多くの研究ではあまり取り上げられてこなかったアプローチをとることで注意制御の問題を解決し、これまで困難であった、複雑な屋外環境における視覚情報に基づいたロボットの誘導を可能にした。また、複雑な屋外環境を忠実に再現した街の模型を用いて

DVS を構築し、ロボットを誘導する実験を行うことで、実際に DVS が非常にロバストにロボットを誘導できることを確認した。

今後の研究課題として、主にシステムの大規模化に伴う問題がある。実際のシステムでは、多くのロボットやその他の物体を同定しなければならない。この問題は、本稿で提案した手法に加え、ロボットに与える制御命令と運動の様子の関係などから解決できると思われる。また、観測によって VA の定性的な位置関係を求める手法 [9] を導入し、得られた位置関係を用いれば、ロボットの誤認識の防止などで有利になるものと思われる。さらに、システムが大規模になった場合の誘導経路の計画や、多数のロボットを限られた数の VA で同時に誘導する際の VA の組織化の問題、人間などロボット以外の移動体に関する処理なども、今後の研究課題として挙げられる。

今日、コンピュータ、ネットワークやマルチメディアなどの技術の急速な発展によって、街には監視カメラをはじめとする多くのセンサが設置され、コンピュータネットワークが張り巡らされている。本論文で述べた DVS は、このように様々な利用目的に応じて常に成長を続けていくインフラ、すなわちネットワークやセンサなどを組み合わせてできる、新しいロボットのためのインフラである。将来的には、これらを統合した新しい情報システムとして、ロボットに限らず人間を含む様々な実世界エージェントに情報を提供する知覚情報基盤 [2] (PII: Perceptual Information Infrastructure) へと DVS が発展していくことを期待している。

## 参 考 文 献

- [1] D. H. Ballard: "Reference frames for animate vision," Proc. IJCAI, pp. 1635-1641, 1989.
- [2] H. Ishiguro: "Distributed vision system: A perceptual information infrastructure for robot navigation," Proc. IJCAI, pp. 36-41, 1997.
- [3] H. Mizoguchi, T. Sato and T. Ishikawa: "Robotic office room to support office work by human behavior understanding function with networked machines," Proc. ICRA, pp. 2968-2975, 1996.
- [4] S. Moezzi: "An emerging Medium: Interactive three-dimensional digital video," Proc. Int. Conf. Multimedia, pp. 358-361, 1996.
- [5] C. S. Pinhanez and A. F. Bobick: "Approximate world models: Incorporating qualitative and linguistic information into vision systems," Proc. AAAI, pp. 1116-1123, 1996.
- [6] V. R. Lesser and D. D. Corkill: "The distributed vehicle monitoring testbed: A tool for investigating distributed problem solving networks," AI Magazine, pp. 15-33, 1983.
- [7] E. H. Durfee and V. R. Lesser: "Partial global planning: A coordination framework for distributed hypothesis formation," IEEE Trans. SMC, Vol. 21, No. 5, pp. 1167-1183, 1991.
- [8] K. Hosoda and M. Asada: "Versatile Visual Servoing without Knowledge of True Jacobian," Proc. IROS, pp. 186-193, 1994.
- [9] 十河, 石黒, 石田: "空間的制約の伝搬に基づく定性的空間構造の同定," 電子情報通信学会論文誌, Vol. J81-D-II, No. 10, pp. 2311-2320, 1998.

1990 年豊橋技術科学大学工学部情報工学課程卒業。1995 年筑波大学大学院博士課程電子情報工学専攻修了。同年米国カーネギーメロン大学ポストドクトラルフェロー、ロボット研究所にて Automated Highway System の研究に従事。1997 年より慶應義塾大学理工学部の有期助手として京都大学情報学研究所にて移動

ロボットの研究に従事。移動ロボット、センサー、ロボットのプログラミング環境に興味を持つ。博士(工学)。(日本ロボット学会正会員)

### 石黒 浩

1986 年山梨大学工学部計算機科学科卒業、1988 年同大学院修士課程修了、1991 年大阪大学大学院基礎工学研究科物理系専攻修了。同年山梨大学工学部情報工学科助手、1992 年大阪大学基礎工学部システム工学科助手、1994 年京都大学大学院工学研究科情報工学専攻助教授、1998 年より 1 年間、カリフォルニア大学サンディエゴ校客員研究員。視覚移動ロボット、能動視覚、パノラマ視覚、分散視覚に興味を持つ。工学博士。人工知能学会、情報処理学会、電子情報通信学会、IEEE、AAAI 各会員。(日本ロボット学会正会員)

### 石田 亨

1976 年京都大学工学部情報工学科卒業、1978 年同大学院修士課程修了。同年日本電信電話公社電気通信研究所入所、横須賀研究所においてソフトウェア工学、知識処理などの研究開発に従事。現在、京都大学大学院情報学研究所社会情報学専攻教授。問題解決、分散人工知能、コミュニケーション、社会情報システムに興味を持つ。工学博士。人工知能学会、情報処理学会、電子情報通信学会、日本ソフトウェア科学会、IEEE、ACM、AAAI 各会員。

### 十河 卓司

1997 年京都大学工学部情報工学科卒業、1998 年同大学院修士課程修了。現在、同大学院情報学研究所博士後期課程在学中。視覚情報処理に関する研究に従事。人工知能学会、情報処理学会 各会員。(日本ロボット学会学生会員)

### 木元 克美