

分散視覚を用いた人間ロボット協調ナビゲーション

岡田 慧* 石黒 浩* 石田 亨*

Human-robot Cooperated Navigation Using a Distributed Vision System

Kei Okada*, Hiroshi Ishiguro* and Toru Ishida*

This paper describes a human-robot cooperation system which enables robust behaviors of robots in a real world by combining human's ability of recognition, inference and decision with robot's autonomy. In previous systems, interactions between operators and robots are designed by programmers. We consider that the interactions should dynamically change through iterative process of human-robot cooperation. In the system, the robots obtain autonomous behaviors by memorizing human's instructions and obtain autonomous behaviors for behaving in complex environments. We have designed a system which consists of vision guided mobile robot, several cameras embedded in the environment and visual information interface with an idea of layered interaction structure, and we have experimented in a town model. We call this system DVS_{coop} (Distributed Vision System for human-robot cooperation).

Key Words: Robot-Human Interaction, Distributed Vision, Mobile Robot Navigation

1. はじめに

これまでに数多くの実世界で活動するロボットの研究が続けられてきたが、現在の技術水準ではロボットの行動範囲が特定の環境内に限定されたり、ロボットを動かすたびに人間が様々なパラメータチューニングを施さなければならなかった。ゆえに現段階で実世界で活動するロボットを実現するには、人間が積極的に介在するロボットシステムが必要である [1]。

従来の自律移動ロボットやテレオペレーションの研究では、ロボット自身がすべてを判断するもの [2]、あるいは人間がすべてを判断するもの [3]、または人間が大局的な判断をしロボットは精密な動作を計画実行するもの [4] などシステムの形態は様々ではあるが、人間とロボットの役割分担は固定的に設計されていた。これに対して我々は、人間による指示とロボットの行動の繰り返しの中で人間の指示がロボット側に蓄積され、ロボットはこの蓄積された情報を用いることで徐々に自律的な活動を行うことが可能になるというように、人間がロボットを使い続けることで人間とロボットの役割が動的に変化するシステムが必要であると考えた。

本論文では、上記の観点から、ロボット-人間インタラクション (RHI: Robot-Human Interaction) を実現する人間ロボット協調システムを提案し、分散視覚システム (DVS : Distributed Vision System) [5] 上に構築した。このシステム

を (DVS_{coop} : Distributed Vision System for human-robot cooperation) と呼ぶ。DVS とは環境に埋め込まれた視覚センサと計算資源をもつ多数の視覚エージェント (VA: Vision Agent) からなるシステムであり、実世界におけるロボットや人間の活動を支援するためのインフラストラクチャの一例として提案されている。すなわち、ロボットは基本的に自律的に行動できる機能を有するものとし、DVS はその機能を補助するものであり、また、必要とされるロボットのタスクに対し VA が足りない場合は、そのタスクに応じて VA が追加されるものとして議論を進める。

DVS_{coop} では、ロボットは人間の指示内容に基づいて行動するだけでなく人間の指示内容を記憶し、それを再現するために内部表現を構成し、この内部表現を利用して自律的に行動する機能を実現する。またロボットがその内部表現に基づいた判断を下すことが不可能な状況では、ロボット側から人間へ判断を要求し、人間はその状況に応じた的確な行動を指示する。ロボットはこの指示内容を自らの内部状態に反映し、より正確な判断が可能な内部表現を獲得する。このように DVS_{coop} では人間がシステムとインタラクションを繰り返すたびに人間の判断内容がロボット側に内部状態として蓄積され、人間がシステムを使えば使うほどロボットは自律的な振る舞いを獲得し、人間の負担が軽減される。

本手法の要点は次の3点である。

- (1) DVS を利用することで、ロボットの行動に適した視点を選択でき、この視覚を用いたロボットへ指示が可能である。
- (2) 人間とシステムはテンプレート画像を用いた情報交換によ

原稿受付 1997年10月20日

*京都大学工学部情報工学科

*Department of Information Science, Kyoto University

リタスクの通信を行う。テンプレート画像を用いることで人間は直感的にシステムの内部表現を理解することが可能であり、また、システムはロバストな処理が可能になる。

(3)人間とシステムのインタラクションは階層構造を持つ。人間とシステムとのインタラクションを繰り返すことで、ロボットはより自律的な振る舞いを獲得するが、人間とシステムはこのロボットの自律性に応じた階層を用いてインタラクションを行う。

2. 人間ロボット協調システム

本章では DVS_{coop} の特徴であるロボットの行動に適した視点を提供する分散視覚システム (DVS)、人間とロボットが共有可能かつ、十分な意図の伝達が可能なテンプレート画像を用いたタスクの通信、人間とロボットのインタラクションの繰り返しによるロボットの自律性の獲得を可能にするコミュニケーションの階層構造について議論する。

2.1 分散視覚システム (DVS)

現在、インターネットに代表される計算機ネットワークは情報通信のインフラストラクチャとして整備されつつあるが、将来は画像処理、音声理解など実世界と計算機内部の世界を結び様々なインターフェースを取り込むことで、ロボットや人間の知覚情報処理機能を支援するインフラストラクチャに発展すると考えられる。分散視覚システム (DVS) [5] は、ロボットや人間の知覚情報処理機能を支援するインフラストラクチャの例として提案されており、Fig. 1 に示されるように環境の様々な場所に視覚センサと計算資源を備えた VA が配置され、おのこの VA が独立に局所環境を観測し、その結果を環境内の人間やロボットに伝えることで、人間やロボットの知覚処理機能を支援する。

ロボットが実世界を認識する際に困難で重要な問題の一つに、タスクの遂行に必要な情報を獲得するための注意制御が考えられる。従来のロボットではどの空間をどの順番で観測するのかという困難な空間的、時間的な注意制御問題を解く必要があった。それに対し DVS が整備された環境では、ロボットは環境の様々な場所に配置された VA を利用することで注意制御問題を簡単にし、様々な視覚情報を利用することで、移動ロボットの誘導 [5] や実時間でのロバストな複数人の行動の追跡 [6] が可能になる。

DVS の特徴は以下の 3 点である。

1. 動的な環境をモデル化しやすい

VA は環境に対して常に静止しており物体の環境に対する運動を直接知覚することができる。例えば従来の一般的な環境認識では、まずできるだけ正確にロボットの移動軌跡を復元し、それを基準として環境の詳細なモデルを構築する。これに対し DVS における環境認識では、あらかじめ環境内の随所に設置された VA をモデルの構築における基準として用いるためにセンサの位置情報を正確に復元するという、時として困難な作業は必要ない。

2. 様々な局所環境を同時に観測できる

ロボットにとり、そのタスクに基づいて適切な注視を行い効果的に計算資源を配分することは重要な機能の一つと考えられ

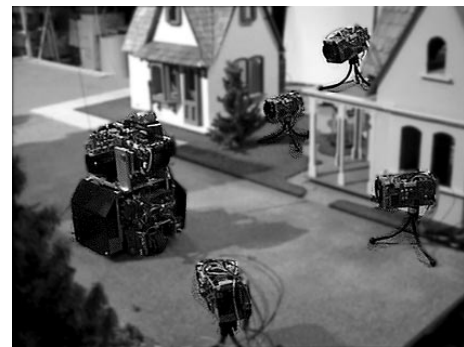


Fig. 1 A mobile robot and vision agents in the DVS

ているが、単一ボディ単一センサの従来のロボットでは注視領域選択の順序を決定するために複雑な行動計画問題を解かなければならない。DVS では環境内の複数の VA により様々な局所環境を同時に観測できるために、注視領域選択の順序の決定に関する制約は存在せず、各 VA は独自に並列に注視すればよい。

3. 行動のための適切な視点を提供できる

DVS では環境に配置された複数の VA がロボットの行動のために適切な視点を提供することができる。例えば、ロボットが障害物を回避するためには、ロボットと障害物との距離を計測しロボットが障害物に衝突する時刻を計算する必要があるが、その際ロボットのボディに設置されたカメラの視点を利用する方法よりも、ロボットと障害物とを側面から同時に観測できる場所に設置された VA の視点を利用する方法の方が、ロボットの障害物との距離を簡単に計算することができる。

人間とロボットの協調動作によりロボットの誘導を行うシステムとしては文献 [3] [7] が存在する。これらのシステムでは人間はロボットのボディに設置されたカメラ画像を利用してロボットの行動を指示するが、 DVS_{coop} は DVS 上に構築されるために人間はロボットのボディに設置されたカメラ画像だけではなく、環境の様々な場所に配置された VA の視点からロボットの行動に適した視点を利用してロボットの行動を指示することもできる。したがって、人間はロボットの位置や向き等に制限されることなくロボットの状態を把握し、適切かつ簡潔なロボットの誘導が可能である。また、 DVS_{coop} ではロボットを自律制御する際に時として困難な問題となるロボットの注視点の選択を、すべてロボット側で処理するのではなく、人間が適切な視点をロボットに指示することでロボット側の負担を軽減している。

2.2 テンプレート画像によるタスクの通信

人間とロボットが通信する共通言語としては従来から自然言語 [8] やロボット言語 [9] などを用いた手法が研究されてきた。しかしこのような方法では人間からロボットへの通信は言語情報を用いるにもかかわらず、人間は観測に基づいてロボットの状態を把握しなければならない。ここではロボットを制御するための情報媒体とロボットの状態を把握するための情報媒体が異なるために、人間はこれらの媒体間で情報を変換する必要があり、時として人間に大きな負担をもたらす。

そこで DVS_{coop} では、人間とロボットの双方が共有可能な

画像情報を用いた通信を行う。人間はこの共有する画像情報を利用してロボットを制御し、同じ画像情報を利用してロボットの状態を把握する。ここではロボットを制御するための情報媒体とロボットの状態を把握するための情報媒体が同一であるために、人間が情報媒体間での情報変換を行う必要がなく、人間の負担を取り除くことができる。

$DV S_{coop}$ では、オペレータは視点の選択と画像の認識とそれに対応するロボットの行動を 1 組にした記述を用いてロボットに動作を指示する。例えばロボットの移動では「ある VA により（視点の選択）ある位置でロボットが発見されたとき（画像の認識）、ロボットをその VA の視線方向に沿って動かす（ロボットの行動）」となる。この視点の選択と画像認識とロボットの行動の対の記述を以下「Action」と呼ぶ。

画像の認識にはモデルベース手法ではなく、外界情報を直接モデルとし画像の局所相関演算を用いて同一輝度領域を探索するテンプレートマッチング法を用いる。モデルベース手法における認識は一般的にセンサ情報から画像特徴を抽出、セグメント化、モデルマッチングと段階的に処理されるが、認識に失敗した際には、処理のどの段階で失敗したのかを人間が把握することは難しい。一方テンプレートマッチング法では、人間は対象物とその背景画像を観測することで認識失敗の原因を容易に把握できる。

視点の選択とテンプレートマッチング法による画像の認識とロボットの行動の対として記述される「Action」はまた、システムにとり直接的な表現である。システムは「Action」の内容を解釈することなく実行し、ロボットを制御することができる。人間はこの単純な機構の故に、システムの状態を簡単に、かつ直観的に把握することができる。

2.3 インタラクションの階層構造

ロボット-人間インタラクション (RHI) では人間からロボットへの指示と、ロボットが人間に対して行う判断の要求を繰り返すことで、人間の指示や判断がロボット側に蓄積され、ロボットが自律性を獲得する機構が重要であると考えられる。そのために $DV S_{coop}$ では人間とシステム間のインタラクションは階層構造を持ち、この階層構造に対応したロボットの制御を行う。つまり、下位の階層ではロボットのハードウェアに近いレベルでの動作の制御を行い、上位の階層では下位層の動作を利用し、より自律性の高い動作の制御を行う。また、各階層における動作はあらかじめプログラミングされているのではなく、人間とシステムがインタラクションを繰り返すことで動作をボトムアップに構築する。

$DV S_{coop}$ におけるインタラクションの階層構造を Fig. 2 に示す。システムは「Action」、「Subtask」、「Task」の階層に対応した動作を記憶管理する。システムの最下層には視覚センサの処理結果に基づき車輪の回転数やステアリング角度等を制御し、「Action」を実行するための視覚フィードバックモジュールが存在する。ここでの例は 3 層であるが階層数は応用により変更される。

システムが稼働した当初は各階層には何も記憶されていない。人間はシステムに視点の選択と画像の認識とロボットの行動を組した「Action」を指示することでロボットを制御し、システ

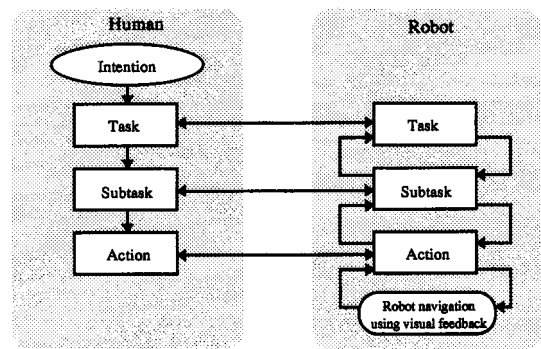


Fig. 2 Layered human-robot interaction structure

ムは人間から指示された「Action」を蓄積し、Action 層に記憶させる。人間はいくつかの「Action」を記憶させた後、これらの「Action」を組み合わせることで「Subtask」を教示することができる。「Subtask」はいくつかの「Action」の連結を自動的に実行することで実現される。

人間がシステムを使えば使うほど、システムはより上位の階層を用いた動作を記憶することができ、人間はこれらのより自律性の高いロボットの動作を用いて、ロボットへ指示を与えることができる。このようにシステムが「Action」の系列を自動的に連続に実行し、人間は個々の「Action」ではなく「Subtask」として指示すればよい状態を、 $DV S_{coop}$ においてはロボットが自律的に動作していると考えられる。

また、システムは適宜人間に適切な判断を求める。例えば視覚フィードバックモジュールにおいてロボットに移動コマンドを送っているにもかかわらずロボットの動きが観測されない場合は、視覚フィードバックモジュールが注視している領域がロボットを示していなかったり、あるいはロボットが移動不能な状況にある可能性がある。このような際にはシステムは人間に注意を促し適切な処理を要求する。またシステムは人間が指示した「Action」を記憶しており、頻繁に実行される「Action」の系列は「Subtask」候補として人間に提案する。人間はタスク、環境を考慮し、適切な系列を「Subtask」としてシステムに記憶させる。

3. アーキテクチャ

$DV S_{coop}$ のアーキテクチャを Fig. 3 に示す。 $DV S_{coop}$ はヒューマンインターフェース (Human Interface)、視覚エージェント (Vision Agent)、ロボット (Robot) から構成され、おのおのがコミュニケーションリンク (Communication Link) で結合されている。

視覚エージェント (VA: Vision Agent)

VA は視覚センサ (Vision Sensor) と計算資源から構成される。画像処理機能として、テンプレートマッチング法による物体認識/追跡が可能な視覚追跡部 (Visual Tracker)、差分演算による移動物体の検出が可能な移動物体検知部 (Moving Object Detector) をもち、おのおのの結果はロボットへ転送される。転送部 (Transmitter) ではヒューマンインターフェースからの命令に基づき視覚センサの画像を転送する。また、テンプレ

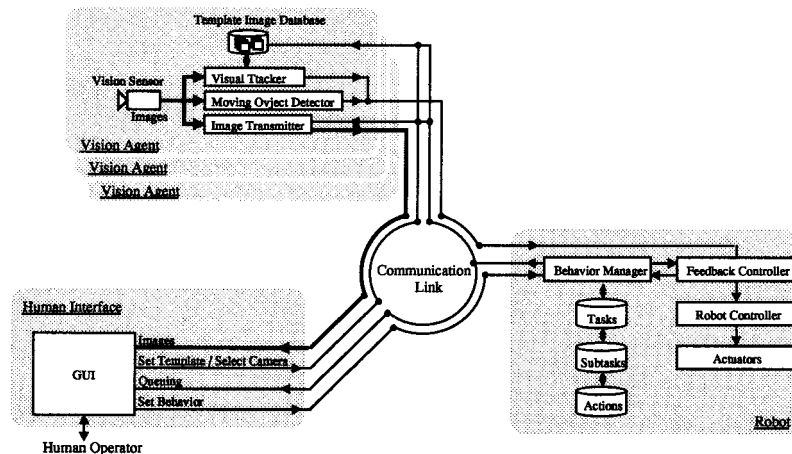


Fig. 3 The architecture of the *DVScoop*

レート画像データベース (Template Image Database) では視覚追跡部で利用するテンプレートが保存されており、ヒューマンインターフェース、ロボットからの命令に基づき新しいテンプレートを設定できる。

ヒューマンインターフェース (Human Interface)

ヒューマンインターフェースは GUI として設計されオペレータはマウス操作でロボットの動作を指示することができる。ヒューマンインターフェースから VA へはテンプレートの設定やカメラ画像の転送要求を行える。カメラ画像の転送要求に従い VA から転送された画像はヒューマンインターフェース上に表示される。また、ヒューマンインターフェースからロボットへはロボットの動作を指示し、逆にロボットからヒューマンインターフェースは動作に関するオペレータの判断を要求する。

ロボット (Robot)

動作管理部 (Behavior Manager) は第 2.3 節で述べたインタラクションの階層構造に対応した動作の記憶管理部である。ヒューマンインターフェースからの動作の指示に基づき「Action」、「Subtask」、「Task」情報を記憶管理し、ロボットの動作を決定する。このロボットの動作に基づき VA へ必要なテンプレートの指定やカメラ画像の選択命令を転送する。フィードバックコントローラ (Feedback Controller) では動作管理部から出力されるロボットの動作命令と VA から得られる画像処理結果情報からロボットコマンドを生成し、ロボット制御部 (Robot Controller) はこのコマンドを用いてアクチュエータ (Actuator) を作動させる。

コミュニケーションリンク (Communication Link)

現時点での現実的なアプローチを前提とした *DVScoop* ではコミュニケーションリンクとしてイーサネットの使用を想定している。ネットワーク上で転送されるデータには、テンプレート追跡結果の座標値、カメラ選択情報などの、データ容量の少ないシンボリックなデータと視覚センサで得られたデータ量の多い画像データがある。視覚センサの解像度を 128×128 画素、8 [bit] としイーサネットの実転送速度を 3 [Mbps] とすると、1秒で 24 枚の画像を伝送することができる。ここで、実際に開発され実験に用いられている Nomad [10] を参考に、人間社会

で活動できるロボットの大きさとして、ボディ幅 80 [cm]、ロボットの速度 10 [km/h] とすると、ロボットが自身のボディ幅の距離を 0.288 [sec] の時間で進むことができる。

このとき、オペレータがロボットの 1 秒間に 15 枚の画像で状態を把握することができる。システムは同時に最大 4 台の VA で得られるカメラ画像を伝送することができる。

システムが同時に 4 台以上の VA のカメラ画像の伝送を要求した場合、システムとオペレータは対話により伝送するカメラ画像を調整する。この手順を以下に示す。同時に伝送できる最大の画像数を N とすると、

- (1) システムが N 台以上のカメラ画像の伝送を要求した場合、システムは重要でないカメラ画像がないかオペレータに問い合わせる。例えば複数のロボットが協調作業を行い、各ロボットの作業範囲が近接している場合、一つの VA で複数のロボットを追跡できる。オペレータが重要でないと判断した VA はシステムの要求リストから外す。
- (2) 1 の手続き後も、システムが N 台以上のカメラ画像の伝送を要求する場合、システムは、追跡しているロボットが一つであるか、複数であるかをオペレータに問い合わせる。
- (3) システムが一つのロボットを追跡している場合、ロボットの速度を低下させることで、同時に伝送できるカメラ画像の数を増加させることができる。
- (4) システムが複数のロボットを追跡している場合、オペレータはどのロボットの速度を低下あるいは停止させるかを判断し、伝送するカメラ画像を調節する。

4. システムの試作と実験

提案したアーキテクチャに基づき試作システムを構築した。

4.1 ハードウェア構成

試作システムのハードウェア構成を Fig. 4 に示す。試作システムでは 1/12 の街モデル上に設置された 12 台の CCD カメラで得られる 12 枚の画像をマルチ画面ユニットに入力し、1 枚の画像に変換する。この画像をワークステーションに接続された富士通社製トラッキングビジョンに入力する。トラッキングビジョンはテンプレートマッチングによる物体追跡をハ-

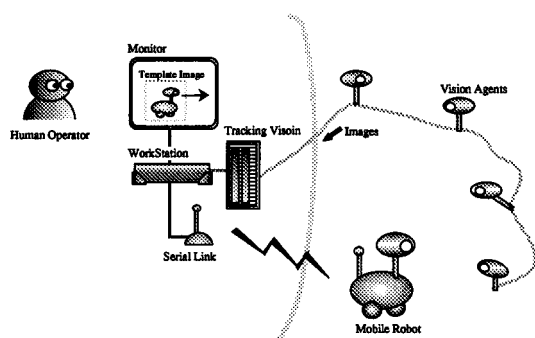


Fig. 4 The hardware configuration for experimental system

ドウェアによりビデオレートで実現できる。ロボットはマイクロマウスを改造したものを利用している。ロボットとワークステーションは RC-232C を介して無線で結合され、オペレータはワークステーションから通信を介してロボットを操作することができる。

4.2 オペレータによる行動の指示

オペレータは視点の選択、画像の認識、ロボットの行動の組で表現される「Action」を利用し、システムにロボットの行動を指示する。「Action」はスタートテンプレート、メソッド、ストップテンプレートの三つのパラメータで表現される。スタートテンプレートはカメラ画像からのロボットの見え方、あるいは注視領域を示し、ストップテンプレートはロボットの行動の終了の際におけるカメラ画像からのロボットの見え方、あるいは注視領域を示す。システムはストップテンプレートで示される画像を認識した際に、ロボットの行動を停止し「Action」を終了する。

メソッドは、視点の選択と行動戦略を示す。視点の選択では環境中に設置されたカメラの画像を用いる方法と、ロボットのボディに設置されたカメラの画像を用いる方法がある。環境中に設置されたカメラ画像を用いる方法では、ロボットの見え方をテンプレート画像として指定し、このロボットの動きをカメラの視線軸方向に沿ってカメラに近づく方向に前進、あるいはカメラを背面にして遠ざかる方向に前進するようにロボットの行動を制御する。また、ロボットのボディに設置されたカメラ画像を用いる方法では、カメラ画像中のある領域を注視領域とし、この注視領域を常にカメラの中心で捉えるようにフィードバックをかけながら注視領域に近づく方向に前進、または注視領域から遠ざかる方向に後進するように、ロボットの行動を制御する。メソッドでの視点の選択と行動戦略に基づき、カメラ画像上でのロボットや注視領域の動きと実際のロボット動きの対応に合わせてスタートテンプレートによる追跡結果をロボットのステアリング角にフィードバックする際のゲインの正負を決定することができる。

4.3 画像処理

視覚フィードバックには局所相関演算によるテンプレートマッチング法による画像の追跡、認識を用いる。

ロボットの移動に伴い、カメラとロボットの位置関係や距離は変化するために、カメラ画像上でのロボットや注視領域の見え方は刻一刻変化する。このようなテンプレート画像の変化に

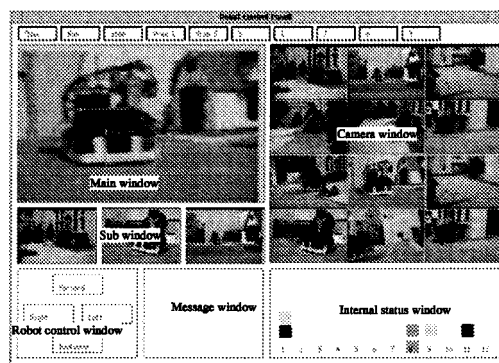


Fig. 5 Robot control panel

対応するために、記憶されたテンプレート画像に基づき大きさの違う三つのテンプレート画像を作り(0.9倍, 1倍, 1.1倍)これらを用いて局所相関演算を行うことで、ロバストな画像の追跡、認識を行っている。また、局所相関演算による相関値がある閾値を越え、記憶されているテンプレート画像とそれに対応するロボットや注視領域の見え方が一致しなくなった時点で、システムは現在の注視領域を新しいテンプレート画像としてシステムに取り込み、テンプレート画像の自動更新を行う。

4.4 システムの動作例

オペレータは Fig. 5 に示されるワークステーションのモニター上に表示されたロボット制御パネルと、入力デバイス(マウス)を用いてシステムを操作する。

ロボット制御パネル上には 12 台のカメラ画像を表示するカメラウィンドウがあり、これらのウィンドウからロボットの制御に利用したい視点を持つカメラウィンドウをクリックすることで、そのカメラ画像がメインウィンドウに表示される。オペレータはこのメインウィンドウ上でテンプレートやメソッドの設定を行う。またオペレータとシステムの対話はロボット制御パネル上に適宜表示されるメニューを介して行われる。過去にメインウィンドウで表示されたカメラ画像はサブウィンドウにスタックされる。ロボットコントロールウィンドウは緊急時などの利用を想定し、ロボットにロボットコマンドを直接送るためのボタンがついている。メッセージウィンドウはロボットからの言語情報によるメッセージを表示する。内部状態ウィンドウでは各 VA が保持するテンプレートの数と、そのテンプレートの現在の相関値がグラフィカルに表示される。

オペレータは以下の手順でシステムを操作する。

「Action」の設定

オペレータはカメラウィンドウをクリックすることでロボットへの指示に適したカメラを選択し、メインウィンドウ上でスタートテンプレートを設定する。次にメニューを介してメソッドを指定する。Fig. 6 にメソッド選択のためにメニューが表示されているところを示す。

「Action」の実行

システムはオペレータが指示したスタートテンプレートの追跡結果を利用し、視覚フィードバックを用いてロボットを制御する。オペレータはメインウィンドウをクリックすることで

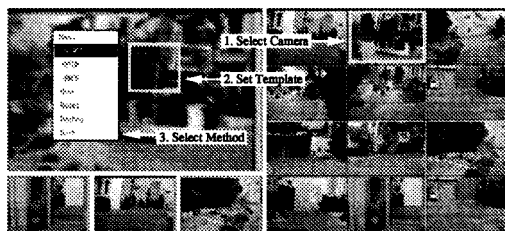


Fig. 6 Set "Action"

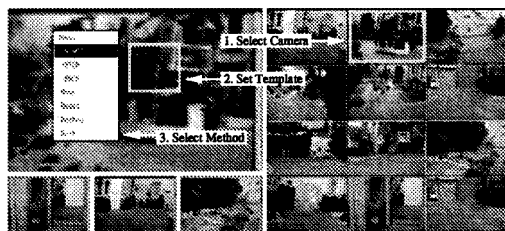


Fig. 7 Detect "Action"

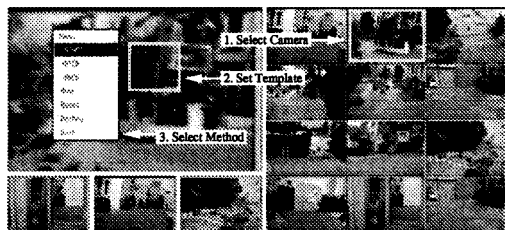


Fig. 8 Propose "Subtask"

ロボットの動きを観察し適宜ロボットを止め、ストップテンプレートを指定する。

「Action」の発見

システムは記憶された「Action」のスタートテンプレート画像をカメラ画像上で常に探索し、発見した際には、その「Action」を発見するという。「Action」が発見されると、システムはメニューを介してオペレータに「Action」の実行を提案する。オペレータは提案された「Action」を実行するか否かを判断し指示する。Fig. 7 に発見した「Action」の実行の判断をオペレータにメニューを介して提案しているところを示す。

「Subtask」の提案

システムは頻繁に実行される「Action」の系列を「Subtask」の候補としてオペレータに提案する。オペレータは提案された「Subtask」候補からシステムが記憶すべき「Subtask」を指示する。Fig. 8 に頻繁に実行される「Action」の系列「Action1」、「Action2」、「Action3」を「Subtask」候補としてオペレータにメニューを介して提案しているところを示す。

「Subtask」の発見

システムは、発見された「Action」を利用する「Subtask」が存在するか調べ、存在すればその「Subtask」の実行を提案する。

「Subtask」の実行

「Subtask」の実行は、それを構成する「Action」を逐次実行

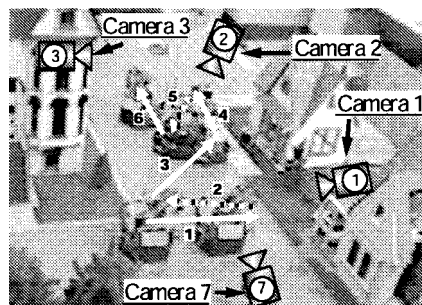


Fig. 9 The results of Exp. (1)

Table 1 The records of Human-Robot communication for Exp. (1)

t	Selected Cameras	Method
1	Camera 1	Go toward a camera
2	Camera on the robot body	Go back away from an attention area
3	Camera on the robot body	Go toward an attention area
4	Camera 7	Go forward with robot's back against a camera
5	Camera 2	Go back away from an attention area
6	Camera 3	Go forward to a camera

することで達成される「Task」に関しても同様である。

4.5 実験結果

4.5.1 実験(1)

DVS_{coop}では人間の簡潔な指示方法によりロボットの複雑な行動を指示できることを実現した。これを示すために実験(1)を行った。

オペレータは利用するカメラの指定、テンプレート画像の指定、ロボットの移動方向の指定によりロボットを制御する。このときカメラの指定とロボットの移動方向として以下の四つの方法がある。

- 環境中のカメラを使ってロボットがカメラに近づく方向に前進する方法。
- 環境中のカメラを使ってロボットがカメラから遠ざかる方向に前進する方法。
- ロボットのボディ上のカメラを使って注視領域に近づく方向に前進する方法。
- ロボットのボディ上のカメラを使って注視領域に遠ざかる方向に後進する方法。

これらの指示方法を組み合わせてロボットの誘導実験を行った結果を Fig. 9 に示す。図中で矢印はロボットの軌跡を示す。実線で示されるロボットの軌跡は前進を示す。点線で示されるロボットの軌跡は後進を示す。ロボットの軌跡を示す矢印の上部の数字は、以下で述べる時間 t を示す。また、Table 1 は、上から時間順にロボットの誘導に使用したカメラ、オペレータの指示方法を示す。左列の時間 t は、システムが「Action」を実行するたびに1ずつ増加される。

例えば時間1では、環境中のカメラ(Camera1)を用いてロボットがカメラに近づく方向(この場合、Fig. 9 の右端の家屋の方向)に前進する「Action」を設定する。この設定は第 4.2

節で説明したロボット制御パネル上のメニューを用いて行う。ロボットが移動し、家屋の前までくるとオペレータはロボットに停止を指示し、「Action」をシステムに登録する。この実験では「Action」の設定に約2秒、ロボットの移動に約2秒の約4秒かかった。また、時間2ではロボットのボディ上のカメラを用いて、注視領域（この場合、Fig. 9の右端の家屋の一部）から遠ざかる方向に後進する。

このように DV_{Scoop} ではロボットのボディ上のカメラだけでなく、環境中のカメラも含めて適切な視点を選択し、ロボットの行動を指示することができる。例えばロボットの後進を考えると、ロボットのボディ上のカメラだけしか利用できない場合、カメラのパン軸が180度回転し背面を視野に入れロボットが停止する位置を設定する必要がある。 DV_{Scoop} では、ロボットの側面に位置するカメラなど、ロボットと環境の関係を直接観測できる視点を持つカメラを利用しロボットを観測し、ロボットの停止位置を設定することができる。

4.5.2 実験(2)

DV_{Scoop} ではロボットの誘導を続けることで、人間がロボットに教示した情報がシステム内に蓄積され、ロボットはこの蓄積された情報を用いて自律的に行動する。また、ロボットは適宜人間に正確な判断を要求する。これを示すために実験(2)を行った。

Fig. 10は、ロボットの軌跡を固定時間間隔で撮影し重ね合わせて表示してある。ロボットの上部の数字は撮影された時点を示し、白抜き数字で表される時点では、人間とシステムのインタラクションが行われている。

Table 2は、上段から時間順にこのとき、システムがロボットの誘導に使用したカメラ、システムからの問い合わせ、オペレータの指示内容、ロボットの自律的行動を示す。左列の数字 t は「Action」を実行するたびに1ずつ増加される。

Table 2の上部の4行、中部の4行、下部の4行がそれぞれがFig. 10の上図、中図、下図に対応する。

実験(2)では、Camera 1に近づくように前進 (Action1)、Camera 2に近づくように前進 (Action2)、Camera 3に近づくように前進 (Action3)、Camera 4に近づくように前進 (Action4)の四つの「Action」を組み合わせてロボットの誘導を実現している。

Fig. 10では、オペレータは「Action」の設定を行い、その実行を指示する。「Action」の設定は、1)適切なVAの選択、2)テンプレート画像を指定、3)ロボットのメソッドの指定、の三つの手順を踏む必要がある。本実験では四つの「Action」で構成されているため、オペレータは「Action」の設定と実行を4回行う必要があるが、図からはオペレータが「Action」の設定を行っている時点(白抜き数字の数)は6時点であることが分かる。これはオペレータが「Action」を設定する際に手間取っているために、いくつかの「Action」の設定で2時点以上の時間を掛けているためである。

中図ではシステムは自律的に「Action」を発見し、その「Action」を構成するテンプレート画像やメソッドの情報をオペレータに提示する。オペレータはそれらの情報を再度設定する必要がなく、システムから提案された「Action」を実行す

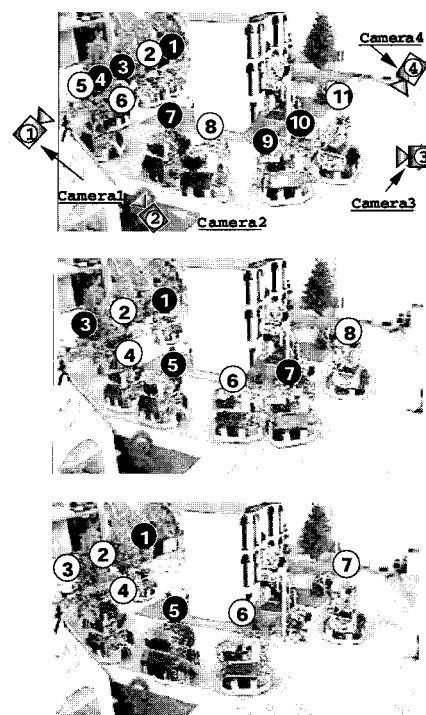


Fig. 10 The results of an Exp. (2)

Table 2 The Records for Exp. (2)

t	Selected Cameras	Proposals from the DV_{Scoop}	Instructions from the operator	Autonomous behaviors of the robot
1	Camera 1		Set Action1	
2	Camera 2		Set Action2	
3	Camera 3		Set Action3	
4	Camera 4		Set Action4	
5	Camera 1	Propose starting Action1	Start Action1	Action1
6	Camera 2	Propose starting Action2	Start Action2	Action2
7	Camera 3	Propose starting Action3	Start Action3	Action3
8	Camera 4	Propose starting Action4	Start Action4	Action4
9	Camera 1	Propose starting Subtask1	Start Subtask1	Subtask1 (Action1, Action2)
10	Camera 2			
11	Camera 3	Propose starting Subtask2	Start Subtask2	Subtask2 (Action3, Action4)
12	Camera 4			

るか否かをメニューを介して指示するだけでよい。図では4時点(図中白抜き数字の数)のメニュー操作で「Action」を実行することができており、オペレータがスムーズにロボットを誘導することができていることが分かる。

この後、実験ではシステムからオペレータへ「Subtask」の提案があった。オペレータは「Action1」から「Action2」の系列を「Subtask1」として記憶させ、また同様に「Action3」から「Action4」の系列を「Subtask2」として記憶させた。

下図では「Subtask」を用いてロボットを誘導している。システムは「Subtask」を発見しオペレータにその実行を提案し、オペレータはメニューを介してその実行を指示している。システムは記憶された「Subtask」を構成する「Action」系列を自動的に実行することでロボットは半ば自律的に行動することができる。図では2時点(図中白抜き数字の数)のメニュー操作でロボットの誘導が完了しており、「Subtask」を利用することでオペレータの負担が軽減しロボットの半自律的な誘導を実現している。このとき、ロボット誘導が完了するまでにかかる時間はすべての「Action」の設定を行った場合の約64%(=7/11)である。

これら実験をまとめると以下の点が指摘できる。人間はテンプレート画像を用いてロボットの行動に適した視点と、行動に必要な注視領域を指示することでロボットにタスクを教示する。テンプレート画像を用いたタスクの通信では、人間が直観的にタスクを指示できるために、人間の負担がない簡潔な教示方法で様々な複雑なタスクを教示できる。また、ロボットの行動に適した視点と、その視点から得られる画像の中でロボットが注視すべき部分をテンプレート画像として保持しタスクを実行するために、複雑な環境などモデル化が困難な状況下においても応用できる。さらに、ロボットは人間に教示された行動だけでなく、その行動を組み合わせることで自律的に行動することができる。

5. おわりに

本論文では人間の支援のもとにロボットをロボパストに活動させるための人間ロボット協調システムを提案し試作システムを作成した。人間とロボットはテンプレート画像を介しコミュニケーションを行い、人間がロボットに指示や教示を行うだけでなく、人間はロボットからの内部表現の提示を観察しながらロボットに最適な意図伝達の方策を獲得する。

本手法の特徴は以下の3点にまとめることができる。1) DVSを利用することで、人間はロボットの行動に適した視

点を用いた指示が可能である。2)人間とシステムはテンプレート画像を用いた情報交換によりタスクの通信を行う。3)人間とシステムとのインタラクションを繰り返すことで、ロボットはより自律的な振る舞いを獲得し、この自律性に応じたタスクの伝達を行う。

今後は各VAがそれぞれ獲得した動作やサブタスクなどの内部状態をVA間で共有する方法と、人間の役割について研究を続ける予定である。

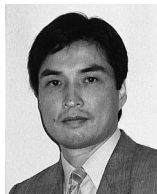
参考文献

- [1] Y. Nakamura, M. Mizukawa and K. Kanayama: "Multimedia Communication Pendant for Sensor-Based Robotic Task Teaching by Sharing Information: Concepts and Evaluations," Proc. Int. Conf. on Intelligent Robots and Systems (IROS'96), pp.1417-1424, 1996.
- [2] 森,小谷,石黒,安富,茅野: "構内道路を走行する自律移動ロボット", 日本ロボット学会誌, vol.5, no.5, pp.361-374, 1987.
- [3] 関本,坪内,油田: "環境中で走行経路を直接指示する移動ロボット操縦のためのヒューマンインターフェース", 第14回日本ロボット学会学術講演会予稿集, pp.337-338, 1996.
- [4] P.T. Boissiere and R.W. Harrigan: "TELEROBOTIC OPERATION OF CONVENTIONAL ROBOT MANIPULATORS," Proc. IEEE Robotics and Automation, pp.576-583, 1988.
- [5] H. Ishiguro: "Distributed vision system: A perceptual information infrastructure for robot navigation," Proc. Int. Joint Conf. Artificial Intelligence (IJCAI 97), pp.111-111, 1997.
- [6] 石黒,佐川,十河,石田: "複数の全方位視覚による人間の長時間行動認識", 第15回日本ロボット学会学術講演会予稿集, pp.553-554, 1997.
- [7] 柴田,松本,稲葉,井上: "人間が搭乗しその場で行動を指示できるパーソナル視覚移動ロボット Hyper Scooter の開発", 日本ロボット学会誌, vol.14, no.8, pp.1138-1144, 1996.
- [8] 藤沢,荒巻,吉村,首藤,阿久根: "視覚情報と言語情報を用いたヒューマンロボットインターフェースの実現法", 第14回日本ロボット学会学術講演会予稿集, pp.339-340, 1996.
- [9] 三宅: "ロボット言語と制御ソフトウェア", 日本ロボット学会誌, vol.14, no.6, pp.772-775, 1996.
- [10] The Nomad Series Catalog. Nomadic Technologies Inc., 1997.



岡田 慧 (Kei Okada)

1974年2月24日生。1997年京都大学工学部情報工学科卒業。現在、東京大学大学院工学系研究科情報工学専攻在籍。実時間視覚処理の研究に従事。(日本ロボット学会学生会員)



石田 亨 (Toru Ishida)

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。1983年から84年にかけて,米国コロンビア大学計算機科学科客員研究員。現在,京都大学大学院情報学研究

科社会情報学専攻教授。工学博士。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。1993年人工知能学会論文賞受賞,1996年人工知能学会10周年記念論文賞。情報処理学会,電子情報通信学会,人工知能学会,日本ソフトウェア科学会,IEEE,ACM,AAAI各会員。



石黒 浩 (Hiroshi Ishiguro)

1963年10月23日生。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。工学博士。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,現在,京都大学大学院工学研究科情報工学専攻助教授。視覚移動ロボット,能動視覚,分散視覚に興味を持つ。人工知能学会,情報処理学会,IEEE,AAAI各会員。

(日本ロボット学会正会員)