

オンラインモデル更新を用いた進化型自律移動ロボットの移動物体認識

Recognition of a Moving Object by an Evolutionary Autonomous Mobile Robot with an On-line Model Update Mechanism

○ 片田 喜章 (神戸大) 西原 志乃 (神戸大)
正 大倉 和博 (神戸大)

Yoshiaki KATADA, Kobe University, katada@rci.scitec.kobe-u.ac.jp
Shino NISHIHARA, Kobe University, Kazuhiro OHKURA, Kobe University

Classical AI approaches to object recognition of an autonomous robot induce critical problems e.g. the frame-of-reference problem, the object constancy problem and the symbol-grounding problem. The main problem with these approaches is their neglect of the interaction between the system and environment. In contrast to this, approaches in embodied cognitive science enable a robot to acquire an appropriate behavior through the interaction with environment. One of such approaches is the evolutionary approach. In this paper, a physical experiment with an autonomous mobile robot equipped with an omni-directional camera was conducted in order to investigate whether the evolutionary approach is applicable to recognition of a moving object. Evolution with an on-line model update mechanism was applied to shorten a long time for which all robots in a population are evaluated in a real environment. The best robot, evolved by the method using the Genetic Algorithms, showed a behavior of achieving the examined task.

Key Words: Evolutionary Robotics, Genetic Algorithms, On-line Model Update, Recognition

1 はじめに

古典的 AI で従来行われてきたマシンビジョンの物体認識・識別に対する主なアプローチは、物体の画像入力を、予め蓄えておいたテンプレートや画像のモデル、つまり、内部表象へ写像するというものであった。しかし、このアプローチを自律ロボットに適用する場合、フレーム問題・物体の定常性問題・記号接地問題など致命的な問題が生じる [1]。

この古典的 AI におけるアプローチに対し、システム自身が高い環境適応能力を有し、環境との相互作用を通して、自らの振舞いを獲得する身体性認知科学 [1] と呼ばれるアプローチがある。ここでは、入力パターンを内部表象に写像することはせず、ロボットが環境において自律的に振る舞い、自らの入力空間を構成することで (センサ・モータ協応)、タスク達成が可能となる。身体性認知科学の 1 つの分野に人工進化を用いて自律ロボットの制御機構を設計しようとする進化ロボティクス (Evolutionary Robotics:ER) がある。このアプローチに基づいて、センサ・モータ協応機構を進化的に設計し、自律エージェントによる物体形状識別を行う研究が報告されている [2][3]。

本稿では、移動物体の認識を可能とするセンサ・モータ協応機構を人工進化を用いて自律移動ロボットに獲得させることを目的とする。ここで、物体が移動する場合にはこれを捕獲し、移動しない場合には捕獲しないというタスクを取り扱う。このタスクの難しさは、ロボットが移動することで、止まっている物体でも移動しているかの如く入力が構成されてしまうことにある。進化させるすべてのロボットを実環境で評価するには膨大な時間がかかるため、我々の研究グループで提案しているオンラインモデル更新を用いたシミュレーションとの併用 [4] を行う。以下、2. ではオンラインモデル更新を用いた進化手法について述べ、3. では問題設定を述べ、実機実験を行う。4. では獲得したコントローラの特性を調べ、人工進化を用いて、自律移動ロボットによる移動物体認識が可能であるか検証する。

2 オンラインモデル更新を用いた進化

ER では、ノイズ等の影響や時間の経過に伴うハードウェアの機械的・電气的特性の変化・周囲の環境の変化 (光量、床面の摩擦係数等) により、センサ入力 s と、モータ出力 a で構成される実環境の入出力関係 $R(s, a)$ とシミュレーションの入出力モデル $M(s, a)$ との間にずれが生じる。この問題に対し、 $R(s, a)$ を基に $M(s, a)$ をオンラインで構成・更新する。このオンラインモデル更新を用いた進化計算プロセスは次のようにまとめられる (Fig.1)。

1. 全個体を実環境で評価する個体とシミュレーションで評価する個体に分ける。
2. 実環境で評価をしている間に、 $R(s, a)$ から s と a のサンプルデータを取得する。
3. 実環境での評価後、得られたサンプルデータを $M(s, a)$ を構成しているデータに加えることで $M(s, a)$ を更新し、更新された $M(s, a)$ を用いてシミュレーションで評価する。
4. 実環境とシミュレーションで得られた評価値に基づき選択を行い、遺伝的操作により次世代の個体を生成する。
5. 終了条件に達するまで 1→4 を繰り返す。

ここで、 $M(s, a)$ を更新する際に、 $R(s, a)$ との整合性を検定するために t 検定を用いる。検定の方法は以下のように行う。

- a) $M(s, a)$ を構成するデータ (データ数: n) の平均を μ_M 、標準偏差を σ_M 、 $R(s, a)$ から得られたデータの平均を μ_R とする。
- b) $\mu_M = \mu_R$ という仮説に基づいて有意性検定 (有意水準 $\alpha = 0.05$) を行う。

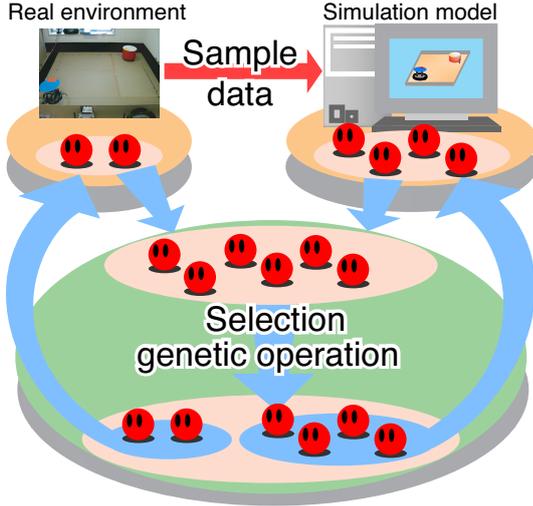


Fig.1 Evolution based on on-line model update

- c) 仮説が棄却されたとき、 $M(s, a)$ を構成するデータと $R(s, a)$ から得られたサンプルデータとの間に差が生じているため、 $M(s, a)$ を構成するデータを補正し、 μ_M を μ_R に近づけるように、差を埋めていく。
- d) 補正後の $M(s, a)$ を構成するデータに関して、仮説が棄却されなくなるまで 2 → 2 を繰り返す。

3 移動物体認識

3.1 問題設定

本稿では進化型自律移動ロボット（以降、ハンターロボット）を用いた移動物体認識問題を取り扱う。移動物体として用いるロボット（以降、対象ロボット）の運動パターンは、動いている場合（以降、移動試行）と停止している場合（以降、停止試行）とする。移動試行では、ハンターロボットは対象ロボットを捕獲し、停止試行では、ハンターロボットは対象ロボットを捕獲しないことをタスクとする。

3.2 実験環境

実験環境を図 2 に示す。実験環境は平面で、2400mm × 2200mm の長方形であり、四方は壁で囲まれている。

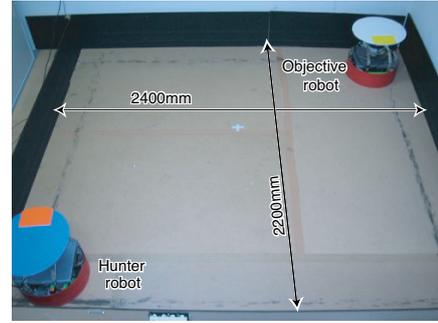
進化の対象はハンターロボットとする。ハンターロボットは全方位カメラを搭載し、画像処理とロボットの制御を行うノート PC を搭載している（図 2(b)）。直径 350mm、独立駆動型の 2 輪式で、コントローラとして隠れノードと出力ノードが完全相互結合したリカレント型の Pulsed Neural Networks(PNNs) を用いる。PNNs の入力ノード数は 128、隠れノード数は 1、出力ノード数は 4 とする。

対象ロボットは全方位カメラを搭載し、ロボットの制御を行うノート PC を搭載している（図 2(c)）。直径 350mm、独立駆動型の 2 輪式で、移動試行におけるコントローラはブライテンベルグビーグル [1] を基に全方位カメラを用いて障害物回避行動をするように設定した。

3.3 シミュレーションモデル

3.3.1 入力モデル

[ハンターロボット] 全方位カメラを用いる場合、一般的に角度情報は信頼性が高く計算も容易である一方、距離情報は計算が複雑になる。しかし、平坦な環境では距離情報は計算が容易であり



(a) Experimental environment



(b) Hunter robot with an omnidirectional camera



(c) Objective robot with an omnidirectional camera

Fig.2 Experimental setup

[5], 取得した画像の中心から同心円上では、全方位カメラの中心からの距離がほぼ一定となる。そこで、図 3(a) に示すように円周方向に一定の角度 (22.5°) で 16 分割し、半径方向に同心円で 4 分割して計 64 個のセルに画像を分割した。各セルにおける対象ロボットと壁の存在情報を PNNs の入力とする（図 3(b)）。シミュレーションではこの入力モデル s_M は一定であると仮定し、更新は行わない。

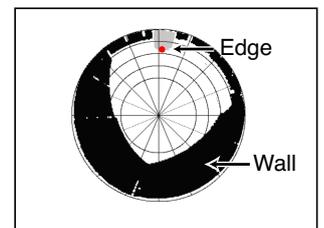
[対象ロボット] 移動試行において、対象ロボットは、全方位カメラを用いてハンターロボットと壁の位置情報を取得し（図 4）、その位置情報を基に障害物回避行動をとる。つまりハンターロボット自ら対象ロボットを捕まえない限り、対象ロボットは環境内を動き続けることができる。

3.3.2 出力モデル

[ハンターロボット] 左右の各モータコマンドは、前進 9 通り、後進 9 通り、停止 1 通りの計 19 通りの出力が可能であり、19 × 19 の計 361 通りのモータコマンドにより行動する。出力モデルは各モータコマンド (i, j) (i : 左モータコマンド, j : 右モータコマンド) に対する移動量 $(x_{(i,j)}, y_{(i,j)}, \theta_{(i,j)})$ とする。これらの移動量は、実環境から得られたサンプルデータを基に正規分布を用いて



(a) Camera image



(b) Binarized image

Fig.3 Omni-directional image plane of a hunter robot

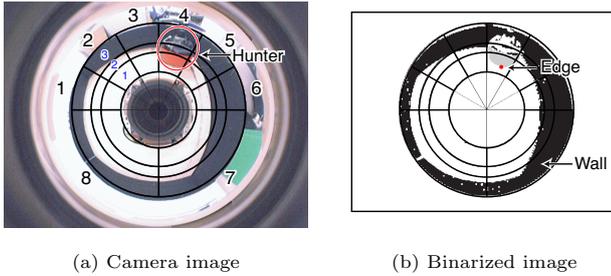


Fig.4 Omni-directional image plane of an objective robot

シミュレーション内でモデル化する [6]. 次に出力モデルの更新について述べる.

- i) 実験環境の上方に CCD カメラを設置し, 画像処理ボード (日立 IP5005) を用いて, ロボットの重心位置を取得する.
- ii) 0.5s の間隔で移動した距離を重心位置から測定し, そのときのモータコマンドに対する実環境での移動量 $(x_{(i,j)R}, y_{(i,j)R}, \theta_{(i,j)R})$ を取得する.
- iii) 実環境の評価が終わった後, 取得した移動量のデータを加える.
- iv) 2.に基づき, データ補正を行う. 仮説が棄却されたとき, 過去 g' 世代の間に実環境から得られたデータの平均 μ_R に出力モデルの平均 μ_M が近づくように補正する. $\mu_R < \mu_M$ の場合, μ_M を大きくするために, モデル環境を構築しているサンプルデータの中から値の小さいデータを削除する. 補正後の平均 μ'_M と μ_R について再び検定をし, 仮説が採択されるまでこれを繰り返す. 同様に $\mu_R < \mu_M$ の場合, 値の大きいサンプルデータを削除する.
- v) 各移動量の平均 μ と標準偏差 σ を計算し, $(x_{(i,j)M}, y_{(i,j)M}, \theta_{(i,j)M})$ を求める.

なお, 予備実験として, 全モータコマンドに対する移動量のサンプルデータを 10 ずつ用意し, 初期世代の出力モデル $(x_{(i,j)M}, y_{(i,j)M}, \theta_{(i,j)M})$ を構築する. 本稿では, 上記の手法を 50 世代毎に適用し, $g' = 500$ として $(x_{(i,j)M}, y_{(i,j)M}, \theta_{(i,j)M})$ を更新していく.

[対象ロボット] 左右の各モータコマンドは, 前進 4 通り, 後進 4 通りの計 8 通りの出力が可能であり, 8×8 の計 64 通りのモータコマンドにより行動する. 出力モデルの更新に関してはハンターロボットと同様とする.

3.4 実験設定

移動試行におけるハンターロボットと対象ロボットの初期姿勢の組み合わせを図 2(a) における {(ハンターロボット, 対象ロボット)} = {(上, 下), (右, 左)} とし, 停止試行におけるハンターロボットの初期姿勢を {(上, 右)} とする. すなわち 1 個体につき計 4 通りの実験を行う.

この 4 通りの実験を各個体毎に 4 回ずつ計 16 試行行う. 移動試行ではハンターロボットが対象ロボットを捕獲するか 40 ステップで試行終了とし, 停止試行では 40 ステップで試行終了とする. 進化計算に Operon GA [8] を用い, ハンターロボットの PNNs [9] の結合係数としきい値を変数とする. 遺伝子長 6700, 個体数 50 とし, トーナメントサイズ 6 のトーナメント選択を行い, エリート戦略 (エリート数 1) を用いる. 突然変異率は 0.00015 とし, 交叉は用いない. Operon GA の他の遺伝的操作の各パラメータの

設定は文献 [8] の設定に従う. 世代数を 2000 世代とし, 50 世代毎に前世代において評価値の高い上位 2 個体を実環境で評価し, 残りの個体と他の世代ではシミュレーションで評価する.

適応度関数を以下のように設定する.

$$Fitness = \frac{\sum_{i=1}^{TrialNumber} (1 - \frac{EvaluateStep_i}{MaxStep})}{TrialNumber} \quad (1)$$

ここで, $TrialNumber$: 試行数 (停止試行数 8, 移動試行数 8:計 16), $MaxStep$: 1 試行の最大ステップ数 (=40), $EvaluateStep$: 実行ステップ数 (移動試行ではハンターロボットが対象ロボットを捕獲するのに要したステップ数, 停止試行ではハンターロボットが対象ロボットから一定距離 (1950mm) 以下に入ってしまった合計ステップ数).

3.5 実験結果

各世代に対する適応度の推移を図 5 に示す. 進化初期ではほぼ半分の試行でのタスク達成を示す 0.5 を保ち, 350 世代付近でより高い適応度を獲得できる個体が現れ, その後, 緩やかに最大適応度と平均適応度が上がっている. 最終世代で得られた最良個体のコントローラについて, タスク達成率を調べるために, 新たに 40 試行 (移動試行 20, 停止試行 20) 行った. 結果を表 1 に示す. 対象ロボットの動きを認識している振舞いが安定して得られていることがわかる. このコントローラを用いたハンターロボットの振舞いの例を図 6 に示す. 移動試行では, ハンターロボットは回転・後進を行いながら対象ロボットを追いかけているが, 停止試行ではハンターロボットは初期位置からあまり移動しないことが分かる.

4 考察

3. において実験時間を考慮し, 対象ロボットの初期位置を固定したこと, および, 本問題設定ではハンターロボットの内側のセルに対応する距離に対象ロボットが入ることが移動を表すとも考えられることから, 対象ロボットの初期位置を変更した場合のタスク達成率を調べる.

初期位置への依存度を調べるために, 環境内でハンターロボットの各セルに対応する地点を対象ロボットの初期位置とし, 20 試行 (初期姿勢上向き試行 10, 初期姿勢右向き試行 10) 行い, タスク達成率を調べた. 結果を図 7 に示す. 場所によっては極端に低いタスク達成率を示すこともあるが, ハンターロボットの内側のセルにあたる初期位置の場合でも高いタスク達成率を示すこともある. このことから, 対象ロボットの初期位置を 1 点で固定しハンターロボットを進化させたため, 初期位置に対する頑健性は乏しいが, 動きから移動物体を認識していることがわかる.

このことを確かめるために, 試行の途中で対象ロボットの動きを停止させた場合のハンターロボットの振舞いを調べる. 表 2 に変化後のタスク達成率を示す. 対象ロボットの動きを変更し

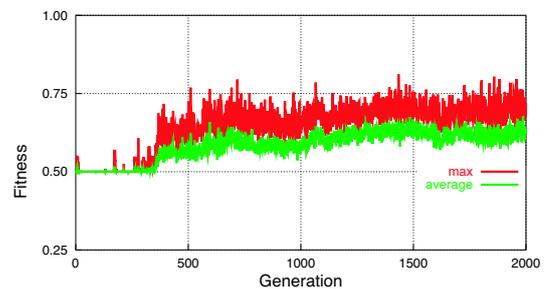
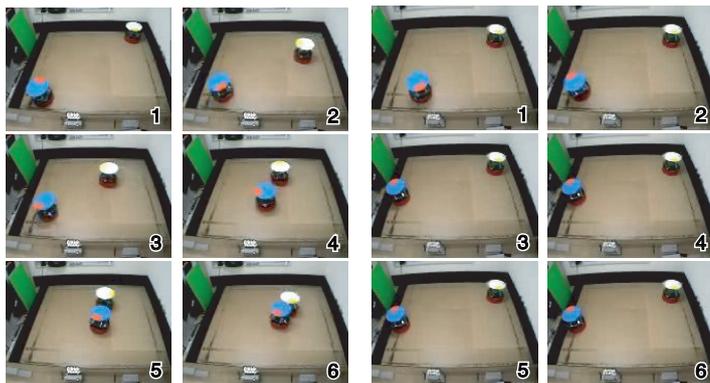


Fig.5 Maximum and average fitness at each generation



(a) Object with moves (b) Object without moves

Fig.6 Behavior of the best evolved robot in the real environment

Table1 Success rate(%) for each trial

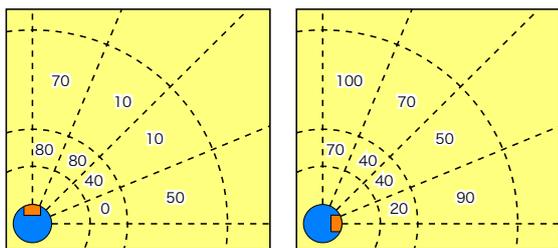
	Upward	Rightward
Move	60	70
Stop	70	100
Total trial	65	85

た時の位置がそのタスクの新たな初期位置となるが、タスク達成率は表1と比べそれほど低くなっていない。その中で動きの変更に対してうまく振舞えた例を図8に示す(番号5で動きを変更)。

対象ロボットが移動している間はハンターロボットは対象ロボットに接近し、対象ロボットが停止するとハンターロボットは対象ロボットから離れるという振舞いが見られた(図8)。以上より、獲得したコントローラは動きから対象ロボットを認識していると考えられる。

5 おわりに

本稿では、人工進化を用いることで、移動している物体を認識する能力を自律ロボットに獲得させることが可能であることを確認した。その進化手法として、オンラインモデル更新を用いた Operon GA を適用した。今後、視覚情報の取得法をも、人工進化の過程で決定されるようシステムを拡張していきたい。



(a) Upward (b) Rightward

Fig.7 Success rate(%) for the initial positions by the best evolved robot in the real environment

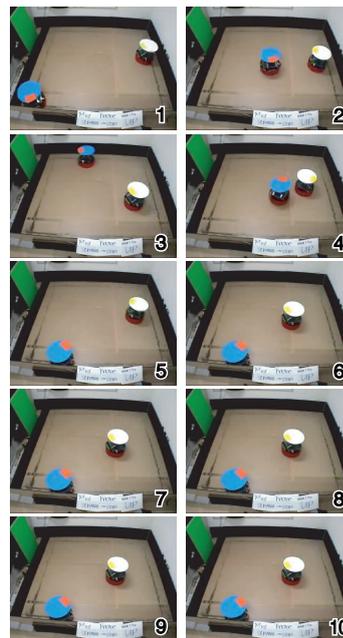


Fig.8 Behavior of the best evolved robot for varying the behavior of an objective robot in the middle of a trial

Table2 Success rate(%) for varying the behavior of an objective robot in the middle of a trial

	Upward	Rightward
move→stop	80	60

文 献

- [1] R. Pfeifer and C. Scheier, Understanding Intelligence, The MIT Press, 1999.
- [2] R. Beer, Toward the evolution of dynamical neural networks for minimally cognitive behavior, Proceedings of From Animals to Animats 4, pp.421-429, 1996.
- [3] S. Nolfi, Adaptation as a more powerful tool than decomposition and integration, Proceedings of the Workshop on Evolutionary Computing and Machine Learning, 13th International Conference on Machine Learning, 1996.
- [4] 高崎 真也, 片田 喜章, 大倉 和博, 田浦 俊春, 進化した自律移動ロボットのオンラインモデル更新に関する一手法, 第 47 回自動制御連合講演会, CD-ROM 予稿集 404, 2004.
- [5] 森 信人, 家田 純一, 松井 渉, 全方位移動機構と全方位視覚を有する小型ロボットによるサッカー競技の実現 -チーム OMNI の戦略-, 人工知能学会 第 6 回 SIG-Challenge 研究会論文集, Vol.1, pp.42-47, 2000.
- [6] 小森谷 清, 大山 英明, 谷 和男, 移動ロボットのためのランドマーク観測計画, 日本ロボット学会誌, Vol.11, No.4, pp.533-540, 1993.
- [7] D. Floreano, S. Nolfi and F. Mondada, Competitive Co-Evolutionary Robotics: From Theory to Practis, Proceedings of the 5th International Conference on Simulation of Adaptive Behaviour, pp.515-524, 1998.
- [8] 片田 喜章, 高崎 真也, 大倉 和博, 可変突然変異率戦略を用いた GA による進化ロボティクスへのアプローチ -ハンターロボットの進化的行動獲得への適用-, 第 17 回自律分散システム・シンポジウム, 講演論文集, pp.299-304, 2005.
- [9] W. Maass and C. M. Bishop, Pulsed Neural Networks, MIT Press, 1998.