強化学習によるパターンの意味付けと記憶に基づく行動の獲得

大分大学　柴田克成、沢津橋由人、宇都宮浩樹

Acquisition of Meaning of Patterns and Memory-based Behaviors by Reinforcement Learning

Katsunari Shibata, Yoshito Sawatsubashi and Hiroki Utsunomiya　Oita University

Abstract: In this paper, by the combination of reinforcement learning and a neural network, the authors try to provide an explanation for the question: why humans can acquire the meaning of patterns and memory-based behaviors, and to demonstrate it using a system with a real movable camera and four displays. When the system moved its camera to the direction of an arrow that was presented on one display, it could find a red circle and get a reward. One kind of arrow was chosen randomly among four kinds at each episode. After learning, the system could move its camera to the arrow direction, and some hidden neurons that represented the arrow direction not depending on the arrow pattern and kept the value after the arrow disappeared from the input image were observed. Generalization to some unseen arrows could be seen to some extent.

1. まえがき

われわれ人間は、さまざまな画像の中から、記号、シンボルを読み取り、その意味を把握し、それを記憶して適切な行動を行なうことができる。たとえば、本稿で取り上げる矢印の問題について考えてみる。われわれは標識等に矢印が書いてあれば、その矢印の方に進みなさいとか、矢印の方を見なさいという意味であると推測し、そちらを向いたり、そちらに進んだりすることができる。また、矢印が見えなくなって、その矢印の細かい形状は忘れても、向きの情報を記憶してそれに沿った行動を行うことができる。ロボットにこのような行動をさせる場合、矢印の向きを認識し、その結果を記憶して行動するプログラムを与えることを通常は考えるであろう。しかし、その際、われわれ人間はどうしてその矢印の意味を把握し、それを記憶して適切に行動できるようになるのかといった根源的な問題は脇に置かれることになる。

しかし、より柔軟で賢いロボットを開発するという観点から考えれば、与えられたプログラムに沿って矢印の方向を向くことよりも、矢印がどういう意味を持つのかを自分でわかるようになるにはどうすればよいかという根源的な問題に取り組むことこそが重要であると考えられる。Brooks も、「”抽象化”は知能の本質であり、解決が困難なところである」と述べ、普通は情報の抽象化の方法を与えて、その後をいかに獲得するかに焦点が当てられているが、実は、抽象化そのものこそ知能の本質ではないかと指摘している[1]。

筆者らは、今まで、強化学習とニューラルネット(NN)の組み合わせで、報酬を得て罰を避ける行動の学習を通して、内部に、そのために必要となる認識や記憶などのさまざまな機能が創発することを示して来た[2]。もしもわれわれ人間が、矢印というものに意味があり、その矢の向いている方向が重要であるということを学習によって獲得できるとしたら、その理由として最も容易に考えられることは、矢印の方を向いたら良いことがあったという経験に基づくということが考えられる。そこで本稿では、実際のカメラを用いたシステムを用い、矢印のパターンを見せ、矢印が向いている方向を見ると報酬が得られるという環境で、強化学習とリカレントニューラルネット(RNN)によって学習することで、RNN が多くの視覚センサ信号の中から矢印の向きの情報を抽出(認識)し、それを記憶して、報酬を得るようになるか検証した結果を報告する。

強化学習と RNN を用いた研究は従来からなされており、特に、Bakker らの研究では実ロボットを使って、後の行動選択のために必要なセンサ信号を特定し、記憶することを強化学習で学習できることを示している[3]。しかしながら、ここでは５つのセンサ信号のうちの一つをそのまま記憶すれば良い問題となっている上、センサ信号を圧縮し、離散状態表現を実現するために強化学習とは別の教師なし学習を適用するとともに、RNN も特殊な構造のものを使用している。本研究では、一般的な RNN を用いて、1,000 を越える画像信号から、強化学習の合目的性によって必要な情報を抽出して記憶し、適切な行動に結びつけることができるかどうかを検証する。

2. システム構成と学習方法

実験環境を図１に示す。カメラを中心とする半径 45cm の円周上に４台のディスプレイを設置する。ディスプレイ１、２のどちらかに青い矢印の画像が表示され、矢印が向いている方向の隣のディスプレイにはゴールを示す赤丸が、反対側の隣のディスプレイには、ゴールでないことを示す緑の×印が表示される。カメラは、最初矢印が画面中央に見える状態からスタートし、カメラを回転させてゴールを示す赤丸が真ん中に見えたら報酬を与えて試行を終了し、緑の×印が見えたら、罰を与える。途中、状態によっては、矢印も赤丸も緑の×印も見えないこともあるため、矢印の情報を覚えておかないとゴールに向かうことができない。

図２に本システムの構成を示す。本システムでは、学習によって記憶を形成することが求められるため、リカレント型のニューラルネットを用いた。本研究では、報酬や罰に基づく学習を通して、何を認識し、意

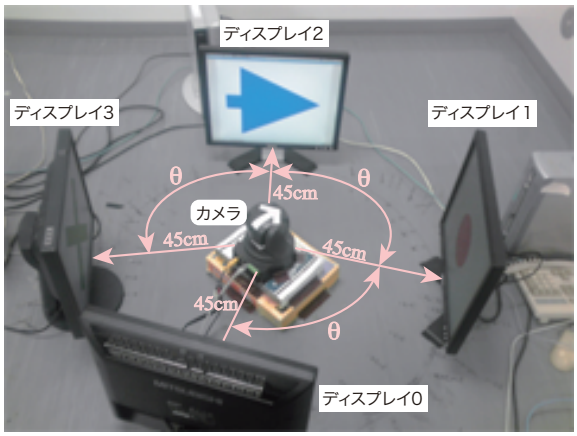

図１　実験環境

味付けして、記憶してそれを行動に結びつけるかの全過程を獲得することを追求するため、カメラから得られた画像の情報を前処理なしで直接ニューラルネットに入力した。具体的には、各ピクセルの RGB の値をそれぞれ-0.5 から 0.5 の値に線形変換して、画素数×3 の信号をリカレントニューラルネットに直接入力した。

カメラの動作信号は連続値を取るとし、連続値出力が可能な actor-critic[4]を採用した。そして、リカレントニューラルネットの出力を２つ用意し、一つは、状態の評価、つまり、critic として用い、もう一つはカメラの動作信号、つまり、actor として用いた。

学習は、強化学習のアルゴリズムに基づいて、教師信号を生成し、それを用いて教師あり学習を行った。まず、TD 誤差 $\hat{r}_{t-1}$ を

$$\hat{r}_{t-1} = r_t + \gamma P(\boldsymbol{s}_t) - P(\boldsymbol{s}_{t-1}) \qquad (1)$$

とする。ここで、r_t は時刻 t で与えられた報酬、γ は割引率(ここでは 0.986)、$\boldsymbol{s}_t$ は時刻 t でのセンサ信号ベクトル、$P(\boldsymbol{s}_t)$ は $\boldsymbol{s}_t$ を入力としたときの critic の出力に 0.4 を加えたものとする。critic の教師信号 $P_{s,t-1}$ を

$$P_{s,t-1} = P(\boldsymbol{s}_{t-1}) + \hat{r}_{t-1} = r_t + \gamma P(\boldsymbol{s}_t)、 \qquad (2)$$

actor の教師信号 $a_{s,t-1}$ を

$$a_{s,t-1} = a(\boldsymbol{s}_{t-1}) + \hat{r}_{t-1} rnd_{t-1} \qquad (3)$$

と計算する。ただし、$a(\boldsymbol{s}_{t-1})$ は $\boldsymbol{s}_{t-1}$ を入力としたときのニューラルネットの actor の出力、rnd_{t-1} は時刻 t-1 で出力に加えた乱数である。そして、$P_{s,t-1}$(実際にはこれから 0.4 を引いたもの)と $a_{s,t-1}$ を用いて、$\boldsymbol{s}_{t-1}$ を入力とした時のニューラルネットに対し、１回だけ BPTT 法[5]で学習させた。

３．実験

3.1 実験の設定と方法

前述のように、実験では４台のディスプレイを使用し、最初は、ディスプレイ間隔 θ を、ディスプレイ同士のすき間がほとんどなく、矢印、赤丸、緑の×印のいずれも見えないという状態がない $\theta = 52$ 度とした。その後、50 試行続けて 20 ステップ以内でゴールに到達するとそのレベルを終了し、ディスプレイ同士の間隔を 8 度広げて次のレベルへ移行した。３台のディスプレイを用い、真ん中のディスプレイに矢印を表示した予備実験では、学習後に、記憶した矢印の向きよりも、背景の違いが行動決定に大きく影響していることがわかった。そこで、ディスプレイの数を４台に増やし、右から２台目のディスプレイ１から左に進んだ場合と、右から３台目のディスプレイ２から右に進んだ場合で、途中に同じ背景が映るようにし、背景だけでは正しい行動の選択ができないようにした。最終的にはこれ以上広げられない $\theta = 100$ 度(図１の状態)まで広げた。カメラは SONY 製の EVI-D70 を使用し、ディスプレイは 19inch(１台は 17inch)のものを使用した。キャプチャする画像は 640×480 画素であるが、ニューラルネットの計算時間の関係から、この画像を 26×20 画素にリサイズした画像を入力として用いた。矢印の画像としては、右向きと左向きの矢印それぞれについ

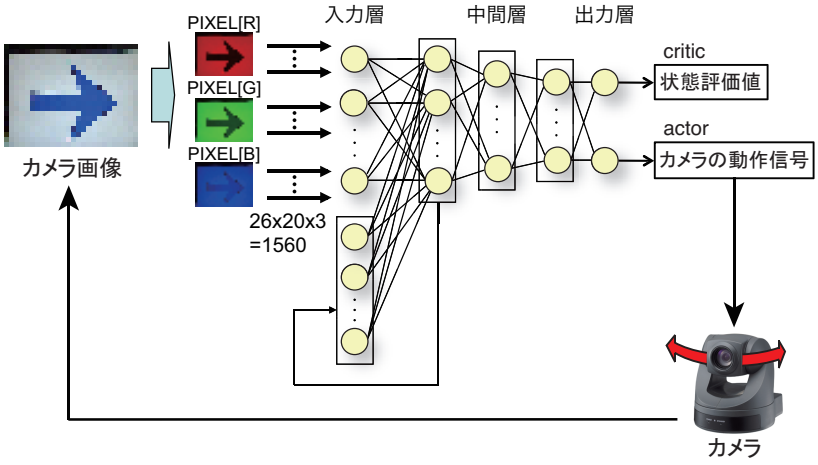

図２　実験システムの構成と信号の流れ

て、図５の上部に表示した４パターンを用意した。単純に入力パターン間の距離（各ピクセルの各色の階調値の差の絶対値を全画素全色について足したもの）を計算すると、左向きの矢印１の画像と一番近いものは右向きの矢印１であり、一番遠いものは、同じ向きである左向きの矢印４であった。

ニューラルネットは５層とし、リカレント構造として、入力側(下位)の中間層の値を次の時間の入力に加えた。入力する画像は 26x20=520 画素なので、ニューラルネットへの入力数は、各画素 RGB で３倍した 1560 個、中間層ニューロン数は、入力側(下位)から 300, 75, 20 個、出力は、critic と actor それぞれ 1 個ずつの 2 個とした。また、中間層、出力層の各ニューロンでは、-0.5 から 0.5 の値域を持つシグモイド関数を出力関数として用いた。また、出力に与える教師信号は、シグモイド関数の飽和領域を避けるため、-0.4 から 0.4 の間に制限した。カメラは actor の出力と比例した角度だけ動かし、出力が 0.4 のとき１ステップで約 7.7 度回転する。critic の値は、ニューロンの出力に 0.5 足した値とした。ゴールした際の報酬は 0.9 とし、(2)式の $P(s_t)$ を 0 として、0.9-0.5=0.4 を教師信号として与えて critic を学習させた。また、ゴールでない方向にカメラが動いて×を中心付近に捉えた場合には、罰として-0.1 を(2)式の r として与えて、試行を続け、同一試行内で 10 回罰を受けたときは試行を終了した。それ以外の場合は、r=0 として(2)式の教師信号に基づいて学習した。ただし、プログラム上のバグのため、θ=60 度から θ=84 度までの間、罰は与えられなかった。また、BPTT でさかのぼる時間は、10 ステップとした。試行錯誤のために加える乱数 rnd は、$\pm 0.5e^{-0.001x}$（ただし、x はゴールに到達すると１増え、失敗すると４減り、0 未満にならない）の範囲の一様乱数とし、学習が成功するとその範囲が減少するように設定した。

3.2 実験結果

3261 試行ですべての学習を終了した。最初のレベルの学習は 1287 試行かかり、途中、カメラが片側にしか行かない状況もあったが、その後は各レベル 158 試行から 490 試行の間で終了した。

図３と図４に学習後のカメラの動きの例を示す。図３は、右から２番目のディスプレイ１に左向きの矢印１が表示された場合、図４は、右から３番目のディスプレイ２に右向きの矢印４が表示された場合である。それぞれ、カメラの画像の変化、および、その際の critic と actor の時間変化の様子を示す。いずれの場合も、カメラの画像の変化および actor の出力から、カメラは矢印の方向に動き、赤丸のゴールに到達していることがわかる。また、critic はほぼ単調に増加しており、理想曲線と近い変化をしていることがわかる。また、step7～10 あたりでは、矢印もゴールも見えず、同じ背景のところを通過しているため、入力画像はほぼ同じであるが、その間も critic は増加し、actor も正しい符号の値を出力しており、矢印の向きの情報を記憶して、正しい状態評価と行動ができていると言える。

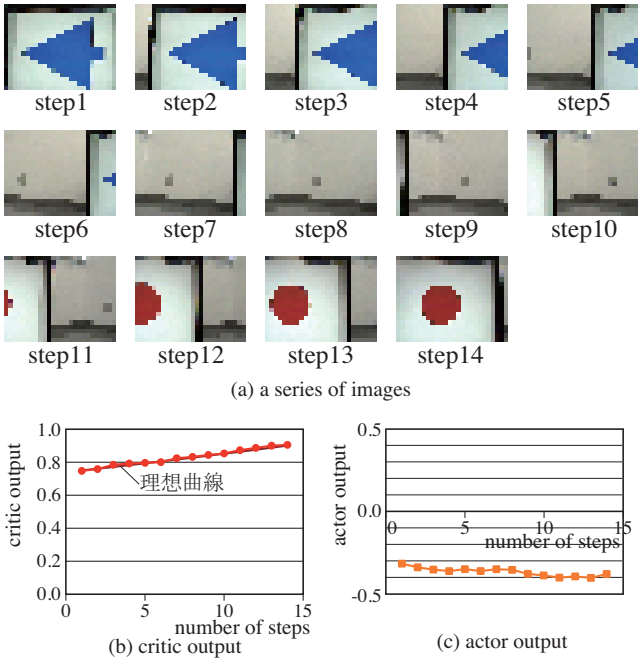

図３ ディスプレイ１に左向きの矢印１を表示した場合のカメラ動作による画像の変化と critic、actor の値の変化

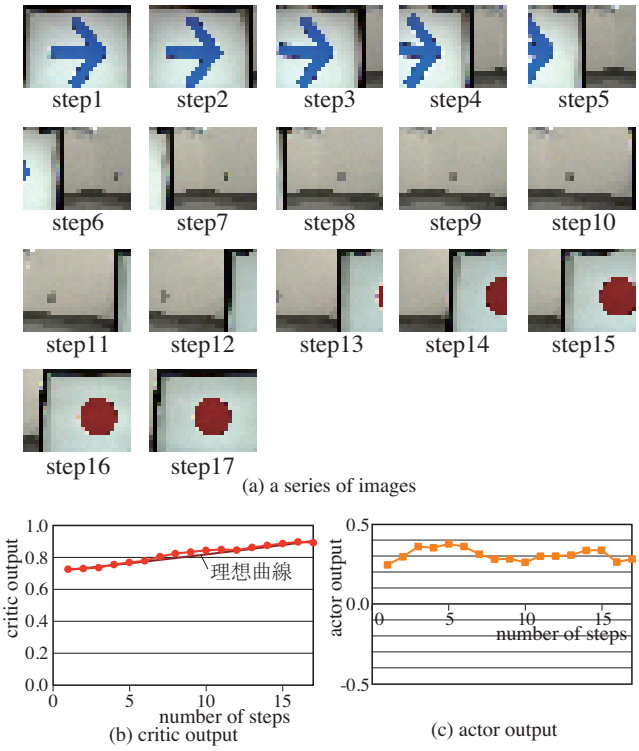

図４ ディスプレイ２に右向き矢印４を表示した場合のカメラ動作による画像の変化と critic、actor の値の変化

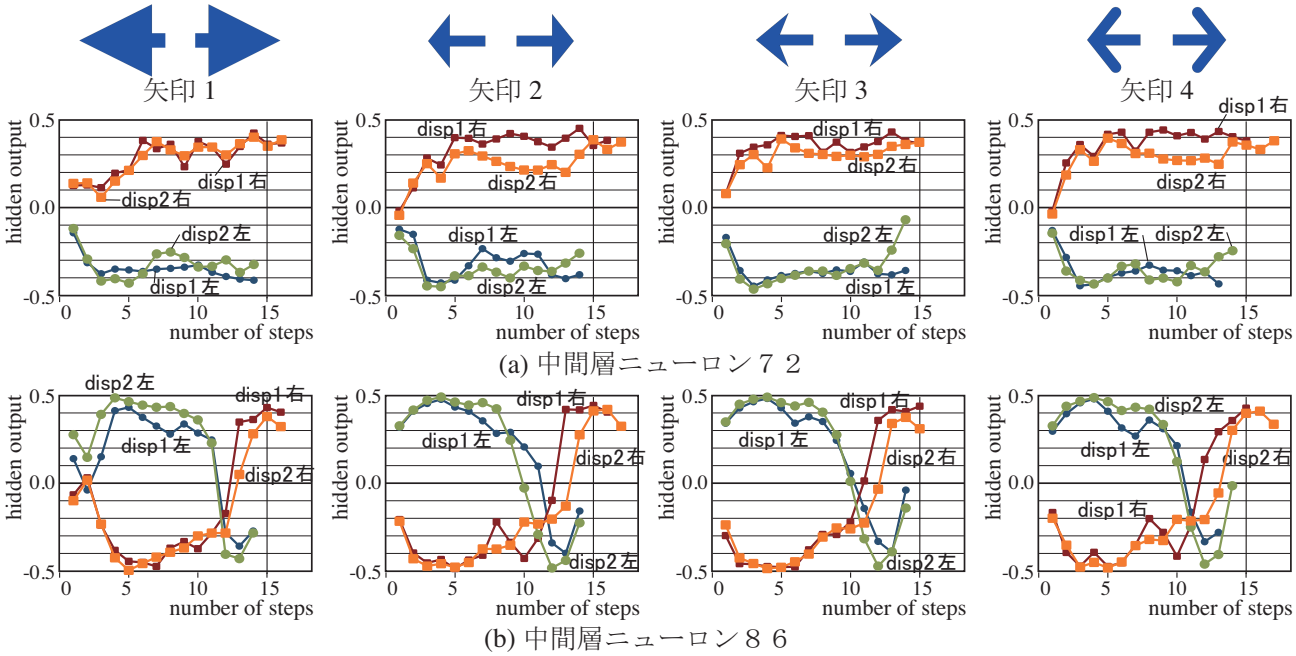

図５　各矢印を提示した場合の、２つの下位中間層ニューロンの出力の変化。”disp2 左” は、ディスプレイ２に左向きの矢印を提示した場合を表す。

次に、図５に、下位中間層の 300 個のニューロンのうち、矢印の向きを検出と記憶に貢献していると思われるニューロンの出力の変化を、矢印ごとに示す。中間層ニューロン 72 は、矢印の向きによって出力が異なり、矢印が見えなくなった後もその出力を保持している。しかし、試行開始当初は、矢印２，４の向きの識別があまりできていない。逆に中間層ニューロン 86 は、試行当初からそれらの識別が明確にできているが、当初の矢印１の識別と、矢印の向きの記憶はできておらず、ニューロン間の役割分担がうかがえる。

また、学習に用いていないパターンを提示した結果を表１に示す。７種類の未学習パターンのうち、５パターンでは、動き始めるのに時間がかかったものもあるが、最終的には正しいゴールに到達しており、ある程度の汎化能力はあることがわかる。失敗はいずれも右矢印であった。学習に用いたパターンでも右向きの矢印の場合のステップ数が多いことから、左向きの矢印の場合と比較して、環境とリカレントネットの相互作用で形成される引き込みが弱かったと考えられる。

まとめ

矢印の方向を向くと報酬がもらえるという設定で、入力画像からパターンの意味づけ(矢印の方向)とその記憶を、リカレントニューラルネットを用いた強化学習を通して学習できることを、実際のカメラを使ったシステムで示した。

入力パターンの位置をランダムにずらすなど、少し問題を難しくすると学習が困難になり、改善の余地は大きい。しかし、報酬や罰に基づく学習の奥深さをある程度示すことができたのではないかと考えている。

表１　矢印を提示した場合のゴールまでのステップ数。数字はディスプレイ１と２の場合の平均。×は不成功、△は片方のディスプレイのみ成功を表す。矢印５から１１は未学習パターン。

(a) 矢印１		(b) 矢印２		(c) 矢印３		(d) 矢印４	
14	16	14	16.5	14	15	13.5	16
(e) 矢印５		(f) 矢印６		(g) 矢印７		(h) 矢印８	
16	△	16	32.5	15	18	15.5	20
(i) 矢印９		(j) 矢印１０		(k) 矢印１１			
14	×	14.5	16.5	14.5	18		

謝辞

本研究は、日本学術振興会科技学術研究費補助金基盤研究(B)#19300070 の補助を受けた。ここに謝意を示す。

参考文献

[1] R.A. Brooks: Intelligence without Representation. *Artificial Intelligence*, Vol. 47, pp. 139-159 (1991)

[2] 柴田克成, “強化学習とニューラルネットによる知能創発”, *計測と制御*, Vol. 48, No. 1, pp. 106-111 (2009)

[3] B. Bakker, et al., “A Robot that Reinforcement-Learns to Identify and Memorize Important Previous Observations”, *Proc. of IROS 2003*, pp. 430-435 (2003)

[4] A.G. Barto, et al., "Neuronlike Adaptive Elements That can Solve Difficult Learning Control Problems", *IEEE Trans. SMC, Vol. 13*, pp.835-846 (1983)

[5] D.E. Rumelhart, et al., "Learning Internal Representation by Error Propagation", in Parallel Distributed Processing (1986)