

第6章 強化学習に基づく能動認識

6.1 背景

我々生物は、様々なセンサを使って外界の情報を取り込み、それに基づいて目的に沿った適切な行動を行うことができる。また、外界の情報を効率的に取り込むために、より良い認識ができる位置にセンサを動かしたり、その場の状況に即したセンサ情報を選択するといった、いわゆる能動認識[山崎 92]の機能も有する。図6.1に示したように、センサを移動させることによって、より良い認識ができ、外界の状態をより正しく認識することができれば、より適切な動作を行い、より効率的に目的を達成することができる。例えば、カエルが目の前を飛んでいる物体を発見したとすると、目を動かして目の前の物体がエサかどうかを認識し、エサだと認識した場合は、飛びついて食べることによってエサという報酬を得る。つまり、生物にとっては、認識自体も、そして、そのためにセンサを移動させるという動作も、食欲などの本能を満たすために必要な一連の動作の一つであると考えることが可能である。このことは、認識や認識のための動作も、食物を食べるために食物に近づくといった目的に直接結び付くような動作と同様に捉えることができることを意味している。

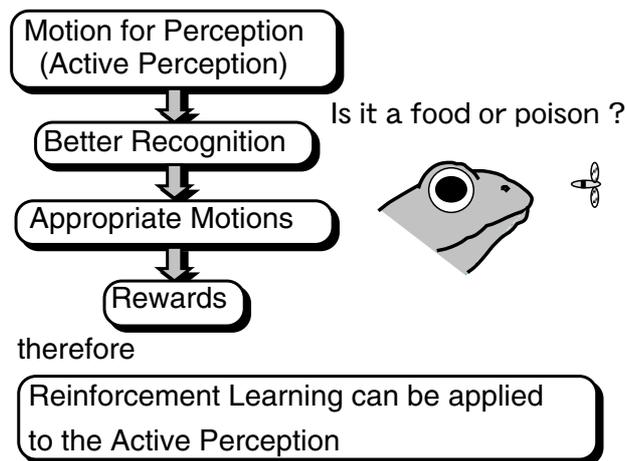


図6.1 なぜ能動認識に強化学習を適用するか

一方、第1章および前章で述べた強化学習は、試行錯誤を基により良い解を学習するという特徴を持ち、教師あり学習（ここでは、直接理想出力である教師信号を与える学習を教師あり学習と定義する）と比較してより自律的な学習である。さらに、通常、与える情報量が教師あり学習に比べ

て少ないことから、学習に時間がかかるが、その分環境に適した柔軟な学習ができる。従来、強化学習は主として、機械（ロボット）に与えられた目的を達成するための動作を学習させるために使われてきた。また、最近では、強化学習にニューラルネットを適用することによって、ニューラルネットの学習、汎化能力が強化学習に有効であることが確認されている。

前述のように、能動認識におけるセンサの移動も、また認識自体も、目的達成動作の一つと捉えることができる。そこで、我々は、強化学習で能動認識の機能を学習させることを考え、さらに、ニューラルネットを用いることで、ニューラルネットの学習、汎化能力を利用することを考えた。通常の強化学習では、強化信号は一連の動作を行った後に得られる、いわゆる遅延強化信号となるため、過去の動作をいかにさかのぼって学習するかが問題となる。しかしここでは、簡単のため、認識結果に対して外部から逐次強化信号が得られるという仮定をおいた。そして、その強化信号からセンサの移動と物体の認識を同時にニューラルネットに学習させるためのシステムの構成法と学習法を提案する。

従来、能動認識は、外部から全く情報を得られない場合について、阪口らによって、観測によるエントロピー減少量の期待値を最大化するという観点から定式化され、触知覚におけるセンサ情報の選択に適用されている[阪口 91][阪口 93]。

Whit1 ける評価を学習によって獲得し、その評価が最大になる注視点を選択するという方法をとっており、センサの移動を連続的な動作として捉えて学習させていない。また、ニューラルネットは用いていない。

6.2 学習アルゴリズム

6.2.1 全体構成

図6.2のような、センサと階層型ニューラルネットよりなる能動認識のための学習システムを提案する。本論文では、センサとして可動視覚センサを例として用いる。視覚センサは複数のセンサセルよりなり、各センサセルは、映し出された物体の面積が各センサセルの全体の面積に対して占める割合を出力するものとする。ニューラルネットは、各センサセルから信号を受取り、層に従って出力を計算する。出力層ニューロンは、認識用ニューロンとセンサ移動用ニューロンの2種類よりなる。そして、認識用ニューロンの出力値が認識結果を示し、センサ移動用ニューロンの出力値がセンサの移動量を決定するという構成になっている。ここでは、簡単のため、認識用ニューロンとセンサ移動用ニューロンをそれぞれ1個ずつ描いてあるが、いずれのニューロンも複数個あってもよい。また、ニューロンの出力関数はシグモイド関数とし、出力値は0から1の値域を持つ連続値とする。

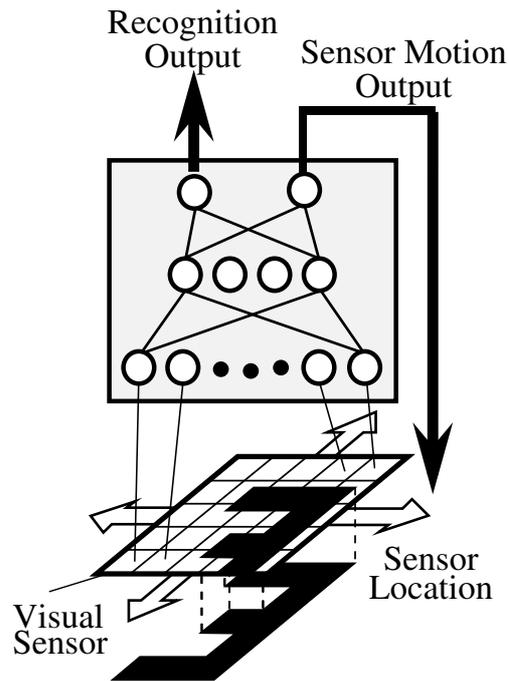


図 6.2 ニューラルネットを用いた能動認識システムの構成

6.2.2 認識の学習

始めに、認識出力の学習に関して述べる。前述のように、認識自体も目的達成のための動作の一つと捉えることができるため、強化学習によって学習を行う。図 6.3 に認識の学習の際の信号の流れを示す。まず、認識用の出力 o_p に対し、平均 0 の微小な乱数 rnd_p を

$$a_p = o_p + rnd_p \quad (p=1,2,\dots,P) \quad (6.1)$$

P: 認識出力の数

のように付加して認識の信号 a_p とする。この時、出力が複数ある場合は、各出力に対し別々の独立した乱数を加える。そして、ここでは、理想的な認識パターン *ideal* が存在するものとして、外部の認識結果評価部 (Recognition Evaluator) において、

$$\phi = - \sum_{p=1}^P (ideal_p - a_p)^2 \quad (6.2)$$

ϕ : 認識出力の評価値

というスカラー量で認識出力が評価される。そして、現時点 T での認識結果 a_p が前の時刻 $T-1$ と比べて良くなったか悪くなったかを表わすスカラー量 r が

$$r(T) = \phi(T) - \phi(T-1) \doteq \frac{d\phi(T)}{dt} \quad (6.3)$$

と計算され、強化信号として出力される。そして、システムはこれを受け取り、自乗誤差 e_p および誤差信号 ε_p を

$$e_p = (r \cdot rnd_p)^2 = \varepsilon_p^2 \quad (6.4)$$

として、つまり、教師信号 s_p を

$$s_p = o_p + r \cdot rnd_p \quad (6.5)$$

としてBP法[Rumelhart 86]に従いニューラルネットを1回だけ学習させる。

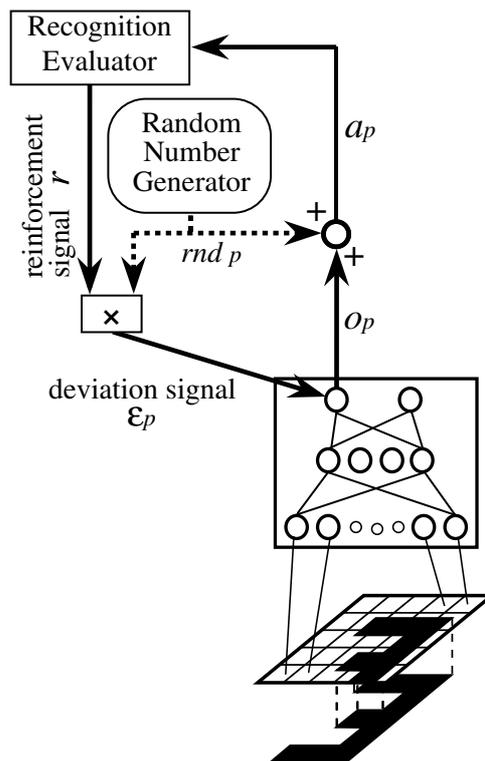


図 6.3 認識の学習

この学習によって、ニューラルネットは現在見えるセンサの画像に対して正しい認識を行おうとする。その結果、センサが、与えられたパターンをうまく認識できる位置にある時には、学習によって次第に正しく認識できるようになるが、物理的にうまく認識できないような位置にセンサがある場合は、いくら学習しても正しく認識できるようにはならない。これに対して、次節で述べるセ

ンサ移動の学習により、システムはより正しく認識できる方向へセンサが移動していくことを学習する。

認識の学習に関しては、理想的な認識パターンを用いて、直接教師あり学習を適用する方が簡単であり、収束も速い。しかし、ここでは、強化学習による認識の学習の可能性を検討する意味で、あえて教師信号から強化信号を生成して学習を試みた。これは、サルに文字を見せて、対応する正しい札を挙げた場合にエサがもらえるという状況に近い。

6.2.3 センサ移動の学習

次に、センサ移動の学習について述べる。図 6.4 にセンサ移動の学習の際の信号の流れを示す。まず、センサ移動用の出力 o_m に対し、平均 0 の微小な乱数 rnd_m を

$$a_m = o_m + rnd_m \quad (m=0,1,\dots,M) \quad (6.6)$$

M: センサ移動用出力の数

のように付加した値 a_m を求める。認識の場合と同様に、出力が複数ある場合は、それぞれに別々の乱数を加える。そして、この a_m を用いて、

$$x_m(T+1) = x_m(T) + \alpha(a_m - 0.5) \quad (6.7)$$

$x_m(T)$: 時刻 T の視覚センサの位置

α : 単位時間あたりのセンサの最大移動量を決定する定数

のように視覚センサを移動させる。移動した後、視覚センサが再び画像を取り込み、ニューラルネットは新しいセンサからの情報を受け取る。そして、再びニューラルネットの計算を行う。こうして得られた新しい認識用出力から、前節のように、認識結果評価部から前の認識用出力よりも良くなったかどうかを示す強化信号 r をもらい、

$$e_m = (r \cdot rnd_m)^2 = \varepsilon_m^2 \quad (6.8)$$

$$s_m = o_m + r \cdot rnd_m \quad (6.9)$$

の値から B P 法に従ってニューラルネットを 1 回だけ学習する。この学習によって、ニューラルネットはより正しい認識ができる方向へのセンサの動作を獲得することができる。

上記の認識の学習およびセンサ移動の学習は並列に行う。つまり、センサ移動用出力に乱数を足したものにしながらセンサを動かし、新しい画像を得る。そして、その得られた画像からニューラルネットの計算を行い、得られたセンサ移動用の出力は次のセンサの移動に用いる一方、認識用の出力はさらに認識出力に対する乱数を加えた後に評価される。従って、ここで生成される強化信

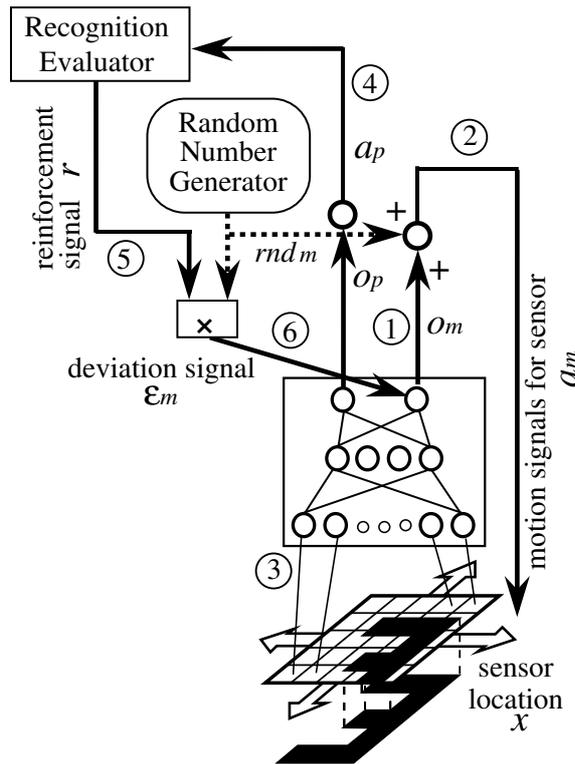


図 6.4 センサ移動の学習

号の計算に使われる認識結果 a_p は、認識用出力に加えた乱数 rnd_p とセンサ移動用出力に加えた乱数 rnd_m の両者の影響を受けている。また、認識用出力、センサ移動用出力が複数個ある場合は、それだけの乱数の影響を受けることになる。そして、得られた強化信号 r は、認識用出力の学習と1単位時間前のセンサ移動用出力の両方の学習に用いられる。この時、加えた乱数間の干渉が考えられるが、加える乱数は平均が0で、それぞれ互いに独立であるため、他の乱数の影響の期待値は0になる。よって、多数の試行を繰り返すことにより、認識、センサ移動共に正しい方向に学習が進む。

また、認識およびセンサ移動の学習は、共に強化学習に基づいているため、式(6.3)と式(6.6)、式(6.4)式(6.5)と式(6.8)式(6.9)といったようにほとんど同一の式で学習方法を記述することができる。これは本学習の特徴である。ただし、加える乱数 rnd の値域については認識と動作に分けて調節する必要がある。

6.3 シミュレーション

6.3.1 1次元センサ動作

図 6.5 に示すように、1次元に配置された6個のセンサセルを持つ左右に移動可能な視覚センサが、大中小(それぞれ視覚センサのセンサセル2個、4個、6個分に相当)の物体を認識するという問題を考える。システム全体は図 6.6 のようになる。図 6.7 のように物体が視覚センサの端に位

置する時は、その物体の大きさは認識できないため、正しく認識するためには、物体がある程度視野の真ん中に見えるようにセンサを移動させることが必要となる。認識用の出力ニューロン数は3個とし、理想出力を大中小の物体それぞれに対し、(0.1, 0.1, 0.9), (0.1, 0.9, 0.1), (0.9, 0.1, 0.1) とした。視覚センサは、1次元の運動を行うため、センサ移動用ニューロンは1個とした。中間層ニューロンは3個設け、ニューラルネットは6 - 3 - 4の3層とした。学習の具体的なフローは以下のようになる。まず、物体と視覚センサの相対位置の初期値を、視覚センサからの出力の合計が1以上となる範囲内で、乱数を用いて決定する。そして、物体を見せてから60単位時間経過するかまたは物体が消え失せるまで、つまり、6個のセンサ出力がすべて0になるまで続け、その後次のパターンを見せるということを繰り返し行った。各センサセルは1×1の大きさとし、センサの1回の動作は-0.2から0.2の範囲、つまり式(6.7)の α を0.4とした。

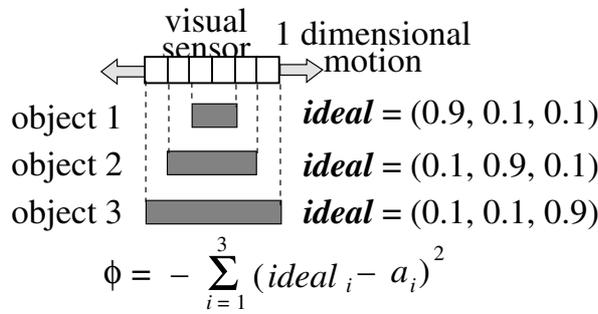


図 6.5 シミュレーションで用いたパターン

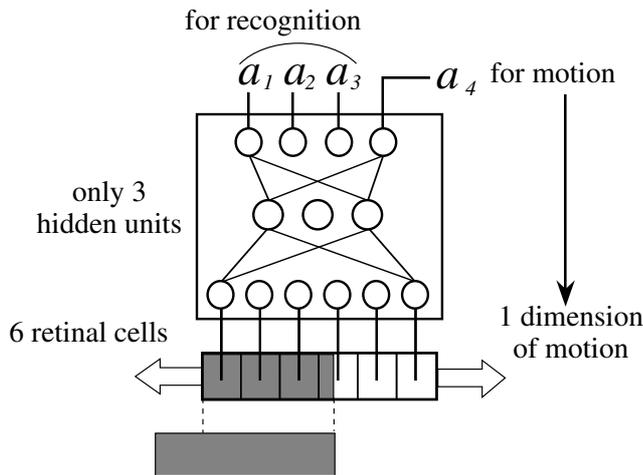


図 6.6 シミュレーションのシステム構成

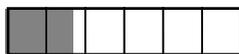


図 6.7 どの物体が提示されたか区別が付かない視覚イメージ

学習の結果を図6.8に示す。それぞれ、各物体を見せた時の物体の中心とセンサの中心の相対位置に対する認識用出力、センサ移動用出力をプロットしている。センサ移動用出力を見ると、学習によって、いずれの大きさの物体をいずれの場所に見せた時も、物体を中心近くに捉えるようにセンサが移動するようになっていることがわかる。また、その中心付近での認識用出力を見ると、対応するニューロンの出力値が高い値となっており、正しい認識ができていていることがわかる。そして、各パターンとも物体が端の方へいくほど、他のパターンとの区別ができなくなっていることもわかる。このように、区別不可能または区別が困難な場合は、区別ができる方向にセンサを動かし、正しく認識を行っていることから、効率的な認識を実現できたと言える。

中間層ニューロンは前述のように3個で学習することができた。表6.1に学習後の各重み値の値を示す。これより、それぞれの中間層ニューロンの役割は、おおむね、(1)両サイドに物体が見えている場合を検出するニューロン、(2)物体が中心に見え、かつ、物体が小さいものを検出するニューロン、(3)物体が主に視野の右側にあることを検出しているニューロンとなっていることがわかった。つまり、(1)最も大きい物体を識別する、(2)最も小さい物体を識別する、(3)センサを右に動かすべきか左に動かすべきかを決定するためにそれぞれ用いられていることがわかった。そして、中間の大きさの物体の認識出力は、最初の2つの中間層ニューロンが発火しない時に発火するようになっている。

表6.1 学習後のニューラルネットワークの重み値

(a) Input layer Hidden layer

		hidden layer		
		1	2	3
input layer	1	5.88	-6.12	2.12
	2	2.02	-0.91	2.78
	3	0.44	-2.80	0.17
	4	0.40	-2.78	-0.54
	5	2.01	-1.15	-0.88
	6	5.77	-5.81	-2.90
	bias	-8.46	6.49	-0.23

(b) Hidden layer Output layer

		output layer			
		1	2	3	4
hidden layer	1	0.56	-5.40	4.88	-0.56
	2	6.49	-6.94	0.60	-0.32
	3	-0.23	0.12	-0.18	6.34
	bias	-2.50	3.03	-2.52	-2.82

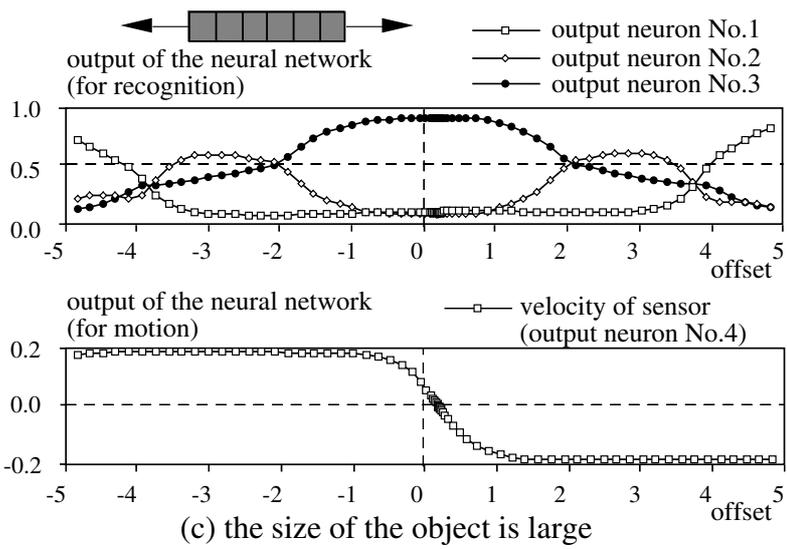
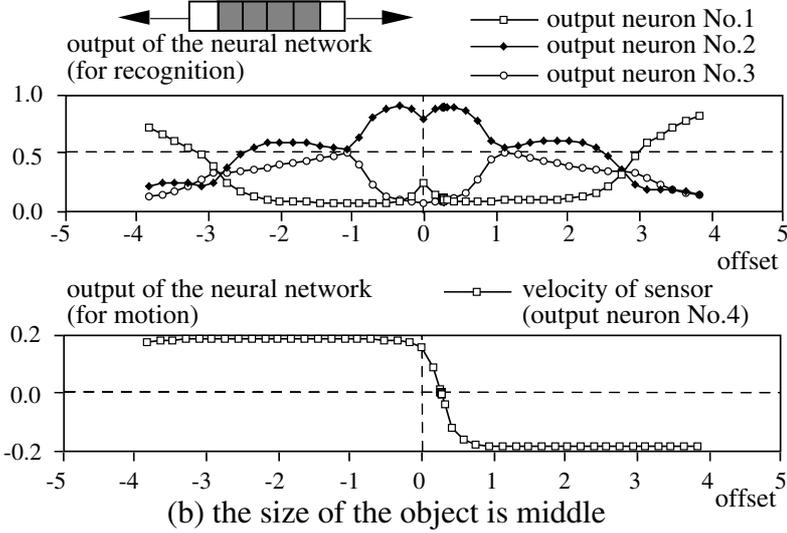
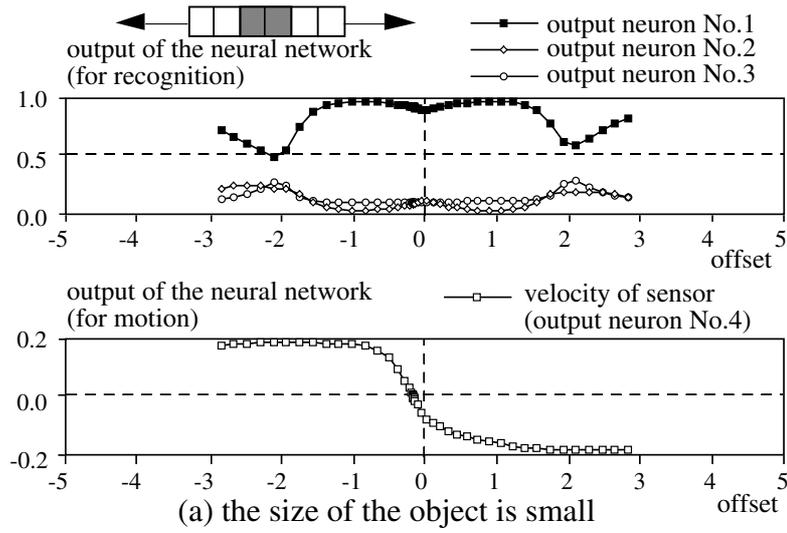


図 6.8 シミュレーションの結果 (物体とセンサの相対位置に対する各出力)

6.3.2 2次元センサ動作および小さいセンサによる識別

従来、文字認識をする際に、文字の重心にセンサを移動させてから認識を行うという方法がよく用いられてきた。そこで、最後に、認識する物体より小さいセンサを用意し、物体の重心にセンサを持ってきても認識できないような場合についても、上記の方法が通用するかどうかを確認した。視覚センサおよび2種類の提示パターンを図6.9に示す。視覚センサは、 2×2 とした。そして、図6.10(a)のように、視覚センサを重心に持ってくると、どちらのパターンを提示しても、4個のセンサ出力がすべて1になり、パターンの区別ができない。また、図6.10(b)のような場合も区別ができず、逆に、図6.11(a)や(b)のような場合は区別することができる。

これで、学習させた結果を図6.12に示す。ここでは、認識用の出力を1個とし、パターン1の場合は1を、パターン2の場合は0を理想出力として与える。そして、図6.12(a)では、パターン1を提示した場合のセンサの位置に対する出力の値を、図6.12(b)はパターン0を提示した場合の出力の値を1から引いた値を白から黒の色で示している。この図より、図6.11で示したように正しく認識できる場所では、正しく認識をし、物体の重心にセンサ中心がある時のように正しく認識できない場所(図6.10)では、どちらともつかない出力値であることがわかる。また、白抜きの矢印でセンサの動作方向を示している。これから、センサは正しく認識ができるところへ移動していることがわかる。また、白の太線は視覚センサ中心の軌道の例を描いている。このことから、視覚センサは徐々に色の黒い方向に向かい、正しく認識できる場所で止まることわかる。また、その途中では、一旦認識の評価値が下がる場合もあることがわかる。

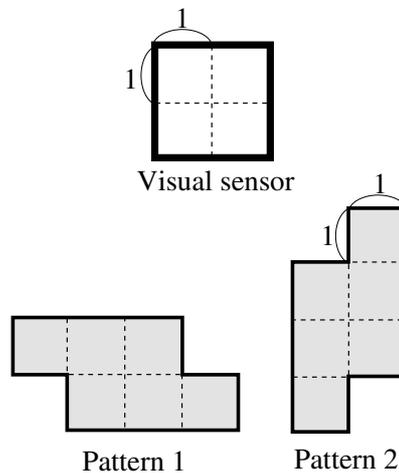


図6.9 シミュレーションで用いたパターンと視覚センサ

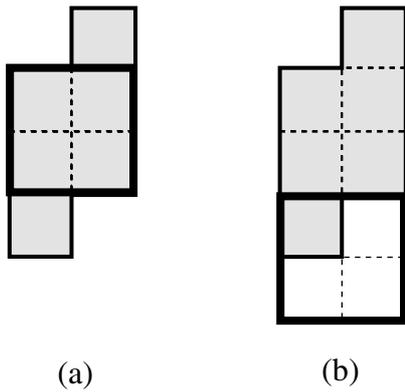


図 6.10 識別不可能なイメージ

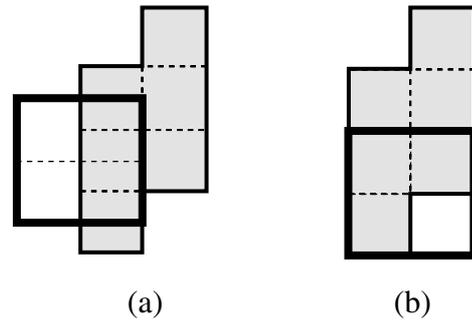


図 6.11 識別可能なイメージ

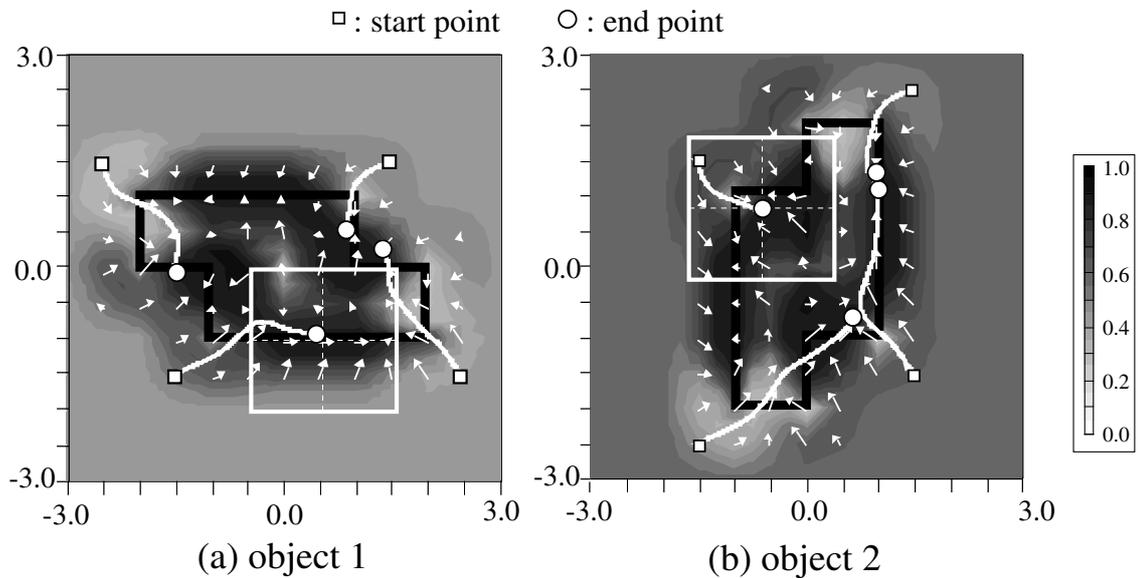


図 6.12 学習後のセンサの位置に対する評価値（色）とセンサの動作ベクトルおよびセンサの動作軌跡の例
（黒い太い線が提示パターン、白い四角がセンサを表わす）

6.3.3 簡単な数字認識と能動認識による効率的認識の検証

能動認識を利用したパターンの識別問題の学習において、識別するパターン数がセンサ動作の次元に比べて大きい場合を考える。するとまず、識別パターン数が多いことでパターン間の識別が難しくなる。さらに、認識用出力の数がセンサ移動用出力の数に比べて多くなるが、識別が難しいところではセンサの移動を学習させればよいことから、効率的な認識が実現できる。よって、センサを移動させないであらゆる見え方に対して認識用の出力を学習させる場合と比較して中間層のニュー

ーロン数が少なくても可能性が高いと考えられる。そこで、0 から 9 までの数字を認識させる問題をシミュレーションした。そして、あわせてセンサに 2 次元動作をさせた場合に問題がないかを確認した。

提示したパターンと視覚センサを図 6.13 に示した。視覚センサは、5 個 × 5 個の 2 次元に配列されたセンサセルを持ち、3 個 × 5 個の各マス目を 0 か 1 に塗りつぶした 10 個のパターンを識別させた。1 個のセンサセルの受容野および提示パターンのマス目の大きさは共に 1 × 1 とし、各センサセルは、投影されたパターンが受容野に占める割合を 0 から 1 の間の値で出力するものとした。例えば、"8" を提示した時にセンサ位置を x, y 共に中心から 0.5 ずつずらした時にセンサから獲得されるイメージは、図 6.14 のようになる。また、認識用出力の数は提示パターン数と同じ 10 個とし、理想出力は、各提示パターンに対応する出力を 0.9、その他を 0.1 とした。センサ移動用の出力数は 2 個とし、それぞれセンサの x 方向 y 方向の移動に用いた。視覚センサは単位時間あたり x 方向 y 方向共に最大 0.05 移動できることとした。ニューラルネットの中間層ニューロン数は 10 個とした。そして、1 つのパターンを提示した時の初期センサ位置は、25 個のセンサ出力の合計が 1 を越える場所からランダムに選択し、200 単位時間経過するか、センサ出力の合計が 1 を下回るまでの間、センサの動作、ニューラルネットの計算および学習を各単位時間毎に行った。その後新しいパターンを提示し、同じことを繰り返していった。

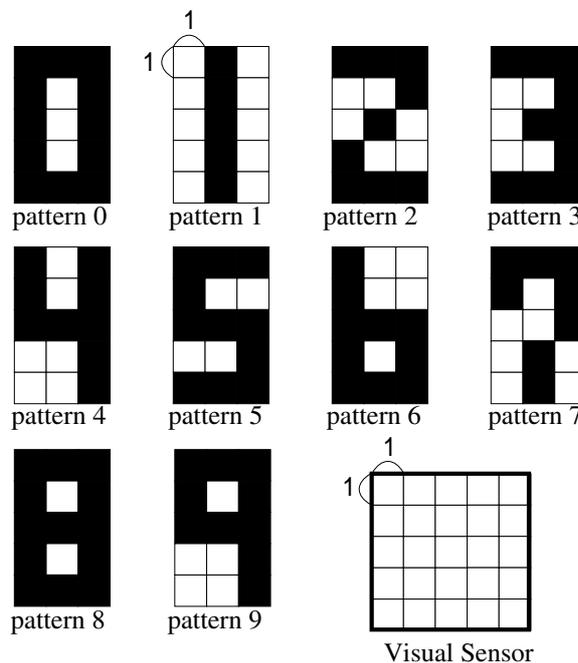


図 6.13 シミュレーションで用いたパターンと視覚センサ

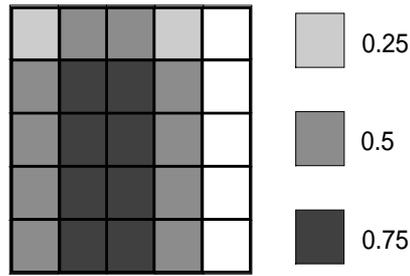


図 6.1.4 サンプル視覚イメージ (8 を提示)

これで、学習させた結果を表 6.2 および図 6.15 に示す。表 6.2 では、学習終了後に、各パターンを、初期位置を変化させて 1000 回提示し、200 単位時間経過した後に、各提示パターンに対応する認識出力が全認識出力中で最大値となった回数を示している。全パターンの平均をとると、99.3%の認識率になり、残りの 0.7%は、見失った、つまり、視覚センサからの信号の総和が 1 を下回ってしまったことを示す。図 6.15 では、0 から 3 までの 4 パターンを提示した場合について、センサとパターンの相対的な位置に対する認識結果の評価値 ϕ (式 (6.2)) を濃淡で示し、さらに、その時の動作出力 (ベクトル) を矢印で表わしている。ただし、各格子点の内、センサ信号の合計が 1 を下回る場合は矢印を示していないため、矢印が表示してある部分がほぼパターンを見失っていないセンサ位置となっている。この結果、提示パターン毎に認識しやすいセンサ位置があり、センサをそこに移動させるように学習が進んでいることがわかる。また、全パターンについて最終的にセンサが到達した位置を見ると、 3×5 の提示パターンの中心 $(x, y) = (0, 0)$ と提示パターンの重心が異なるものについては、全般的に、重心に近いことがわかった。

比較のために、センサを初期位置に固定し、認識だけ学習をさせた場合のシミュレーションも行った。ただし、初期位置を前のシミュレーションと同じにすると物理的に識別不可能なパターンも含まれてしまうため、 $-1.0 < x < 1.0$ 、 $-0.5 < y < 1.5$ という前のシミュレーションと比較してかなり狭い範囲の中から乱数によってセンサ位置を選択し、学習を行った。

結果を、表 6.3 に示す。この表は表 6.2 と同様に、各パターンを 1000 回提示した時に、各提示パターンに対応する認識出力が全認識出力の内最大であった回数を示す。これを見ると、誤認識がかなりあり、総平均で 93.2%の認識率となった。これに対し、中間層のニューロンを 20 個に増やすと、認識率が 98.9%、30 個に増やすと 99.5%と認識率の向上が見られた。これより、中間層ニューロン数 10 個で認識率が低かったのは、中間層ニューロン数が少なすぎたことが原因と考えられる。センサの移動まで学習させた場合には、中間層ニューロン数 10 個で 99%以上の認識率を実現できたことと比較すると、センサの移動まで学習させることによって効率的な認識が実現でき、より少ない中間層ニューロン数で学習ができたと言える。

表6.2 数字の認識結果

		Number of being maximum output unit (1000 presentations/pattern)										
		0	1	2	3	4	5	6	7	8	9	disappear
Presented Pattern	0	996	0	0	0	0	0	0	0	0	0	4
	1	0	985	0	0	0	0	0	0	0	0	15
	2	0	0	995	0	0	0	0	0	0	0	5
	3	0	0	0	999	0	0	0	0	0	0	1
	4	0	0	0	0	991	0	0	0	0	0	9
	5	0	0	0	0	0	982	0	0	0	0	18
	6	0	0	0	0	0	0	995	0	0	0	5
	7	0	0	0	0	0	0	0	989	0	0	11
	8	0	0	0	0	0	0	0	0	996	0	4
	9	0	0	0	0	0	0	0	0	0	994	6

correct mistake disappear
 9922 0 78

表6.3 センサを固定した場合の数字の認識結果

		Number of being maximum output unit (1000 presentations/pattern)										
		0	1	2	3	4	5	6	7	8	9	
Presented Pattern	0	732	0	0	0	0	193	0	0	75	0	
	1	0	1000	0	0	0	0	0	0	0	0	
	2	0	0	883	2	0	49	0	66	0	0	
	3	0	0	0	974	0	24	0	2	0	0	
	4	0	0	0	0	912	2	73	10	0	3	
	5	0	0	0	0	0	1000	0	0	0	0	
	6	0	0	0	0	10	0	965	17	8	0	
	7	0	0	0	0	0	0	0	1000	0	0	
	8	0	0	0	0	0	0	0	0	999	1	
	9	10	0	0	0	80	0	9	4	42	855	

correct mistake
 9320 680

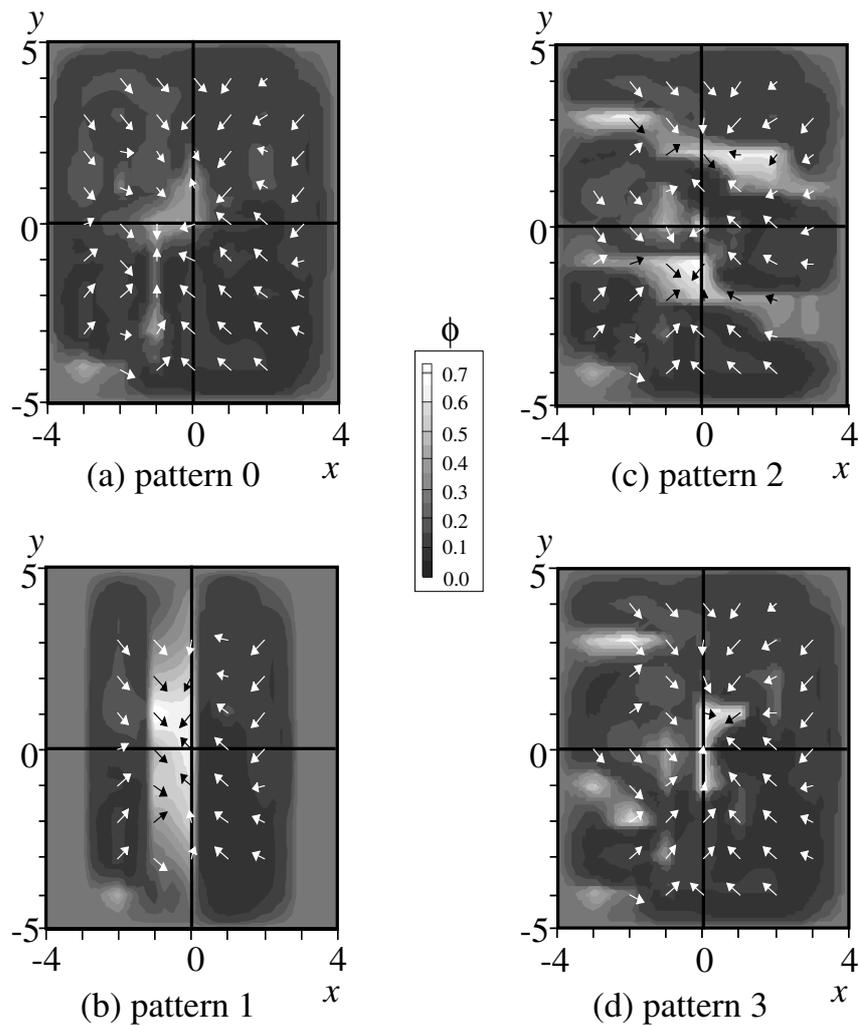


図 6.15 学習後のセンサと物体の相対位置に対する評価値とセンサの動作ベクトル

6.4 考察および今後の課題

上記のシミュレーションを通して気づいた点、問題点を述べる。

まず始めに、数字の認識の問題において、センサの移動まで学習させた場合、上記のように、少ない中間層ニューロン数で効率的な学習が実現できたが、さらに学習を進めていくと、逆に認識率が落ちるという現象が見られた。0のパターンを提示した場合の結果を図 6.16 に示す。この図のセンサの動作ベクトルと評価の値を見ると、図の右上の で囲んだ部分のように、センサが、提示パターンを見失う方向に動いていってしまったり、その少し下の で囲んだ部分のように、局所的に見てまわりより正しく認識ができるところにトラップされて、最も認識が正しくできる所まで移動することができず、結果的に誤認識をしてしまっていることがわかった。これは、学習の初期には、全体としての大まかなセンサの動きを学習するが、学習が進むにつれて、評価の曲面のより細かい凹凸に適応したより細かい動作を学習することができるようになり、結果的にローカルなとこ

るにトラップされてしまうためと考えられる。これは、非常に大きな問題であるが、遅延強化学習において使われるような、後に得られる報酬を事前に評価するような手法を使って解決できるのではないかと考える。ただし、この場合は、最終ゴールをどう設定するかという問題が残る。

また、1回の認識動作の完了をどうするかという問題もある。本論文中のシミュレーションでは、視野内をある程度以上の速度でセンサが動いた時に、センサが端から端まで十分に移動できるということから時間を設定し、その時間の経過を認識動作の完了とした。その他、センサが動作しなくなった時や最大認識出力と2番目に大きい出力の差が一定以上になった時等を認識動作の完了とする規準が考えられる。しかし、我々人間が行っている認識は、文脈や置かれている状況等によって

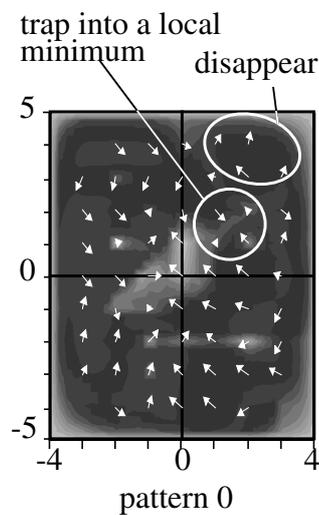


図 6.16 学習を続けた場合の評価値とセンサの動作ベクトル

変化してくるため、完了の規準もより複雑なものと考えられる。

本研究は、元々能動認識を学習することによってより適切な動作を行い、より報酬が得られるという発想に基づいたものである。従って、この能動認識の学習を通常の強化学習といかに結びつけていくかという点も今後の大きな課題である。また、前述のような遅延強化学習を適用する場合の最終ゴールや1回の認識動作の完了をどうするかといった問題も、本来の目的指向の強化学習の中に組み込むことによって、特に意識することなく、自ずと学習によって獲得されてくるものではないかと考える。

また、ここでは、認識のコーディングを、従来文字認識等でよく使われるように、各パターンに対して出力を1つ割り振って、そのパターンが提示された時はその出力の理想値を0.9、他を0.1とした。しかし、これも効率的なパターンコーディングとは言えない。そこで、これに関しては、強化学習と統合することで自律的にコーディングまで決定されることが期待される。

次に、行ったシミュレーションにおいて、学習の初期には、センサは、その位置によらず直線にある方向に動き、その結果視野から物体が見えなくなった。そこで、物体が見えなくなったらペナルティを与えた、つまり、式(2)で求めた評価値からさらにある値(ここでは、0.2)を引くと

いう設定を行ったところ、物体が視野の外に消えないようにするセンサの移動が早期に学習でき、学習の加速が見られた。我々人間も、物体が視野から消える状態を他から区別して認識しているように見える。このことから、我々も視野から物体が消えた状態を検出し、何らかのペナルティの信号を生成する機構を持っている可能性も考えられる。また、前章の物体追跡のモデルのところでも述べたように、外界の情報の変化がなくならないようにという学習によって物体の追跡動作を学習させることが可能である。このモデルとの融合も有効であると考えられる。

また、上記で用いたニューラルネットは、リカレント構造になっていないため、過去の履歴を認識や動作に利用することができない。しかし、我々は、例えば物体を見失っても、消えていった方に目を動して物体を再び見えるようにすることができる。このように、過去の履歴を反映した認識および認識のための動作を実現するためには、リカレント型のニューラルネットの導入が今後必要であると考えられる。

我々が物体を認識する時を考えると、通常視野の中心に物体を捉えるが、本論文で提案したような学習によって説明が付くのではないかと考えられる。数字の認識の実験では、動作後の視覚センサの中心が提示パターンの中心よりも提示パターンの重心に近くなる傾向があることを述べた。このような傾向は、単純な図形を人間に見せた時の視点の留まる位置に関する Kaufman らの実験結果 [Kaufman 69] でも見られる。また、小さい視覚センサを用意し、提示パターンの重心にセンサを移動すると逆に認識できないような設定で学習を行ったところ、重心ではなく、認識しやすい位置にセンサを動かして正しい認識を行うこともできた。また、視野の中心でパターンを捉えることが多いことから、われわれの目の網膜上の視細胞の密度が中心ほど大きいという不均一性は、能動認識を行うための効率的な形であると考えられることができる。

また、本論文では、1つのニューラルネットの出力を認識用とセンサ移動用に分けるという形をとったが、認識用とセンサ移動用にニューラルネットを分割することも可能である。最初のシミュレーションにおいては、中間層のニューロンが、(1)(2)は認識用に、(3)はセンサ移動用に使われていることから、ニューラルネットを一体化したことによる中間層ニューロン数の減少の効果は見られなかった。しかし、数字の認識のシミュレーションでは、認識用ニューロンとセンサ移動用ニューロンの両方と大きな結合を持つ中間層ニューロンが存在したことから、両ニューラルネットを一体化したことによって中間層ニューロンの効率的な利用に結びついた可能性が大きい。また、それ以外には、ニューラルネットを一体化することによって、与えられた問題に対し、中間層ニューロンを認識用とセンサ移動用に柔軟に配分できるという利点が考えられる一方、学習の収束が遅くなるという欠点が予測される。

最後に、我々生物との比較を行う。我々人間は、通常、サッケード、つまり、跳躍的な視点の移動を行うことによってパターンを認識していると言われている [乾 93]。本学習では、視覚センサの動きは連続的であり、そこが大きく違うところである。サッケードを行う場合でも、跳躍先をどこに持っていかは学習が必要であり、こう考えることによって本学習が適用できると考える。また、サッケードを行う場合、サッケード中には認識を行わなくても良いため、移動中の認識のための時間が節約されるという考え方もできる。今後、両者を比較検討し、生物の良い点を取り入れていく必要がある。

6.5 まとめ

認識や認識のための動作を目的達成のための動作の一つであると捉えることにより、強化学習によって両者を学習することを提案した。そして、能動認識におけるセンサの動きと認識自身を強化学習によって並列に学習させるためのシステムの構成と学習方法を提案した。認識も認識のための動作も共に強化学習を用いているため、ほとんど同じ学習方法で両者の学習を実現することができた。

この学習システムを用いて、簡単な認識問題のシミュレーション行ったところ、このシステムは、センサの動かし方に関する情報を一切与えていないにもかかわらず、物理的にパターンを識別できないような位置にセンサがあっても、センサを動かして正しい認識を行うように学習ができた。ただし、学習をさらに進めると、認識に対する評価値のローカルミニマにトラップされたり、視野からはずれる方法へセンサが動いてしまうという状況が見られた。

また、うまく認識できた時の最終的なセンサの位置は、提示パターンの重心近くに行く傾向が見られたが、単純に重心にセンサを移動させると区別ができないようなパターンを用意したところ、区別ができる場所にセンサを移動させる学習を行うことができた。

また、0から9の10個のパターンについて簡単な文字認識の問題を学習させたところ、全てのパターンについて、センサの動作後ほぼ正しい認識ができるようになった。そして、センサを動かさないであらゆる見え方に対して認識を学習した場合と比較して、ニューラルネットの中間層のニューロン数が少なくても高い認識率を示すことがわかった。また、学習後のニューラルネットを解析したところ、中間層ニューロンでは、認識、センサ動作を決定するために、センサ信号である入力空間を効率的にコーディングしていることもわかった。