

LF-013

パターン情報ベースの命題論理式のための推論機の提案  
Reasoning Machine for Pattern-based Propositional Logic

須藤 明人†  
Akihito Sudo

坪山 学†  
Manabu Tsuboyama

張シンリ†  
Chenli Zhang

佐藤 彰洋†  
Akihiro Sato

長谷川 修†  
Osamu Hasegawa

## 1. まえがき

人工知能における論理に関する研究の歴史は長い。しかし非定常な環境下において汎用的なタスクを扱う必要のある知能ロボット等にとっては既存の推論システムの枠組みでは不十分である。既存の推論システムが不十分である理由はいくつかあるが、筆者らは既存のシステムのほとんどがシンボルベースの推論しか行えないことが重要な理由のひとつであると考えている。代表的なシンボルベースの推論機としてプロダクションシステムがあるが、これは事前に人間の専門家が if-then ルールの形でシステムにシンボルベースの知識を与えておく必要がある。システムの扱うタスクの範囲が限定されていていれば、必要とされる知識を事前にシステムに与えることでタスクを解決できる。一方、非定常な環境下で自律的に活動することを要求される知能ロボット等にとっての十分な知識を事前に全て列挙することは不可能であり、新たな環境に適応するために自律的に知識を追加学習していくことが必要となる。このようなシステムが実世界からセンサーを通じて獲得する情報はパターン情報であり、シンボル化はされていない。この情報を利用してシンボルベースの推論機によって推論を行うためにはパターン情報をシンボル化する必要があるが、現在のパターン識別機の性能はまだ不十分であり、例えば猫と犬を識別することすら簡単なことではない。そのため現在のところパターン情報をシンボル化する装置と従来のシンボルベースの推論機を組み合わせることはあまり良い方法ではない。将来的に人間と同等の能力を有する識別機が開発されたとしても、ふたつの理由からシンボルベースの推論機だけでは不十分である。第一にシステムが自分でシンボルを生成しなければならないという問題がある。非定常な環境ではシステムに事前に全てのシンボルを与えることは不可能であり、システムは環境に対応して自らシンボルを生成することが求められるが、シンボル生成は行動や推論を含めた思考の結果なされるものであるためシンボル化のためにパターン情報を利用した推論が必要となろう。第二にシンボル化に馴染まないパターン情報が存在するという問題がある。例えば風景や絵などの画像の色彩や構図を扱うときにはパターン情報をそのまま扱う必要がある。

そこで本論文では、パターン情報をシンボル化せずに推論を行うためのシステムを提案する。提案手法はパターン情報を原子命題とし、この原子命題に連言(かつ)、選言(または)、否定を付加した命題を含意(ならば)で結ぶことで作られる任意の if-then ルールを学習することができる。例えば、A から F をパターン情報としたとき、「 $A \rightarrow B$ 」や「 $((A \wedge B) \vee C) \rightarrow (D \wedge \neg E \wedge F)$ 」といった知識が学習の対象となる(本論文では連言、選言、否定、含意を  $\wedge$ ,  $\vee$ ,  $\neg$ ,  $\rightarrow$  を用いて表す)。If-then ルールを学習したシステムに対

しファクトにあたるパターン情報が入力されると、学習した if-then ルールを利用して演繹推論を行い、いくつかの結果を導き出すことができる。もし「 $(A \wedge B) \rightarrow (C \vee D)$ 」と「 $C \rightarrow (E \wedge \neg F)$ 」を学習したシステムに対し「 $A \wedge B$ 」を入力すると「 $C \vee D$ 」「 $(E \wedge \neg F) \vee D$ 」を出力することができる。提案手法が扱う範囲はシンボルベースの推論でいえば命題論理の枠組みで演繹推論を行うことに対応する。

パターン情報ベースの推論機を非定常な環境下で自律的に活動することが必要な知能ロボット等へ応用する場合、上段で述べた機能に加えて追加学習、汎化性能、ノイズ耐性の4点が必要になる。まずシステムが学習すべき if-then ルールを人間が事前に全て用意し与えることはできないため、システムは知識を追加的に学習できることが必要である。また、学習によって得た知識と類似したデータがファクトとして入力されても入力データと似ている知識を用いて推論を行える汎化性能があることが望ましい。ただ、類似データを用いた推論がある程度行えるにしてもそれには限界がある。そのため、シンボルベースでは単一の if-then ルールで表せる知識もパターン情報ベースでは複数の if-then ルールとして学習することが必要になってくるだろう。しかしこれによって推論結果に重複が生じてしまう恐れがある。例えばシステムが環境から  $\{a_i \rightarrow b_j \mid i, j = 1, 2, 3, \dots\}$  といった複数の if-then ルールを学習したとする。これらが別々の知識であるとシステムが認識すると、ファクトとして  $a_1$  を入力すれば推論結果として  $b_1, b_2, \dots$  が得られる。学習データが本当に別の知識であればそれでよいが、学習した複数の知識がシンボルベースでは  $A \rightarrow B$  という単一のルールで表せる知識であった時は、計算量の削減や推論結果の利用の観点から、状況に応じ推論結果の重複を避けて一つだけ推論結果を出力することもできたほうがよい。さらに、実環境下ではシステムへの入力にノイズが混じることは避けがたく、一定のノイズ耐性を持っていることが求められる。

提案手法はパターン情報で表される if-then ルールの単なる蓄積やそれを用いた推論にとどまらず、上記の要求も満たしている。これらの機能を実現するために、提案手法はクラスタリングやトポロジー学習を目的としたニューラルネットワークモデルである SOINN[1] や連想モデルである SOINN-AM[2] のアルゴリズムを拡張して用いている。これにより追加学習、類似データの取り扱い、推論結果の重複の回避、ノイズ耐性の機能を持たせている。

本論文に直接関係する先行研究は非常に少ない。最近では山根らが非単調型ニューラルネットワークモデルを利用したパターンベース推論[3]を提案しているが、連言(かつ)、選言(または)、否定を扱うことができないという大きな問題のほか、2値データしか扱えない、追加学習によって既存の知識が徐々に破壊されるといった問題をかかえている。そのため知能ロボット等への応用には不十分である。

† 東京工業大学 総合理工学研究科

‡ 東京工業大学 像情報工学研究施設

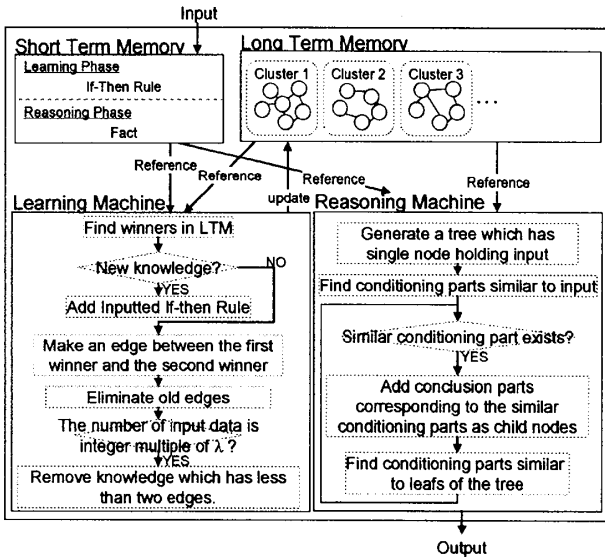


図1: 提案手法のアーキテクチャ

2. 提案手法

図1に示したように、提案手法は短期記憶、長期記憶、学習機、推論機からなる。短期記憶には入力データが一時的に保存される。長期記憶には学習したif-thenルールがエッジで結ばれることでオンラインでクラスタリングされながら蓄積されていく。提案手法は、if-thenルールを獲得する学習フェーズと、獲得したif-thenルールと入力されたファクトをもとに推論を行う推論フェーズに分けることができる。学習フェーズではパターン情報を原子命題とするif-thenルールが入力され、所定のアルゴリズムに従ってそのデータを学習する。その際、学習データを長期記憶に追加すべきかどうかや、既に長期記憶にあるデータで削除すべきものがあるかどうかをオンラインで判断している。また、長期記憶のif-thenルールは学習中にオンラインでクラスタリングされる。これらの機能により追加学習および推論結果の重複の回避を同時に行うことができる。入力されたif-thenルールの全てをただ長期記憶に蓄積していくのではこれらを実現することはできない。

2.1 学習フェーズ

学習データとして if-then ルールが入力されると最初に「リテラルの連言→リテラルの連言」の形をした複数の if-then ルールに次の方法で分解する。まず入力された if-then ルールの条件部と結論部をそれぞれ選言標準形に変形する。任意の命題論理式は選言標準形に変形することに注意されたい。そして変形した if-then ルールの条件部と結論部の選言肢をひとつずつ選びそれらを含意で結んだ if-then ルールを全ての組み合わせについて作る。すると、もし選言標準形に変形したとき条件部と結論部の選言肢の数がそれぞれ M 個、N 個であったとすると M×N 個の「リテラルの連言→リテラルの連言」の形の if-then ルールを得る。分解された if-then ルールは一時的に短期記憶に保存される。

分解されたif-thenルールを単純に長期記憶に保存するだけでは推論フェーズでの計算量の増加および推論結果の重複を招く。そこで提案手法は類似データの排除および長期記憶にあるif-thenルールのオンラインクラスタリングを以

下に従って行う。なお以下は分解されたif-thenルールについて個別に行う。本節ではこれ以後分解されたif-thenルールを単に学習データと呼ぶ。まず、本節の最後に述べる方法で学習データが長期記憶にある知識と同じクラスに属するか判定される。もし長期記憶の知識と同じクラスには属しないと判定されれば、学習データを新たに長期記憶に追加し、その学習データについての処理を終了する。もし既存の知識と同じクラスに属すると判定されたなら、学習データの類似知識が既に長期記憶にあると考えられるので、長期記憶への追加はせず次のように長期記憶を更新する。まず学習データ  $k_i$  と長期記憶  $k_m$  との距離  $d(k_i, k_m)$  を次式で計算する。

$$d(k, k') = \min_{\{k_1, \dots, k_M\} = \{1, 2, \dots, M\}} \sum_{i=1}^M \|P_i - P'_k\| + \min_{\{l_1, \dots, l_N\} = \{1, 2, \dots, N\}} \sum_{j=1}^N \|Q_j - Q'_k\| + \min_{\{k_1, \dots, k_M\} = \{1, 2, \dots, M\}} \sum_{i=1}^M \|\tilde{P}_i - \tilde{P}'_k\| + \min_{\{l_1, \dots, l_N\} = \{1, 2, \dots, N\}} \sum_{j=1}^N \|\tilde{Q}_j - \tilde{Q}'_k\|$$

ここで、学習データおよび長期記憶の知識をそれぞれ

$$\left( \bigwedge_{i=1}^M P_i \right) \wedge \left( \bigwedge_{j=1}^N \neg Q_j \right) \rightarrow \left( \bigwedge_{k=1}^{\tilde{M}} \tilde{P}_k \right) \wedge \left( \bigwedge_{l=1}^{\tilde{N}} \neg \tilde{Q}_l \right)$$

$$\left( \bigwedge_{i=1}^M P'_i \right) \wedge \left( \bigwedge_{j=1}^N \neg Q'_j \right) \rightarrow \left( \bigwedge_{k=1}^{\tilde{M}} \tilde{P}'_k \right) \wedge \left( \bigwedge_{l=1}^{\tilde{N}} \neg \tilde{Q}'_l \right)$$

とおいた。このとき学習データと最も距離の小さい長期記憶の知識(第1勝者)と2番目に小さい知識(第2勝者)の間にエッジが無ければ、それらと同じクラスとするためそれらの間に年齢1のエッジを生成する。もし既にエッジが存在していればそのエッジの年齢を1に戻す。第2勝者以外と第1勝者との間にエッジがあればそれらのエッジの年齢を1増やす。このとき  $\lambda_{edge}$  を越えたエッジは削除される。学習回数がパラメタ  $\lambda$  の整数倍であるならばエッジを一本しか持たない知識を削除する。

本節の最後に、入力された学習データが既存の知識と同じクラスに属するかを判断する方法を述べる。学習データが入力されると、第1勝者および第2勝者のそれぞれの類似度閾値  $s$  を次式で計算する。

$$s = \begin{cases} \max_{k \in N} d(k, k_w) & (if N \neq \emptyset) \\ \min_{k \in A} d(k, k_w) & (if N = \emptyset) \end{cases}$$

ここで  $N$  は第1(2)勝者とエッジで結ばれている知識の集合、 $A$  は長期記憶に存在する全ての知識の集合、 $k_w$  は第1(2)勝者の知識を表す。もしも学習データと第1勝者の距離が第1勝者の類似度閾値よりも小さく、かつ第2勝者についても同様のことが成り立てば、学習データは第1勝者および第2勝者と同じクラスであると判断する。第1勝者または第2勝者の類似度閾値よりも入力ベクトルまでの距離が離れていれば、入力ベクトルは既存のノードとは同じクラスではないと判断する。

2.2 推論フェーズ

推論フェーズではパターン情報に連言、選言、否定を付与した論理式がファクトとして入力される。ファクトが入力されると、学習フェーズで長期記憶に保存された if-then ルールと入力されたファクトを利用して推論を行い、入力したファクトを根とする OR 木が出力される。この木の各節はリテラルの連言を保持しており、これらを選言で結んで作られる複数の選言標準形を推論結果とみなすことができる。以下にアルゴリズムの詳細を述べる。

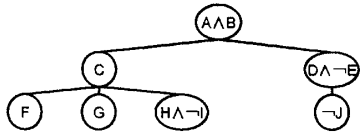


図2: 出力されるOR木の例

ファクトが入力されると入力されたファクトを根とし、それ以外の節はもたない木を生成する。次に、長期記憶に蓄えられているif-thenルールの条件部とファクトとの距離を計算し、閾値 $\delta$ 以下の距離になったif-thenルールの結論部のみを先の木に入力されたファクトの子ノードとして追加する。ただし、同一のクラスタから複数のif-thenルールを追加することを避けるため、閾値を下回るif-thenルールが同じクラスタに複数ある場合は距離の最も小さいif-thenルールのみが子ノードに追加される。もし、閾値以下の距離になるようなif-thenルールが長期記憶に存在しない場合は処理は終了する。その後、追加された子ノードを入力されたファクトと同様に扱い、次々に子ノードを生成していく。この手順は、入力されたファクトと長期記憶の断片的な知識を組み合わせる多段の推論を行っていることに相当する。例として、長期記憶として $A \wedge B \rightarrow C \vee (D \wedge \neg E)$ ,  $C' \rightarrow F \vee G \vee (H \wedge I)$ ,  $D' \wedge \neg E' \rightarrow \neg J$ を保持しているときに $A' \wedge B'$ がファクトとして入力されたときに出力される木を図2に示した。ここで $(A, A')$ ,  $(B, B')$ ,  $(C, C')$ ,  $(D, D')$ ,  $(E, E')$ は閾値に対して十分に距離が小さいパターン情報のペアである。

出力される木の各節にはデータとしてリテラルの連言が保持されているが、これらは単独では推論結果として正しくない。なぜならば、各節はもともと選言で結ばれたものを分解して学習されたものだからである。図2で言えば $A \wedge B \rightarrow C \vee (D \wedge \neg E)$ なのであって、 $A \wedge B \rightarrow C$ ではない。そこで、出力された木をOR木として扱い、節を選言で結んだ選言標準形が推論結果として正しいものであるとみなす。例えば図2の木が出力された場合は、推論結果が $C \vee (D \wedge \neg E)$ ,  $C \vee \neg J$ ,  $F \vee G \vee (H \wedge I) \vee (D \wedge \neg E)$ ,  $F \vee G \vee (H \wedge I) \vee \neg J$ であったとみなす。このように、推論結果は選言標準形をしているが、これは提案手法が可能性についての推論を行っていることと解釈することができる。例えば現在地に関する情報をファクトとして入力した結果、出力として「スーパーまたはコンビニエンスストアまたは薬局」に相当する結果を得たとしよう。これは現在地からどのような場所へ移動できるかという可能性を推論によって発見したと考えられる。

### 3. 実験

実画像を用いて環境の因果関係や可能性について追加学習させ、学習した知識を用いて推論を行わせた。学習時のパラメータは $\lambda_{edge} = 100$ ,  $\lambda = 50$ とした。用いた画像は図3に示したような $56 \times 46$ ピクセルの14種類の画像についてそれぞれアングルを変えて撮影した画像を各20枚ずつ合計280枚を用意した。学習させたif-thenルールは「 $A \rightarrow B$ ,  $B \rightarrow D \vee E$ ,  $E \rightarrow (C \wedge N) \vee F \vee M$ ,  $(C \wedge N) \rightarrow (G \wedge I) \vee (H \wedge J)$ ,  $(G \wedge I) \rightarrow K$ ,  $(H \wedge J) \rightarrow L$ 」である。これらをシンボルで書くならば「閉じたドア $\rightarrow$ 開いたドア, 開いたドア $\rightarrow$ 壁 $\vee$ 廊下, 廊下 $\rightarrow$  (研究室 $\wedge$ 表札)  $\vee$ エレベータ $\vee$ 階段, (研究室 $\wedge$ 表札)  $\rightarrow$  (机1 $\wedge$ 閉まった引出し1)  $\vee$  (机2 $\wedge$ 閉まった引出し2), (机1 $\wedge$ 閉まった引出し1)  $\rightarrow$ 開いた引出し1,

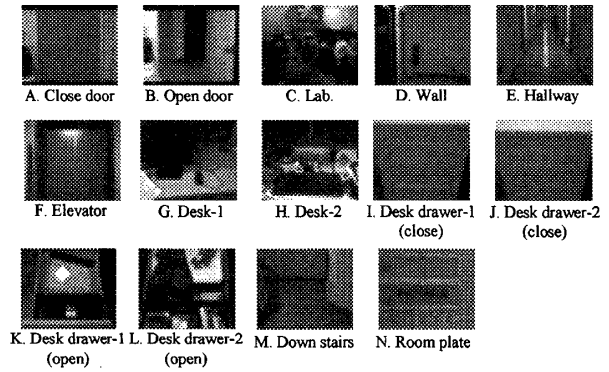


図3: 実験に用いた画像の例

(机2 $\wedge$ 閉まった引出し2)  $\rightarrow$ 開いた引出し2」である。学習データとして実際に入力されたデータは、各原子命題をあらわす20枚の画像をそれぞれランダムに1枚ピックアップしてできるif-thenルールであり、そのようにランダムに選択してできるif-thenルールの入力をそれぞれのif-thenルールについて4400回繰り返した。例えば $A \rightarrow B$ を学習させる際には、A, Bに対応する画像をそれぞれ20枚の画像から1枚ずつ選び、それをif-thenルールとして学習させることを繰り返した。

学習によって980個のif-thenルールが長期記憶に蓄積され、それらは10個のクラスタに分類された。クラスタに属するif-thenルールはそれぞれ「 $A \rightarrow B$ ,  $B \rightarrow D$ ,  $B \rightarrow E$ ,  $E \rightarrow (C \wedge N)$ ,  $E \rightarrow F$ ,  $E \rightarrow M$ ,  $(C \wedge N) \rightarrow (G \wedge I)$ ,  $(C \wedge N) \rightarrow (H \wedge J)$ ,  $(G \wedge I) \rightarrow K$ ,  $(H \wedge J) \rightarrow L$ 」に対応する知識だけであった。これはシンボルベースで表せば単一のルールとなるようなif-thenルールをうまくクラスタリングできていることを意味する。表1に各クラスタが保持するif-thenルールの数(クラスタのメンバー数)とデータ圧縮率を示した。データ圧縮率は「1 - (クラスタのメンバー数 $\div$ 与えた学習データの数)」で評価した。学習データとして合計で4000種類のif-thenルールが与えられたが、そのうちの75.5%にあたる3020個のif-thenルールは長期記憶には蓄積されなかったことがわかる。これは類似した知識を長期記憶に蓄積しないために導入したアルゴリズムがうまく働いていることを意味している。ここで、上記の学習手順がバッチ学習ではなく追加学習になっていることに注意されたい。上記の手順で正しく知識を学習できたことは、いったん学習が終了した後新たな学習データが追加されたとしても以前に学習した知識を忘却することなく新たな知識を学習することや、同じか極めて類似したデータの長期記憶への追加によるメモリの浪費を避けることが可能であることを示している。

学習後、ファクトとして図4の根の画像を入力して $\delta = 0.3$ として推論を行わせたところ、if-thenルールとして断片的に学習したデータを組み合わせると図4のOR木が推論結果として出力された。これは入力したファクトと学習させたif-thenルールから得られる結果として不足も重複もない必要十分なものであるといえる。この結果は、閉じたドアから階段、エレベータ、引出し等に到達可能であることと、それらの場所に至るまでの経路をシステムが推論によって見出すことができたことと解釈できる。引出しに関する推論の部分では提案手法が連言を扱うことができることが

表1: 各クラスタに保存された知識の数と圧縮率

クラスタに対応する知識	保存した if then ルールの数	圧縮率(%)
A→B	87	78.25
B→D	145	63.75
B→E	50	87.5
E→(C∧N)	64	84
E→F	31	92.25
E→M	146	63.5
(C∧N)→(G∧I)	84	79
(C∧N)→(H∧J)	111	72.25
(G∧I)→K	140	65
(H∧J)→L	122	69.5
合計	980	75.5

有効に働いている。もし連言を扱えないと、机の画像を知識として与えずに閉じた引出しの画像だけを学習することになり、異なる引き出しでも見た目は同じであるため正しく推論を行うことができない。推論結果に重複が無いのは if-then ルールをクラスタリングしながら学習できたためである。もしクラスタリング結果を無視して、閾値よりも小さい if-then ルールの結論部を全て節として追加するようにして推論を行わせたところ、3,804,840,166 個もの節が生成された。これらの節の持つ画像は全て図3のBに対応する20枚の画像に一致し、シンボルスで解釈するならば1つの節で表現すべき内容であった。この結果は if-then ルールをクラスタリングすることの有効性を示している。またノイズ耐性を確かめるため、図3のAにあたる閉じたドアを異なったアングルで撮影した画像20枚にガウシアンノイズを加えたものを同一のノイズレベルについて10枚ずつ用意し、それらの画像をファクトとして与えた。各ノイズレベルについて”正しく推論した回数÷推論を試みた回数”をプロットしたところ図5のような結果を得た。この結果はあるSN比を境界にして、それよりノイズの多い画像は未知画像だと判断されて推論が行われず、ノイズが少ない画像に対しては原画像のときと同様の正しい推論を行っていることを意味する。本手法は $\delta$ の設定で境界になるSN比を調整できる柔軟性を持つ。パターン推論機の先行研究である[3]はこのようなことを行えない。汎化性能の評価は今後の課題だがノイズ耐性の結果は提案手法が汎化性能を持っていることを示唆するものである。

## 5. むすび

実環境下で自律的に活動することが要求されるシステムにとって従来のシンボルベースの推論が不十分であることをふまえ、パターン情報ベースの推論機の提案を行った。提案手法はパターン情報を原子命題としそれらの原子命題に連言・選言・否定を付与した任意の形の if-then ルールを学習することができ、学習した知識を利用して多段の推論を行うことができる。これら推論機としての基本的な機能以外にも if-then ルールの追加学習、推論結果の重複の回避、汎化性能、ノイズ耐性を実現している。提案手法は if-then ルールをオンラインでクラスタリングしながら学習していくが、将来的にシステムの内部的なシンボル獲得につながる可能性もあると考えている。今後は提案手法の定量的な評価を行い、その後は研究をふたつの方向に発展させたい。

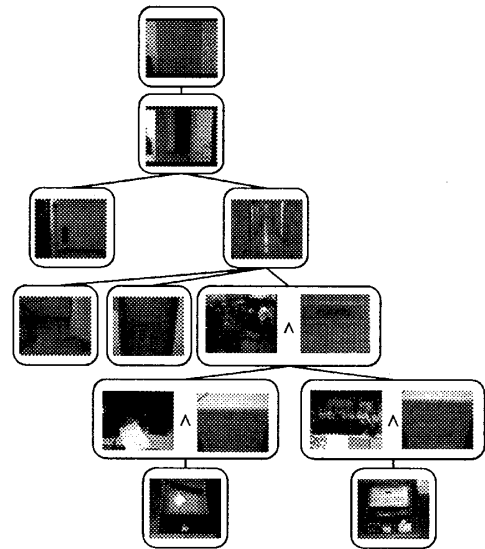


図4: 推論結果

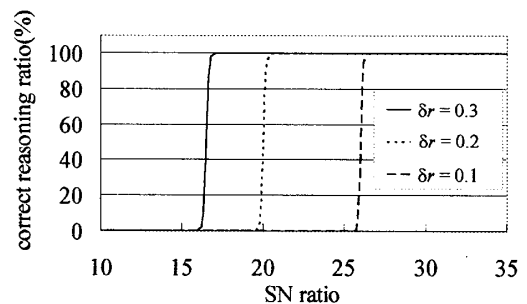


図5: ノイズの乗った画像からの推論

ひとつは提案手法を智能ロボットに適用するという方向である。これにより、環境から自律的にパターン情報ベースの if-then ルールを学習し、学習した if-then ルールを用いた推論によってタスクを解決する汎用的な智能ロボットが開発できるものと期待している。もうひとつはそのための基盤技術として本論文で提案したシステムを帰納推論、発想推論等が行えるように拡張するという方向である。それにより、より複雑なタスクへの対応が可能となる。

**謝辞** 本研究の実施にあたり NEDO 産業技術研究助成事業から支援を頂きました。記して感謝いたします。

## 参考文献

- [1] F. Shen and O. Hasegawa, "An incremental network for on-line unsupervised classification and topology learning," *Neural Networks*, Vol.19, No.1, 2006.
- [2] Sudo, A., Sato, A. and Hasegawa, O. "Associative Memory for Online Incremental Learning in Noisy Environments," *IJCNN'07*, 2007.(accepted)
- [3] 山根健, 蓮尾高志, 末光厚夫, 森田昌彦: 軌道アトラクタを用いたパターンベース推論, 電子情報通信学会論文誌(D), J90-D, 933-944, 2007.