

考察：「画像理解」

画像理解は人間（ロボット）の思考過程の重要なコアとなる処理です。思考はイメージを基本にして、それを中心手がかりに問題解決をしていく過程です。人間はイメージの変形操作を駆使して思考しているのです。ロボットの思考機構も当然イメージベースの操作を基本処理として実現されるでしょう。

そこでまずは、外界からの基本データとして画像が与えられたとして、それを知識として取り込むこととなりますので、この部分の機構を考えていきたいと思います。

画像を知識に取り込むには2つの大きな処理が必要になります。

(1) 画像を解析して、画像に表現されているオブジェクト群（人物とか乗り物とか）を3次元ボクセルとして把握する。

(2) 必要に応じて、オブジェクト間の関係や、オブジェクトの属性を解析して、意味記号表現を得る。

(1)、(2)によって、画像は理解されたこととなります。画像は写真などの2次元のものとしします。

1. 画像解析

(1)の処理は次のようなステップを踏んで行われるはずですが、無数の並列処理が行われるでしょう。

(1-1) エッジ検出

オブジェクトを切り出す手がかりとなる、画素値（RGB）の変化の激しいところ・・・エッジを検出していきます。これは画像全体を幾つかに分割して平行して処理をしていくこととなります。

(1-2) ライン検出

エッジラインを検出していきます。これも画像全体を幾つかに分割して平行して処理をしていきます。

(1-3) 閉曲線検出

ラインが閉曲線をつくっていたらその曲線図形を検出します。これは曖昧性のある処理ですので、(1-2)の結果から無数の可能性を見だしていくのですが、確信度をもって画像全体との関係性を評価しながら処理していきます（弛緩法）。

(1-4) 幾何図形の当てはめ

(1-3)の閉曲線が円なのか、三角なのか、四角なのか等々と幾何学的性質を解析していきます。これも確信度をもって、画像全体との関連性を評価しながら処理していきます（弛緩法）。

(1-5) オブジェクト推定

人物が自動車かなど、具体的な知識ベースのデータと照合して、幾何学図形を評価していきます。部分的な特徴から全体のオブジェクトの同定をしていきます。これも確信度によって、画像全体との関連性を評価しながら処理を行っていきます（弛緩法）。

画像図形の解析結果はマップシステムで管理していきます。それだと並行処理にも適していますし、データの組み合わせもコンパクトに管理できるからです。

また、細かいパターン認識は、目的に特化した狭い範囲の処理をするコンポーネントプロセスを沢山作っていき、それらを統合して一般の画像を認識していくのが現実的です。コンポーネントもアルゴリズムを共通にして、違いは制御データとして持つようにしていきたいと思います。

こうした設計方針にしますと、オントロジーとの整合性も出てきて見通しがよくなります。オントロジーな画像・・・プリミティブ画像を定義して、そのテンプレートマッチングを基本とすればいいのです。複雑な図形はプリミティブ画像の組み合わせとして定義できます。それは無論、曖昧性管理、弛緩法を前提として実現していくのですが。確信度はテンプレートが完全一致すれば一番高くします。まったく一致するところがないと確信度は0になります。そうした総合的なプロセスの実行結果として画像理解はなるという考え方です。

プリミティブはイメージデータでテンプレートを構成します。プリミティブ画像解析プロセスは入力画像の範囲をパラメータに貰って、テンプレートマッチングしていくのです。組み合わせパターン認識プロセスは画像解析をプリミティブ画像解析プロセスに落とし込んで結果を評価します。落とし込む条件とサブプロセス配置は外部コマンドデータによります。そんなプロセスがあるのです。

条件を振らしながら、ゆるぎを起こして、弛緩法によって最適な認識結果を得ていきます。

コマンドの形式は次の通りです。

【パターン認識コマンド】

(1) オブジェクト定義

(プリミティブ識別子;プリミティブ図形,条件属性、厳密さ;・・・)で定義する。
プリミティブ図形はイメージデータの識別子である。

[例]人体の定義

(body;+ellipse_ball,+vertical_long,0)・・・胴体を楕円で定義

(head;+ball,0)・・・頭部をボールで定義

ballとかellipse_ballはプリミティブパターン識別子で、実際に動かすパターン認識プロセスの識別子だったり、テンプレートパターンの識別子だったりします。

(2) オブジェクト関係定義

(視点プリミティブ識別子、対象プリミティブ識別子;関係プリミティブ、条件属性、厳密さ;・・・)

[例]人体の定義

(head,body;+connect,+vertical,0)・・・頭が胴体に接続していることを示す。

connect は接続を示すプリミティブ関係です。

【パターン認識コンポーネントの組み合わせ】

パターン認識プロセス群はコンポーネントウェアとして実現します。だから基本的にデータフロー計算機のようなアーキテクチャで実行されます。一般にパターンはパターン認識コマンドで定義された文法表現で、サブコンポーネントに処理が任せられ、プリミティブパターンマッチングのコンポーネントで最終的にパターンが解析されます。パターンの認識は最上部のコンポーネントが組み合わせを弛緩法で評価して実現します。

【テンプレートマッチング】

イメージデータの解析の基本は、知識の中のイメージとのテンプレートマッチングであります。線画像の一致度を求めて行います。一致度は対応する輪郭線を適当に区切って、点对応させて、そのユークリッド距離の和が最小となるような値です。イメージの回転角とか変位距離、拡大縮小によって、対応線画像の一致度は異なってきますから、適当な条件を求めて一致度を計算する事になります。線画像の回帰直線を一致させることで最適な一致度を求めることができるでしょう。

システムは基本的にコンポーネントウェアとして実現します。基本的には、次の技術で実現するコンポーネントウェアです。

(1) データフロー計算機

データ駆動型のプロセス管理にするには、変更柔軟に対応できるフレームワークでシステムを作成するのが有利でしょう。

(2) 多元多重並行処理

一つのデータの塊を、部分部分に切り出して、解析を行い、部分的なデータを元のデータに付加したり、全体の中で部分を評価したりする処理になります。部分のプロセス群をオーケストレーションするにはそれを可能とするデータ構造にしなくてはなりません。基本的には黑板システムにしますが、黑板システム内を識別子のついたデータの塊でうめるようにすることになります。XML - DOMですね。そんな技術が良いと思います。

(3) 組織化パターン認識

パターン認識を無数のプリミティブパターン認識プロセスで実現し、プリミティブ以外のコンポーネントは、全体の中の部分、部分から全体への推測といった弛緩法のための評価を実施していきます。

2. オブジェクト間関係解析

オブジェクト認識の結果を利用して、指定の2つ、3つのオブジェクト間の関係を解析します。関係としては、次のものがあります。

(1) 格

位置関係、作用関係（英語の前置詞に相当）を解析していきます。それには画像解析で得た、オブジェクトの認識結果が前提となります。

知識は視点を中心とした属性のマップで表現します。マップは、ファジー集合で記号に対応させます。

(2) 属性

オブジェクトの属性（色、大きさなどのメトリック種類とその値）を解析します。

知識はメトリックを属性マップと記号対です。ファジー集合になっているはずですが。

(3) 動詞

動作や操作の認識を行います。それはあるいみ画像解析結果への解釈の問題となり、知識ベースの内容とのマッチングで得ていく認識結果です。

意味フレームの時系列によって、事象とテンプレートマッチングし、その時系列の識別子が動詞になります。

(4) 名詞

オブジェクトを同定します。オブジェクトとしての浅い認識でなくて、人物名とか、何処の土地であるかなどの具体的な情報を知識ベースから得る処理です。

属性セットが一致したら、そのセットの識別子が名詞です。

オブジェクト間関係は、基本的に、属性セットが満たされているかどうかを解析し、対応するカテゴリ記号（識別子）を得る過程です。

3. パターン認識のプランニング

画像認識を弛緩法で実現すると語りました。しかし、普通は無数の画素線分があって、それらの組み合わせを総当たりすることは現実的に無理です。組み合わせ爆発が起きてしまいます。そこで、パターン認識には画素線分を適当に選択して効率よく処理していく方法をとらねばならないことが分かります。それは、塊を先ず検出して、その塊の近傍を解析しつつ次の塊にプロセスを移していくという方略を採るべきということです。塊は普通一つか2つです（注視がなされているから注視点を画像として切り出すことになります）。それで、人の顔だと分かれば、目とか、鼻、口など配置は絞れますから、重点的に画素線分を絞り込めます。そんな、パターン認識もプランニングしながら部分解析を行っていくとよいでしょう。それは人間の脳でもやっていることです。

そして、パターン同定には次の技術がコアとなると考えているのですがどうでしょうか。

(1) 弛緩法

全体と部分のパターンの整合性を漸近的にとり、最後に意味ある認識結果を得る技術。

(2) プライミング

永続化データをアクセスする時間を節約するために、現在の処理にもっとも関係深いものだけをインコアメモリに持つ技術。メモリには永続化データでなくて、そのデータの識別子だけを持つようにすればメモリ消費を少なくできます。そんな工夫。

(3) 漸詳細化

パターンマッチングをいきなり詳細化して行うのではなく、なにか大枠の特徴で絞り込んで認識候補を選択していき、選択して残ったもののなかで更に詳細にパターンマッチングして最終的な候補を得ていく技術。

(4) 文脈管理

全体の画像の傾向を、個々のパターンマッチングとは独立して解析していき、個々のパターンの最終認識結果の選択に活かしていく技術。弛緩法の援助、プライミングの制御、漸詳細化のプランニングの制御に用いていく。

オブジェクト認識は画素線分の配置をメインのキーデータとしてパターンマッチングして行くはず。そのとき、どの画素線分が重要なのか、ゴミなのかはあらかじめは分かりません。幾つかの候補の中から選択して組み合わせて行かねばなりません。とするならば、画素線分の選択とその属性によって、知識ベース内のパターンから大枠の候補を先ず得ていく事にしなくてはなりません。各画素線分とその属性から幾つかの候補群ができます。基本的に and 処理で最も沢山の候補に入っているパターンを選択すればいいのですが、それでも文脈とか弛緩法とかで少しずつ絞っていくのが優れています。もちろんプライミングして、候補の枝刈りとアクセスの高速化も果たします。最後に詳細な特性情報を基に認識パターンを確定していきます。

今は、こんなことでコーディングを開始しようと思っていますが、もっと現実的で優れた方法があるかもしれません。

おわり