

コンピュータビジョンにおける
パターンマッチングの数理的手法に関する研究

平成2年10月

梅山 伸二

コンピュータビジョンにおける パターンマッチングの数理的手法に関する研究

内容梗概

コンピュータビジョンとは、目で外界を見てその状況を認識・理解する人間の視覚情報処理機能を、機械を用いて実現させることを目的とした研究分野である。本論文は、現在精力的に研究が進められつつあるコンピュータビジョン研究における基礎的な課題の一つである、パターンマッチング手法についての研究をまとめたものである。計算機内に外界を表現する場合、大きく分類して、対象シーンやモデルを、その構成要素とそれら構成要素間の関係を用いて表現する関係・構造記述と、二次元あるいは三次元空間内の幾何的プリミティブ（点・線分・領域など）の分布パターンとして表現する空間記述の二つの場合があるが、本論文では、前者の場合の例として重みつきグラフ間の最適対応問題を、後者の場合の例として二次元あるいは三次元空間内の点パターン間のマッチング問題を題材として取り上げ、それぞれその問題の持つ固有の特徴・性質を十分に考慮したマッチング手法を考察、提案する。その概要は以下の通りである。

第1章の序論では、本研究の意義、目的などについてのべる。

第2章では、関係・構造記述間の一般的なパターンマッチング問題として重みつきグラフ間の最適対応を求める問題を取りあげ、与えられたグラフの隣接行列（無向グラフの場合）あるいは隣接行列から導かれるあるエルミート行列（有向グラフの場合）の固有値展開を利用して、求める対応を解析的に決定する手法を提案し、計算機実験によりその有効性を示す。また、それに関連して、グラフの自己同型写像に関する幾つかの結果を述べる。

第3章では、空間記述されたパターン間のマッチング問題として、二次元あるいは三次元空間内の点パターン間のマッチング問題を取りあげ、空間的パターンの持つ最も基本的な情報である位置（座標）情報のみを用いたマッチング手法を提案し、計算機実験によりその有効性を検討する。提案する手法は木探索アルゴリズムに基づいており、柔軟で、相似変換（回転、平行移動、スケール変換）により関係づけられた任意の次元の点パターンに対して適用可能である。

次に第4章では、第3章で述べた点パターン間のマッチング手法を拡張し、平面上のパターンに限られるが、パラメータ化された点パターン間のマッチング手法を考察する。

さらに第5章では、第3章、第4章で提案した点パターンマッチングアルゴリズムの応用として、部分形状による認識、位置決め問題を取りあげ、認識対象が通常のモデルで表現可能な場合、および第4章で述べたようなパラメータ化されたモデルで表現される場合それぞれについて、認識、位置決め実験を行なう。

最後に第6章、結論では、本研究で得られた結果を要約する。

目次

| | | |
|-------|-------------------------|----|
| 1 | 序論 | 1 |
| 2 | 構造記述されたパターン間のマッチング | 6 |
| | — 重みつきグラフ間の最適対応問題 | |
| 2.1 | はじめに | 6 |
| 2.2 | 問題設定 | 8 |
| 2.2.1 | 重みつきグラフとその隣接行列 | 8 |
| 2.2.2 | 重みつきグラフ間の最適対応問題 | 9 |
| 2.3 | 重みつき有向グラフ間の最適対応 | 11 |
| 2.3.1 | 準最適対応の決定 | 13 |
| 2.3.2 | 準最適対応の決定法 I | 15 |
| 2.3.3 | 準最適対応の決定法 II | 20 |
| 2.4 | 重みつき無向グラフ間の最適対応 | 23 |
| 2.5 | 最適対応手法から導かれるグラフのいくつかの性質 | 25 |
| 2.6 | 計算機実験と考察 | 27 |
| 2.6.1 | 実験方法 | 28 |
| 2.6.2 | 結果および考察 | 30 |
| 2.7 | むすび | 32 |
| 3 | 空間記述されたパターン間のマッチング I | 38 |
| | — 空間内の点パターン間のマッチング | |
| 3.1 | はじめに | 38 |

| | | |
|-------|-----------------------------|-----|
| 3.2 | 点パターンマッチング問題 | 40 |
| 3.3 | 点パターンマッチングアルゴリズム | 42 |
| 3.3.1 | 探索木と深さ優先探索 | 42 |
| 3.3.2 | 既発見のマッチングの大きさによる枝刈り | 44 |
| 3.3.3 | スケールの適格範囲による枝刈り | 45 |
| 3.3.4 | 探索アルゴリズム | 56 |
| 3.3.5 | 実行例 | 59 |
| 3.4 | さまざまな条件、制約下でのマッチング | 63 |
| 3.4.1 | 鏡像パターン | 63 |
| 3.4.2 | 変換パラメータに対する制約 | 64 |
| 3.4.3 | 実行例 | 65 |
| 3.5 | 点パターン間の全体マッチング問題 | 70 |
| 3.5.1 | 重心点一致戦略 | 70 |
| 3.5.2 | スケールの適格範囲による枝刈り | 71 |
| 3.5.3 | 適格範囲に対する初期制約 | 74 |
| 3.6 | 計算機実験による点パターンマッチングアルゴリズムの評価 | 77 |
| 3.6.1 | 実験方法 | 77 |
| 3.6.2 | 結果および考察 | 80 |
| 3.7 | むすび | 86 |
| 4 | 空間記述されたパターン間のマッチング II | 88 |
| | — パラメータ化された点パターン間のマッチング | |
| 4.1 | はじめに | 88 |
| 4.2 | パラメータ化された点パターン間のマッチング | 90 |
| 4.3 | 探索木の枝刈り | 93 |
| 4.3.1 | $EL_o(\Phi)$ の近似 | 93 |
| 4.3.2 | $EL_c(\Phi)$ の近似 | 98 |
| 4.4 | 実行例 | 98 |
| 4.5 | むすび | 100 |

| | | |
|-------|---------------------------|-----|
| 5 | 点パターンマッチングの形状認識への応用 | 102 |
| 5.1 | はじめに | 102 |
| 5.2 | 形状認識に特有の枝刈り法 | 104 |
| 5.2.1 | 対応順序の無矛盾性を用いた枝刈り | 104 |
| 5.2.2 | 頂点の内角による枝刈り | 104 |
| 5.3 | 実験と結果 | 106 |
| 5.3.1 | 点パターンマッチングアルゴリズムによる部分形状認識 | 106 |
| 5.3.2 | パラメータ化されたモデルによる形状認識 | 113 |
| 5.4 | むすび | 120 |
| 6 | 結論 | 121 |
| | 附録 | 124 |
| | 参考文献 | 125 |
| A | 定理 2.1 の証明 | 133 |
| B | 定理 3.2 の証明 | 137 |

第 1 章

序論

コンピュータビジョンとは、目で外界を見てその状況を認識・理解する人間の視覚情報処理機能を、機械を用いて実現させること、つまり、カメラを通じて得られた画像から、外界の物理的対象物に対する明白で意味のある記述を計算機を通して構成すること、を目的とした研究分野であり、その内容は、視覚情報入力技術、特徴抽出技術など多岐にわたっている。ボトムアップなコンピュータビジョンプロセスの非常に簡単化されたパラダイムとしては、次のような四段階のプロセスが挙げられ (図 1.1) [51, 54]、このようなプロセスのそれぞれの段階について、現在、非常に精力的な研究が進められつつある [6, 31, 54]。

1. 視覚情報入力： TVカメラなどを用いて対象シーンを視覚情報 (入力画像) として計算機に入力する。
2. 特徴抽出： 入力画像から線や領域といった画像特徴を抽出し、画像の持つ情報を構造的に記述するための要素とする。
3. 記述作成： 局所的な特徴の関係を明らかにし、対象シーンの記述を作成する。
4. パターンマッチング： 以上の処理で求めた特徴と、あらかじめ知識として蓄えられている認識対象のモデルとをマッチングし、シーンに含まれる具体的な対象物を認識する。また、それらの特徴や位置などをシーンの記述として記録する。

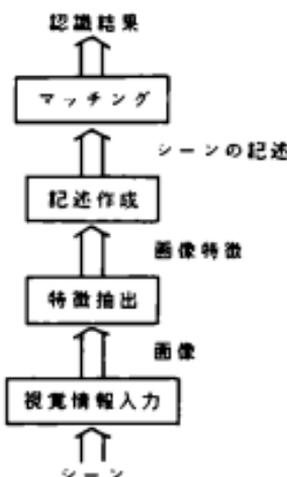


図 1.1: ボトムアップなコンピュータビジョンプロセス

例えば、コンピュータビジョン研究における重要な課題の一つとして、物体の二次元形状（輪郭）からその物体を認識する、あるいはその物体のシーン内での位置を決定する問題を考えてみる。この場合様々な手法が考えられるであろうが、代表的な手法としては、まずTVカメラから入力された対象シーンやモデルとなる物体の画像を二値化し、その輪郭の多角形近似を求めることにより、画像特徴（外形特徴）を抽出する。次に、そのようにして得られた多角形近似の、例えば外形に沿った頂点列をシーンおよびモデルの記述として採用し、最後に何らかのマッチング手法を用いてそれら記述間のマッチングを行なうことにより、モデルとなる物体のシーン内での位置を決定することになる。実際の認識システムにおいては、パターンマッチングの後、得られたマッチングが本当に意味のある解となっているかを検証する手続きなども更に必要であろう。

本論文は、このように広範な研究が進められつつあるコンピュータビジョン研究のうち、上記の四段階の最終段、パターンマッチング手法に関して筆者が従来行ってきた研究をまとめたものである。

コンピュータビジョンにおけるパターンマッチング問題においては、処理の対象に対する様々な知識を用いてマッチングを行なうことが可能であり、またそれらの知識を利用することにより、実行の大幅な効率化が可能となる。このため、計算速度が要求される実際のシステムのインプリメントなどにおいては、対象に対する知識に依存したマッチング手法を採用することが少なくない。また、マッチング問題の多くは組合せの問題となり、その計算量が非常に大きなものになりがちなため、様々なヒューリスティックを用いて、計算量を低減させようとすることも多い。しかしこのような場合、その手法は問題の対象に依存した、直観的でその場限りのものになりがちであり、また、最終的にどのような解が得られるのか、得られる解の意味が曖昧なものになるという弊害も生じる。この意味から、今後のコンピュータビジョン研究を進めていく上では、このように個々の問題に依存した手法ではなく、様々な問題に共通する基盤的なマッチング手法を確立することが重要であろうと思われる。本論文は、コンピュータビジョンにおける様々なパターンマッチング問題に共通すると思われる本質的な問題を取り上げ、それらに対する一般的な解法を数理的な立場から考察することを目的とするものである。このため、上で述べたコンピュータビジョンのプロセスの内、最初の三段階までは、既に何らかの方法により達成されていると仮定する。

対象シーンやモデルの記述・表現は、対象をその構成要素とそれら構成要素間の関係を用いて記述する関係・構造記述（例えばグラフ表現）と、対象を二次元あるいは三次元空間内の幾何的プリミティブ（点・線分・領域など）の分布パターンとして表現する空間記述の、大きく二種に分類可能であるが[6]、これらに対応して、パターンマッチングの手法も二種に分類される。つまり、前者のように構造記述されたパターン間のマッチングを行なう場合には、グラフ理論的なマッチング手法が必要であろうし、後者のような空間的パターン間のマッチングを行なう際には、二次元あるいは三次元空間内の物体の持つ幾何学的制約を十分に利用したマッチング手法を開発することが重要であると思われる。本論文では、このような観点から、構造記述間のマッチングおよび空間的パターン間のマッチングの両者について、それぞれその問題の持つ固有の特徴・性質を十分に利用したマッチング手法を提案する。

本論文の構成は以下のようになっている。

まず第2章では、関係・構造記述間の一般的なパターンマッチング問題の一つとして、重みつきグラフ間の最適対応問題を取りあげる。ここで言う重みつきグラフとは、通常のグラフの各アークに対してそれぞれ重みが与えられたグラフであり、それらの間の最適対応問題とは、与えられたある評価関数を最小にするようなグラフノード間の対応関係を決定する問題である。通常このような問題に対しては、木探索法 [73, 57] や緩和法 [35, 36] を用いる場合が多いが、本論文では与えられたグラフの隣接行列（無向グラフの場合）あるいは隣接行列から導かれるあるエルミート行列（有向グラフの場合）の固有値展開を利用して、求める対応を解析的に決定する手法を提案する。この手法により、対応を行なう二つのグラフが十分に同型に近いような場合には、それらの間の準最適な対応を高速に求めることが可能である。

次に第3章では、空間記述されたパターン間の代表的なマッチング問題として、二次元あるいは三次元空間内の点集合間のマッチング問題を取りあげる。これは、空間内に二つの点集合が与えられた場合に、一方の点集合を相似変換を用いてもう一方の点集合上に重ね合わせ、できるだけ多くの点に対応する（重なり合う）ようにする問題である。コンピュータビジョンに対する一般的なアプローチの一つとして、各画像から特徴点を抽出し、それら特徴点間のマッチングを、特徴点の持つ位置・意味ラベル・順序などさまざまな特性を手掛りとして行なう方法が考えられるが、本論文で取り扱う点集合間のマッチング問題は、このような問題をより抽象化し、空間的パターンの持つ最も基本的な情報である位置（座標）情報のみを用いることにより、そのようなマッチングがどこまで可能であるかを考察したものである。空間内の点集合はしばしば点パターンと呼ばれるため、この問題は点パターンマッチング問題と呼ばれる。

点パターンマッチング問題に対する従来の手法の中には、緩和法のように最終的に得られる解がどのようなものとなるか、直観的に判りづらいものも少なくないが、ここではこれを避けるため、まずマッチングの許容誤差を定義するものとして δ 制限という概念を導入し、二つの点パターンが与えられた時に、この制限を満足する大きさ最大（対応点数が最大）のマッチングすべてを効率的に生成するアルゴリズムを提案する。

さらに次の章では、空間記述間のパターンマッチングのもう一つの例として、パラメータ化されたパターンを用いたマッチング問題を考える。例えば“はさみ”を考

えてみよう。はさみは支点を中心にして2つの刃が互いに交差し回転する。このような対象をその交差角に独立に、また交差角それぞれについて全てのモデルを準備するようなことなく認識しようとするれば、対象のモデルとして交差角をパラメータとして持つようなモデルを用意し、そのようなモデルとのマッチングを考える必要がある。また、例えば縦横のプロポーショナルなどの多様性も含めた、多種多様な重を一つのモデルを用いて表現しようとする場合なども同様である。このようにパラメータ化されたモデルを用いることにより巧く表現可能な対象は、現実の世界において決して少なくない。第4章では、このようなパラメータ化された対象をモデル化するために、点パターンの意味を拡張し、パラメータ化された点パターンの概念を導入する。そして、そのような点パターン間のマッチング手法としてパラメータ化された点パターン間のマッチングアルゴリズムを提案する。パラメータ化された点パターンは非常に柔軟な概念であり、上で述べたようなパラメータ化された対象のモデルを、非常に広い範囲でカバー、表現することが可能である。

そして、第5章では、第3章、第4章で提案した点パターンマッチングアルゴリズムの応用として、部分形状による認識、位置決め問題をとりあげる。認識対象が通常の(単独の)モデルで表現可能な場合、および第4章で述べたようなパラメータ化されたモデルで表現される場合それぞれについて、第3章および第4章の点パターンマッチングアルゴリズムを適用し、認識、位置決め実験を行なう。認識対象のシルエットの多角形近似の各頂点を外形の特徴点として採用し、そのような点の集合を点パターンとしてアルゴリズムを適用する。

最後に第6章では、結論として本研究で得られた結果を要約し、また将来に対する課題について述べる。

第 2 章

構造記述されたパターン間のマッチング

— 重みつきグラフ間の最適対応問題

2.1 はじめに

構造記述 (structural description)、つまりある対象物を、その構成要素と各構成要素の属性、およびそれら構成要素間の関係を用いて記述したものは、実世界における構造を持った対象を表現する上での一つの有力な手段である。例えば、“人間”という対象は、顔や胸のようなその構成要素、および大きさ・形・色などのようなその属性、あるいは“顔は体の上にある”といったそれら構成要素間の関係を用いて表現することが可能であり、また化学構造式なども、原子とそれらの間の化学結合で決定される一種の構造記述と考えることができる。このように構造記述において、各属性や関係の持つ値は、長さや重さのように量的なもの、あるいは形や色のように質的なもののいずれでもありえる。

さて、このような構造記述を取り扱う際には、例えば、与えられたデータからそれらを表現する構造記述をいかにして構成するか、あるいはいくつかの構造記述が与えられたとき、それらの分類をいかにして行うかといった、さまざまな種類の問題が考えられるが、そのような問題の中でも最も基本的でかつ興味ある問題として、構造記述間のマッチングの問題が挙げられる。つまり、二つの構造記述が与えられたとき、

それぞれの構造記述における属性、あるいは関係が最も良く一致するような構成要素間の対応関係を見つける問題である。

二つの構造記述間の最適マッチングの問題は、さまざまな分野における基本的問題であり、特に構造的パターン認識 (structural pattern recognition) や画像理解 (image understanding) の領域においては数多くの研究が行われてきている。例えば You [73] は分枝限定法 (branch-and-bound) を用いた木探索的手法によって構造記述の一種である重みつきグラフ間の最適対応問題を解いており、Tsai and Fu [57, 58]、Shapiro and Haralick [52]、Ghahraman, Wong, and Au [22, 23] らもまた、記号のあるいは数値的属性を持ったグラフ間の対応決定を木探索により行なう手法を提案している。一方 Kitchen [35, 36] は、構造記述の属性・関係が質的な場合 [35]、あるいは量的な場合 [36] それぞれについて、二つの構造記述間の最適対応決定の問題を緩和法 (relaxation) を用いて解くことを試みている。また、コンピュータビジョンの分野でよく用いられる Hough 変換の発想を利用した解法なども提案されている [34]。しかし、木探索のように組合せ的な手法はいわゆる“組合せ的爆発”を生じやすく、構成要素数の大きな構造記述間のマッチングは困難であり、一方、緩和法は、計算量的には有利であるかもしれないが本質的に局所的な最適化手法であり、また適合係数等の定め方が結果に大きな影響を及ぼすため、この係数をどのように定めるかが大きな問題となる。

構造記述は、一般的には属性付きの (ハイパー) グラフによって表現することが可能であり、重みつきグラフ、つまり、グラフの各アークにそれぞれ重みが与えられたグラフは、量的構造記述の一つの簡単な例となる。本章では、ノード数の等しい二つの重みつきグラフ間の最適対応問題を取り扱い、木探索法や緩和法ではなく、解析的な手法を用いることにより、この問題を近似的に解く手法を提案する [61, 62, 64, 67]。提案する手法は、与えられたグラフの隣接行列 (adjacency matrix、無向グラフの場合) あるいは隣接行列から導かれるあるエルミート行列 (有向グラフの場合) の固有値展開を利用するものであり、対応を行なう二つのグラフが十分に同型に近いような場合には、準最適な対応を非常に高速に求めることが可能である。提案する手法は木探索法のような“組合せ的爆発”を伴わず、また緩和法のように局所的な手法ではなく、大局的な最適化手法となっている。

グラフの何らかの取り扱いに際して、グラフの隣接行列の固有値（スペクトル）を利用する方法は広く行なわれているが [29, 59, 56]、このような場合には、通常、スペクトルのみが注目され、それと同時に定められる固有ベクトルに対しては、あまり注意が払われないことが多かった。しかし行列の持つ情報は、固有値ばかりでなく、固有ベクトルに対しても当然受継がれているはずであり、この意味で、固有ベクトルの情報を用いてグラフの持つ性質や、グラフに対する何らかの操作を実現することが可能な場合もあると考えられる。本章で示すマッチング手法は、このような立場から、グラフの隣接行列（あるいは隣接行列から導かれるあるエルミート行列）の固有ベクトルの持つ情報を十分に活用しようとするものである。

次の 2.2 節では重みつきグラフ間の最適対応問題を定式化し、続く 2.3 節、2.4 節においてそれぞれグラフが有向および無向の場合についてその近似解法を与える。2.5 節では、最適対応手法から簡単に導かれるグラフのいくつかの性質について触れ、最後に 2.6 節において分枝限定法など他の幾つかの最適対応手法との比較実験を行う。

2.2 問題設定

2.2.1 重みつきグラフとその隣接行列

重みつきグラフ G を順序対 (V, w) で定義する。ここで V はグラフのノードの集合であり、 w はグラフの各ノード対、つまりアーク (v_i, v_j) ($v_i \in V, v_j \in V, v_i \neq v_j$) に対してある値 $w(v_i, v_j)$ を与える重みづけ関数である。グラフの大きさを $|V| = n$ とし、グラフの各ノードにはそれぞれを区別するため v_1, v_2, \dots, v_n というラベルが付けられているものとする。重みつきグラフ $G = (V, w)$ について、その重みづけ関数 w が対称であるとき、つまりすべての $v_i, v_j, v_i \neq v_j$ について $w(v_i, v_j) = w(v_j, v_i)$ が成立するとき、そのグラフを無向グラフと呼び、逆に w が対称でない場合には有向グラフと呼ぶ。

明らかに、通常のグラフ理論で扱われるグラフは w が二値の場合の重みつきグラフである。また、 w が 0 から 1 の間の値をとるとき、そのようなグラフはしばしばファジーグラフと呼ばれるが [15]、ここでは重みづけ関数の値域に対してそのような制限は与えず、どのような値もとることができるものとする。

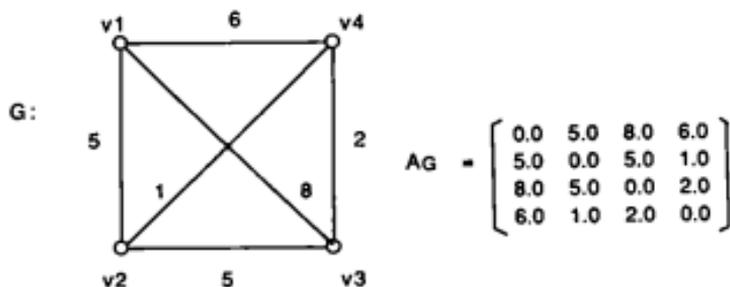


図 2.1: 重みつき無向グラフ G とその隣接行列 A_G

次に、重みつきグラフ $G = (V, w)$ の隣接行列 A_G を $n \times n$ 行列として次のように定義する。

$$A_G = [a_{ij}] \begin{cases} a_{ij} = w(v_i, v_j) & \text{if } i \neq j \\ a_{ii} = 0 & \text{otherwise} \end{cases} \quad (2.1)$$

G が無向グラフのとき A_G は対称行列となる。図 2.1 に 4 個のノードを持つ重みつき無向グラフ G およびその隣接行列 A_G を示す。

2.2.2 重みつきグラフ間の最適対応問題

前節の定義をもとにしてこの節では重みつきグラフ間の最適対応問題に対する定式化を与える。

$G = (V_G, w_G)$ および $H = (V_H, w_H)$ をそれぞれ n ノードを持つ重みつきグラフであるとする。このとき、重みつきグラフ間の最適対応問題を G と H 間のある意味での“差”を最小にするような G と H のノード間の一対一対応関係 ϕ を見つける問題であると定義する。この場合、どのような“差”をグラフ間に定義するかが問題となり、この点から“差”が距離の定義を満足するための条件なども研究がおこなわれているが [53, 63]、解析、計算の容易さから、本論文ではグラフ間の差としてアークに付けられた重みの二乗誤差を評価関数として採用する。すなわち $V_G = \{v_1, v_2, \dots, v_n\}$ 、 $V_H = \{v'_1, v'_2, \dots, v'_n\}$ をそれぞれ G 、 H のノードの集合として、

ある一対一対応 ϕ に対する G 、 H 間の差 $J(\phi)$ を次式で定義する。

$$J(\phi) = \sum_{i=1}^n \sum_{j=1}^n (w_G(v_i, v_j) - w_H(\phi(v_i), \phi(v_j)))^2 \quad (2.2)$$

さらに A_G 、 A_H をそれぞれ重みつきグラフ G 、 H の隣接行列であるとすると、上式の $J(\phi)$ は順列行列を用いて次のように書き直すことができる。

$$J(P) = \|PA_G P^T - A_H\|^2 \quad (2.3)$$

ここで P は一対一ノード対応 ϕ を表現する順列行列、 $\|\cdot\|$ は行列のユークリッドノルム ($\|A\| = (\sum_{i=1}^n \sum_{j=1}^n |a_{ij}|^2)^{1/2}$) であり、“ T ” は転置を示す。つまり重みつきグラフ間の最適対応問題は (2.3) 式の評価関数 $J(P)$ を最小にするような順列行列 P を見つける問題となる。

通常の (二値) グラフにおいては、二つのグラフのノード集合間の一対一対応でノード間の隣接関係を保存するようなものが存在する場合、それら二つのグラフは同型であるといわれる。同様に重みつきグラフにおいても、(2.2) 式の $J(\phi) = 0$ とするようなノード間の一対一対応関係 ϕ が存在するとき、それらのグラフは同型であると呼ぶことにする。このとき、(2.3) 式から明らかに次式が成立するような P が存在する。

$$PA_G P^T = A_H \quad (2.4)$$

通常の (二値) グラフは重みつきグラフの一種であるから、重みつきグラフ間の最適対応問題は、グラフ理論、アルゴリズム論などにおいて盛んに研究されているグラフの同型判別問題 [50, 21, 60] をその中に含むことになる。

重みつきグラフ間の最適対応問題は組合せの問題であり、一般にこのような組合せの問題に対して厳密な解を求めることは計算時間の上から容易ではない [20]。このため、この場合には準最適解を効率的に求める手法を開発することが実際問題として重要となる。本章の目的は重みつきグラフ間の最適対応問題に対する準最適解を与える高速な手法を提案すること、つまり最適値ではないが最適値に非常に近いような評価関数の値を与える順列行列を効率的に求める手法を提案することである。

2.3 重みつき有向グラフ間の最適対応

この節ではまず重みつき有向グラフ間の最適対応問題について考察する。重みつき無向グラフの最適対応問題については有向グラフの場合の特別な場合として扱うことが可能であり、2.4 節で簡単に述べる。

G 、 H をノード数 n の重みつき有向グラフ、 A_G 、 A_H をそれぞれの隣接行列であるとする。まず A_G 、 A_H から次のように複素行列 E_G 、 E_H を構成する。 A_G 、 A_H の対称成分、反対称成分をそれぞれ A_{GS} 、 A_{GN} 、 A_{HS} 、 A_{HN} とする。

$$A_{GS} = \frac{A_G + A_G^T}{2} \quad (2.5)$$

$$A_{GN} = \frac{A_G - A_G^T}{2} \quad (2.6)$$

$$A_{HS} = \frac{A_H + A_H^T}{2} \quad (2.7)$$

$$A_{HN} = \frac{A_H - A_H^T}{2} \quad (2.8)$$

A_G 、 A_H の対称成分を実数部、反対称成分を虚数部とする複素行列 E_G 、 E_H を定義する。

$$E_G = A_{GS} + iA_{GN} \quad (2.9)$$

$$E_H = A_{HS} + iA_{HN} \quad (2.10)$$

E_G 、 E_H は明らかにエルミート行列 ($A = A^*$ が成立する行列、ここで "*" は共役転置を示す) である。

$$E_G^* = A_{GS}^T - iA_{GN}^T = A_{GS} + iA_{GN} = E_G \quad (2.11)$$

$$E_H^* = A_{HS}^T - iA_{HN}^T = A_{HS} + iA_{HN} = E_H \quad (2.12)$$

そこで P を順列行列として

$$J'(P) = \| P E_G P^T - E_H \|^2 \quad (2.13)$$

とすると、

$$J'(P) = \| P(A_{GS} + iA_{GN})P^T - (A_{HS} + iA_{HN}) \|^2$$

$$\begin{aligned}
&= \| (P A_G S P^T - A_{HS}) + i(P A_G N P^T - A_{HN}) \|^2 \\
&= \| P A_G S P^T - A_{HS} \|^2 + \| P A_G N P^T - A_{HN} \|^2 \quad (2.14)
\end{aligned}$$

であるから、 $J'(P)$ を最小にする P は A_G 、 A_H の対称、反対称成分それぞれの差の和を最小にするような順列行列となっていることがわかる。

この $J'(P)$ が $J(P)$ と同値であることは次のように示すことができる。まず

$$A_G = A_{GS} + A_{GN} \quad (2.15)$$

$$A_H = A_{HS} + A_{HN} \quad (2.16)$$

であるから、(2.3) 式から

$$\begin{aligned}
J(P) &= \| P(A_{GS} + A_{GN})P^T - (A_{HS} + A_{HN}) \|^2 \\
&= \| A_{GS} + A_{GN} \|^2 + \| A_{HS} + A_{HN} \|^2 \\
&\quad - 2\text{tr}(P(A_{GS} + A_{GN})P^T(A_{HS} + A_{HN})^T) \quad (2.17)
\end{aligned}$$

である。 A を対称行列、 B を反対称行列とすると、一般に

$$\text{tr}(AB^T) = 0 \quad (2.18)$$

$$\| A + B \|^2 = \| A \|^2 + \| B \|^2 \quad (2.19)$$

が成立するので、

$$\begin{aligned}
J(P) &= \| A_{GS} \|^2 + \| A_{GN} \|^2 + \| A_{HS} \|^2 + \| A_{HN} \|^2 \\
&\quad - 2\text{tr}(P A_{GS} P^T A_{HS}) - 2\text{tr}(P A_{GN} P^T A_{HN}^T) \quad (2.20)
\end{aligned}$$

となる。一方明らかに (2.14) 式から

$$\begin{aligned}
J'(P) &= \| A_{GS} \|^2 + \| A_{GN} \|^2 + \| A_{HS} \|^2 + \| A_{HN} \|^2 \\
&\quad - 2\text{tr}(P A_{GS} P^T A_{HS}) - 2\text{tr}(P A_{GN} P^T A_{HN}^T) \quad (2.21)
\end{aligned}$$

であり、 $J'(P) = J(P)$ となることがわかる。

さて、グラフ G 、 H 間の最適対応は $J(P)$ (もしくは $J'(P)$) を最小にするような順列行列 P で定義されるが、これを直接求めることは前節でも述べたように非常

に困難である。しかし、一方、評価関数 J' の定義域を順列行列 P の範囲からユニタリ行列 W の範囲まで拡大した場合（ここで、転置 T は共役転置 * に置き換える）、 $J'(W) = \|WE_GW^* - E_H\|^2$ を最小にするようなユニタリ行列 W は、次に述べるように複素行列 E_G, E_H の固有値分解を利用することにより、陽に求めることができる。順列行列はユニタリ行列の一種であるからこの定義域の拡大は極めて自然なものである。本章で提案するアルゴリズムはこのようなユニタリ行列 W をてがかりとして、 G, H 間の準最適な対応を求めるものである。

これからの議論においては複素行列 E_G, E_H のそれぞれ n 個の固有値は、いずれも互いに異なっているものと仮定する。これは一般の重みつきグラフに対しては通常、問題なく成立する仮定である。

2.3.1 準最適対応の決定

まず定理 2.1 を示す。この定理は本質的には二つの正規行列 (normal matrix) の差のノルムの下界に対する Hoffman & Wielandt [30] の定理に基づくものである。Hoffman & Wielandt [30] は線形計画法に基づく非常にエレガントな証明をその定理に対して与えているが、ここでは Wilkinson [71] に従ってより初等的な証明を付録 A に与える。

定理 2.1 A, B を $n \times n$ エルミート行列 ($A = A^*, B = B^*$ が成立する行列、ここで * は共役転置を示す) で、それぞれ相異なる固有値、 $\alpha_1 > \alpha_2 > \dots > \alpha_n$ 、 $\beta_1 > \beta_2 > \dots > \beta_n$ を持つとする。またその固有値展開を

$$A = U_A \Lambda_A U_A^* \quad (2.22)$$

$$B = U_B \Lambda_B U_B^* \quad (2.23)$$

とする。ここで U_A, U_B はユニタリ行列、 $\Lambda_A = \text{diag}(\alpha_i)$ 、 $\Lambda_B = \text{diag}(\beta_i)$ である。このとき、 $\|WAW^* - B\|^2$ を最小にするユニタリ行列 W は次式で与えられる。

$$W = U_B S U_A^*, \quad S \in S \quad (2.24)$$

$$S = \{\text{diag}(s_i) \mid s_i \text{ is any complex number satisfying } |s_i| = 1\} \quad (2.25)$$

またこのときの最小値は $\sum_{i=1}^n (\alpha_i - \beta_i)^2$ である。

この定理から、評価関数 J' の定義域を順列行列の範囲からユニタリ行列の範囲まで拡大した場合、 $J'(W)$ を最小にするようなユニタリ行列 W は

$$W = U_H S U_G^*, S \in S \quad (2.26)$$

で与えられることがわかる。ここで、複素行列 E_G 、 E_H の固有値展開を

$$E_G = U_G \Lambda_G U_G^* \quad (2.27)$$

$$E_H = U_H \Lambda_H U_H^* \quad (2.28)$$

としている。

グラフ G と H が同型の時にはある順列行列 P に対して (2.4) 式が成立するため、(2.26) 式のユニタリ行列 W が正確に順列行列となるような対角行列 S が S 内に存在するはずである。つまり、そのような S を S 内で探索することにより求める順列行列、つまりグラフ G 、 H 間の最適な対応を得ることが原理的に可能である。

次にグラフ G 、 H が同型でない場合であるが、この場合にも、もしそれらが同型に近い場合には、(2.26) 式で定義される W の中には順列行列に非常に近いユニタリ行列が存在する可能性が高く、そのようなユニタリ行列から得られる順列行列（ユニタリ行列に一番近い順列行列）は二つのグラフ間の最適対応、あるいは最適な対応に非常に近い対応となっていることが期待できる。そこで (2.26) 式の W の中から順列行列に最も近いものを探索し、対応する順列行列を準最適な対応として採用するという手法が考えられる。本論文で示す重みつきグラフ間の対応手法はこのようなアイデアに基づくものである。ただし (2.26) 式で定義される W は無限個存在するため、それらの中から目的の行列を見つけ出すためにはもう一段の工夫が必要となる。

このような対応の決定法は G 、 H が非常に同型に近い場合にのみ妥当であるかもしれない。しかし、たとえ二つのグラフが十分同型に近くなかったとしても、このようにして決定される順列行列には G と H 間の大局的な対応関係が反映されていると考えられる。このため例えば、その解は山登り法のような繰返しの解改良法に対する良好な初期値となることが期待される。このことは後に 2.6 節で計算機実験によって示される。

2.3.2 準最適対応の決定法 I

評価関数 $J'(W)$ を最小にするユニタリ行列は (2.26) 式で与えられる。このようなユニタリ行列 W の中から、最も順列行列に近いものを選び出すためには (2.25) 式で定義される無限個の S の中から $W = U_H S U_G^*$ を最も順列行列へ近づけるような S を決定する必要がある。このために、ここでは順列行列が二重確率行列 (doubly stochastic matrix) の一種であるという性質を用いる。二重確率行列とは要素が全て非負で各行各列の要素の和がそれぞれちょうど 1 となるような (実数) 行列のことであり、ある行列が順列行列であるための必要十分条件は、その行列が二重確率でユニタリ (要素の非負性からもちろん直交) であるということである。このように、順列行列は明らかに二重確率行列であるから、(2.26) 式で定められる W のうち、最も二重確率行列に近いものを探索する。

行列 W の二重確率行列への近さの測度として次式を用いる。

$$\begin{aligned} D(W) &= \|We - e\|^2 \\ &= 2n - e^T W e - e^T W^* e \end{aligned} \quad (2.29)$$

$e = (1, 1, \dots, 1)^T$ である。また、 W はユニタリ行列であるから $\|W^* e - e\|^2 = \|We - e\|^2$ が成立するため列方向への和は考慮しなくとも良い。 $W = U_H S U_G^*$ を代入すると

$$\begin{aligned} D(S) &= 2n - e^T U_H S U_G^* e - e^T U_G S^* U_H^* e \\ &= 2n - h^* S g - g^* S^* h \\ &= 2n - 2\{Re(h_1^* g_1 s_1) + Re(h_2^* g_2 s_2) + \dots + Re(h_n^* g_n s_n)\} \end{aligned} \quad (2.30)$$

となる。ここで

$$g = U_G^* e = \begin{pmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{pmatrix} \quad (2.31)$$

$$h = U_H^* e = \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{pmatrix} \quad (2.32)$$

とする。 g_i 、 h_i 、 s_i を極座標表現して、

$$g_i = |g_i| e^{i\alpha_i} \quad (2.33)$$

$$h_i = |h_i| e^{i\beta_i} \quad (2.34)$$

$$s_i = e^{i\theta_i} \quad (2.35)$$

とすると、

$$h_i^* g_i s_i = |g_i| |h_i| e^{i(\alpha_i - \beta_i + \theta_i)} \quad (2.36)$$

$$\operatorname{Re}(h_i^* g_i s_i) = |g_i| |h_i| \cos(\alpha_i - \beta_i + \theta_i) \quad (2.37)$$

であるから、 $D(S)$ を最小にするためには

$$\alpha_i - \beta_i + \theta_i = 0 \quad (2.38)$$

でなければならない。つまり次のように各 s_i を選べば良いことがわかる。

$$s_i = e^{i(-\alpha_i + \beta_i)} \quad (2.39)$$

次に、このように決定された $W = U_H S U_G^*$ に最も近い順列行列を決定する。このために $\|P - W\|^2$ を最小にする順列行列 P を求める。 W の実数部を W_r 、虚数部を W_i とすると、

$$\begin{aligned} \|P - W\|^2 &= \|P - (W_r + iW_i)\|^2 \\ &= \|P - W_r\|^2 + \|W_i\|^2 \\ &= \|P\|^2 + \|W_r\|^2 - 2\operatorname{tr}(P^T W_r) + \|W_i\|^2 \\ &= 2n - 2\operatorname{tr}(P^T W_r) \end{aligned} \quad (2.40)$$

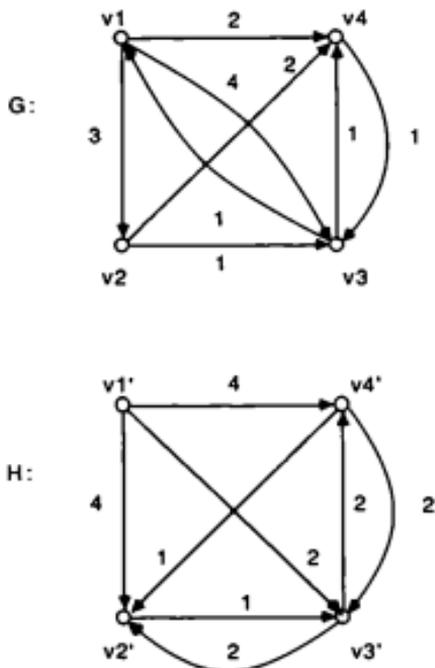


図 2.2: 重みつき有向グラフ間の最適対応問題の例

であるから、 $\text{tr}(P^T W_r)$ を最大にする P を求めれば良い。 $\text{tr}(P^T W_r)$ を最大化する順列行列を決定する問題は、行列 W_r の各行各列からそれぞれちょうど 1 個ずつ要素を選んでその和を最大にする問題であるから、これは重みつきバイパータイトグラフのマッチング問題に他ならず、Hungarian method を用いることにより、 $O(n^3)$ の時間で解くことができる [44]。

計算例

提案する手法の理解を明確にするため、図 2.2 に示す重みつき有向グラフ G 、 H に対して手法を適用してみる。

これらのグラフの隣接行列は次のようになる。

$$A_G = \begin{pmatrix} 0.0 & 3.0 & 4.0 & 2.0 \\ 0.0 & 0.0 & 1.0 & 2.0 \\ 1.0 & 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \end{pmatrix} \quad (2.41)$$

$$A_H = \begin{pmatrix} 0.0 & 4.0 & 2.0 & 4.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 2.0 & 0.0 & 2.0 \\ 0.0 & 1.0 & 2.0 & 0.0 \end{pmatrix} \quad (2.42)$$

これから、

$$E_G = \begin{pmatrix} 0.0 & 1.5 & 2.5 & 1.0 \\ 1.5 & 0.0 & 0.5 & 1.0 \\ 2.5 & 0.5 & 0.0 & 1.0 \\ 1.0 & 1.0 & 1.0 & 0.0 \end{pmatrix} + i \begin{pmatrix} 0.0 & 1.5 & 1.5 & 1.0 \\ -1.5 & 0.0 & 0.5 & 1.0 \\ -1.5 & -0.5 & 0.0 & 0.0 \\ -1.0 & -1.0 & 0.0 & 0.0 \end{pmatrix} \quad (2.43)$$

$$E_H = \begin{pmatrix} 0.0 & 2.0 & 1.0 & 2.0 \\ 2.0 & 0.0 & 1.5 & 0.5 \\ 1.0 & 1.5 & 0.0 & 2.0 \\ 2.0 & 0.5 & 2.0 & 0.0 \end{pmatrix} + i \begin{pmatrix} 0.0 & 2.0 & 1.0 & 2.0 \\ -2.0 & 0.0 & -0.5 & -0.5 \\ -1.0 & 0.5 & 0.0 & 0.0 \\ -2.0 & 0.5 & 0.0 & 0.0 \end{pmatrix} \quad (2.44)$$

となる。これらの行列の固有値および固有ベクトル行列はそれぞれ次のように計算される。

$$\Lambda_G = \text{diag}(4.764, 0.111, -1.339, -3.536) \quad (2.45)$$

$$\begin{aligned}
 U_G = & \begin{pmatrix} 0.380 & 0.189 & 0.030 & 0.704 \\ 0.399 & -0.049 & -0.264 & -0.109 \\ 0.499 & 0.170 & -0.231 & -0.457 \\ 0.408 & -0.494 & 0.758 & -0.121 \end{pmatrix} \\
 & + i \begin{pmatrix} 0.485 & 0.248 & -0.118 & -0.113 \\ 0.183 & -0.612 & -0.433 & 0.402 \\ 0.112 & 0.503 & 0.316 & 0.306 \\ 0.000 & -0.000 & 0.000 & -0.000 \end{pmatrix} \quad (2.46)
 \end{aligned}$$

$$\Lambda_H = \text{diag}(5.689, -0.352, -1.392, -3.946) \quad (2.47)$$

$$\begin{aligned}
 U_H = & \begin{pmatrix} 0.451 & 0.254 & -0.394 & -0.430 \\ 0.450 & 0.700 & 0.188 & 0.490 \\ 0.447 & -0.267 & 0.714 & -0.204 \\ 0.495 & -0.582 & -0.379 & 0.522 \end{pmatrix} \\
 & + i \begin{pmatrix} 0.371 & -0.073 & -0.083 & -0.493 \\ -0.112 & -0.054 & 0.081 & 0.105 \\ -0.004 & 0.166 & -0.378 & -0.086 \\ 0.000 & -0.000 & -0.000 & 0.000 \end{pmatrix} \quad (2.48)
 \end{aligned}$$

これらから、

$$g = U_G^* e = \begin{pmatrix} 1.858e^{-0.433i} \\ 0.231e^{-2.496i} \\ 0.375e^{0.673i} \\ 0.596e^{-1.542i} \end{pmatrix} \quad (2.49)$$

$$h = U_H^* e = \begin{pmatrix} 1.860e^{-0.137i} \\ 0.112e^{-0.335i} \\ 0.401e^{1.244i} \\ 0.606e^{0.897i} \end{pmatrix} \quad (2.50)$$

となる。(2.39)式に従って各 s_i を定めると、

$$S = \text{diag}(e^{0.296i}, e^{2.140i}, e^{0.571i}, e^{2.439i}) \quad (2.51)$$

この S を用いれば、

$$\begin{aligned}
 W &= U_H S U_G^* \\
 &= \begin{pmatrix} 0.876 & 0.233 & 0.041 & -0.126 \\ -0.076 & -0.134 & 0.790 & 0.493 \\ 0.320 & 0.095 & -0.211 & 0.758 \\ -0.117 & 0.852 & 0.353 & -0.155 \end{pmatrix} \\
 &\quad + i \begin{pmatrix} 0.194 & -0.244 & 0.200 & -0.153 \\ 0.119 & -0.152 & 0.184 & -0.196 \\ -0.146 & 0.243 & -0.345 & 0.266 \\ -0.196 & 0.227 & -0.106 & 0.105 \end{pmatrix} \quad (2.52)
 \end{aligned}$$

が得られる。この W の実数部に対して重みつきバイパートグラフのマッチング問題に対する Hungarian method を適用すると次式の順列行列 P が得られる。

$$P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad (2.53)$$

つまり、 G の v_1 は H の v_1' に、 v_2 は v_3' に、 v_3 は v_4' に、 v_4 は v_2' に対応づけられることになる。これは最適対応である。

2.3.3 準最適対応の決定法 II

決定法 I を用いることにより通常 S は一意に決定されるが、たまたま $h_i^* g_i = 0$ となるような場合には、 s_i を決定することができなくなってしまふ。このようなことは、例えばグラフの対称性が高い場合、つまりグラフが自明でないような自己同型写像を持つような場合に起こりえる（これについては後述する）。このような場合についても準最適対応を決定するため、本節では前節のように随に S を決定することなく、準最適対応を決定する手法を考察する。

さて、ユニタリ行列 $W = U_H S U_G^*$ を最も順列行列に近づけるように S を選択した時の、対応する順列行列 P を決定したいわけであるから、

$$\begin{aligned} D'(P, S) &= \|P - U_H S U_G^*\|^2 \\ &= 2n - 2\operatorname{Re}\{\operatorname{tr}(P^T U_H S U_G^*)\} \end{aligned} \quad (2.54)$$

を最小にする S および P を (同時に) 決定することになる。ところで、順列行列 P の表現する順列を $\pi(i)$ (つまり P の $(i, \pi(i))$ 要素が 1) とし、 $U_H = [h_{ij}]$ 、 $U_G = [g_{ij}]$ 、 $S = \operatorname{diag}(s_i)$ とすると、

$$\operatorname{tr}(P^T U_H S U_G^*) = \sum_{i=1}^n \sum_{j=1}^n s_j h_{ij} g_{\pi(i), j}^* \quad (2.55)$$

であり、明らかに、

$$\begin{aligned} \operatorname{Re}\left\{\sum_{i=1}^n \sum_{j=1}^n s_j h_{ij} g_{\pi(i), j}^*\right\} &\leq \left|\sum_{i=1}^n \sum_{j=1}^n s_j h_{ij} g_{\pi(i), j}^*\right| \\ &\leq \sum_{i=1}^n \sum_{j=1}^n |s_j h_{ij} g_{\pi(i), j}^*| \\ &= \sum_{i=1}^n \sum_{j=1}^n |s_j| |h_{ij}| |g_{\pi(i), j}| \\ &= \sum_{i=1}^n \sum_{j=1}^n |h_{ij}| |g_{\pi(i), j}| \end{aligned} \quad (2.56)$$

であるから、次式が成立する。

$$\begin{aligned} \operatorname{Re}\{\operatorname{tr}(P^T U_H S U_G^*)\} &\leq \sum_{i=1}^n \sum_{j=1}^n |h_{ij}| |g_{\pi(i), j}| \\ &= \operatorname{tr}(P^T O_H O_G^T) \end{aligned} \quad (2.57)$$

ここで O_H および O_G は行列 U_H 、 U_G の各要素の絶対値を要素とする行列である。

$$O_H = [\tilde{h}_{ij}], \quad \tilde{h}_{ij} = |h_{ij}| \quad (2.58)$$

$$O_G = [\tilde{g}_{ij}], \quad \tilde{g}_{ij} = |g_{ij}| \quad (2.59)$$

行列 O_H , O_G の各行ベクトルの長さは 1 であり、その要素は非負であるから、 $O_H O_G^T$ の各 (i, j) 要素 x_{ij} (O_H の i 行ベクトルと O_G の j 行ベクトルの内積) の大きさは次のように制限され、

$$0 \leq x_{ij} \leq 1 \quad (2.60)$$

任意の順列行列 P に対して

$$\text{tr}(P^T O_H O_G^T) \leq n \quad (2.61)$$

となる。そこで、(2.57) 式、(2.61) 式から、任意の P 、 S について、次式が成立することがわかる。

$$D'(P, S) \geq 2n - 2\text{tr}(P^T O_H O_G^T) \geq 0 \quad (2.62)$$

以上の考察から、陽に S を求めない準最適対応決定法として、 $D'(P, S)$ を最小とする順列行列 P を、 $D'(P, S)$ の S に関する下界 ($2n - 2\text{tr}(P^T O_H O_G^T)$) を最小とする順列行列 P で代用することが考えられる。この式を最小化する問題、言い換えれば

$$\text{tr}(P^T O_H O_G^T) \quad (2.63)$$

を最大化する順列行列 P を決定する問題は、前節と同様に重みつきバイパートグラフのマッチング問題であるから、行列 $O_H O_G^T$ に対して Hungarian method を適用することにより求めることが可能である。このような準最適対応決定法を決定法 II と呼ぶ。

実際、グラフ G と H が同型であると仮定すると、 $W = U_H S U_G^*$ を正確に順列行列とするような対角行列 \hat{S} が S 内に存在し、その対角行列 \hat{S} とその時の順列行列 $\hat{P} = U_H \hat{S} U_G^*$ に対して $D'(\hat{P}, \hat{S}) = 0$ となるため、(2.62) 式から

$$\text{tr}(\hat{P}^T O_H O_G^T) = n \quad (2.64)$$

が成立する。この式は、 \hat{P} がまさしく (2.63) 式を最大化する順列行列に他ならないことを示している。つまり、二つのグラフが同型の場合には、確かに決定法 II によって最適対応を決定できることが判る。

計算例

図 2.2 に示す重みつき有向グラフ G 、 H 間に対して、決定法 II を適用する。

これらのグラフの隣接行列から導かれるエルミート行列 E_G 、 E_H とその固有値展開は (2.43) (2.44) (2.45) (2.46) (2.47) (2.48) 式のとうりであるから、

$$O_G = \begin{pmatrix} 0.616 & 0.312 & 0.121 & 0.713 \\ 0.439 & 0.614 & 0.507 & 0.417 \\ 0.511 & 0.531 & 0.392 & 0.550 \\ 0.408 & 0.494 & 0.758 & 0.121 \end{pmatrix} \quad (2.65)$$

$$O_H = \begin{pmatrix} 0.583 & 0.264 & 0.403 & 0.654 \\ 0.463 & 0.702 & 0.205 & 0.501 \\ 0.447 & 0.315 & 0.808 & 0.221 \\ 0.495 & 0.582 & 0.379 & 0.522 \end{pmatrix} \quad (2.66)$$

となる。これらから、

$$O_H O_G^T = \begin{pmatrix} 0.957 & 0.895 & 0.956 & 0.753 \\ 0.886 & 0.947 & 0.966 & 0.751 \\ 0.629 & 0.891 & 0.834 & 0.977 \\ 0.905 & 0.984 & 0.998 & 0.840 \end{pmatrix} \quad (2.67)$$

が得られる。前節と同様にこの行列に対して Hungarian method を適用すると、次式の順列行列が得られる。これは (2.53) 式と同じものであり、やはりこの場合にも最適対応が得られている。

$$P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad (2.68)$$

2.4 重みつき無向グラフ間の最適対応

グラフ G 、 H が重みつき無向グラフの場合には、それぞれの隣接行列 A_G 、 A_H はそれ自身対称行列であるから、それらに対して直接定理 2.1 を適用することができ

る。この場合、 A_G 、 A_H は共に実数値行列であるから、それぞれの固有値展開を

$$A_G = U_G \Lambda_G U_G^T \quad (2.69)$$

$$A_H = U_H \Lambda_H U_H^T \quad (2.70)$$

とすると (U_G 、 U_H は直交行列)、前節と同様の議論により、無向グラフの場合についても次の二種の準最適対応決定法が得られる。¹

準最適対応の決定法 I

無向グラフの場合は各行列が実数値行列であるため、計算はいくらか簡単になる。

$$g = U_G^T e = \begin{pmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{pmatrix} \quad (2.71)$$

$$h = U_H^T e = \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{pmatrix} \quad (2.72)$$

として

$$s_i = \begin{cases} 1 & \text{if } h_i g_i \geq 0 \\ -1 & \text{if } h_i g_i < 0 \end{cases} \quad (2.73)$$

として $S = \text{diag}(s_i)$ を決定し、 $W = U_H S U_G^T$ に最も近い順列行列を準最適対応として決定する。

¹ G 、 H が無向グラフの場合には、評価関数 J の定義域をユニタリ行列の範囲まで拡大するのではなく、直交行列の範囲まで拡大すると考えることにより、

$$S' = \{\text{diag}(s_i) \mid s_i = 1 \text{ or } s_i = -1\}$$

とできる。この場合、探索すべき範囲は有限 (2^n 個) である。次数 n の順列行列の個数は $n!$ であるから、このようなアプローチにより解の探索範囲は $n!$ から 2^n へ減少することになる。例えば $n = 10$ の場合で $n! = 3628800$ 、 $2^n = 1024$ であるから、この差は小さくない。梅山 [61, 64] が提案している対応手法は直接このような探索を行なうものである。

最適対応の決定法 II

この場合も有向グラフの場合とまったく同様であり、 O_G, O_H を行列 U_G, U_H の各要素の絶対値を要素とする行列として、 $\text{tr}(P^T O_H O_G^T)$ を最大とする順列行列を求める。

2.5 最適対応手法から導かれるグラフのいくつかの性質

グラフ間の最適対応手法に対する考察から、グラフの自己同型写像に関するいくつかの性質を容易に導くことができる。

定理 2.2 (Mowshowitz [40]) 無向グラフ G の隣接行列 A_G の固有値はすべて異なるとする。この時、 G の自明でないどの自己同型写像もその位数は2である。

証明: A_G の固有値展開を $A_G = U \Lambda U^T$ とし、グラフ G の自明でない自己同型写像を順列行列で表現したものを P ($P \neq I$) とする。この時、前節の議論および脚注から

$$P = USU^T, \quad S \in S' \quad (2.74)$$

と表現できる。 S の各要素は1か-1であるから、

$$\begin{aligned} P^2 &= USU^T USU^T \\ &= US^2 U^T \\ &= I \end{aligned} \quad (2.75)$$

となる。ゆえに P の位数は2である。(証明終)

定理 2.3 (Chao [12]) 有向グラフ G の隣接行列から作られるエルミート行列 E_G の固有値がすべて異なるとする。この時、 G の自己同型写像全体が作る群はアーベル群(可換群)である。

証明: E_G の固有値展開を $E_G = U \Lambda U^*$ とする。グラフ G の相異なる自己同型写像を順列行列で表現したものを P_1, P_2 とする。前節の議論から

$$P_1 = US_1 U^*, \quad S_1 \in S \quad (2.76)$$

$$P_2 = US_2U^*, S_2 \in S \quad (2.77)$$

と表現できる。対角行列どうしは可換であるから、明らかに

$$\begin{aligned} P_1P_2 &= US_1S_2U^* \\ &= US_2S_1U^* \\ &= P_2P_1 \end{aligned} \quad (2.78)$$

よって、 G の自己同型写像全体はアーベル群である。(証明終)

定理 2.4 有向グラフ G の隣接行列から作られるエルミート行列 E_G の固有値がすべて異なるとする。 G に自明でない自己同型写像が存在するならば、 E_G の固有ベクトルのうち、その要素の総和が 0 となるものが存在する。

証明： E_G の固有値展開を $E_G = U\Lambda U^*$ とする。前節の議論から

$$J(P) = \|PE_GP^T - E_G\|^2 \quad (2.79)$$

を最小 (= 0) とする順列行列 P は

$$P = USU^* \quad (2.80)$$

と表現できる。 $S \in S$ である。 G がもし自明でない自己同型写像を持つ場合には、上式で $S = I$ という自明な解以外に $S \neq I$ であるような解も存在することになる。ところが一方、もし E_G の固有ベクトルのうちにその要素の総和が 0 となるものが存在しないとすると、2.3.2 節で述べた S の決定手続を用いることにより、この S は完全に一意に決定されるはずである。これは $J(P)$ を最小にする S が複数存在するという仮定に矛盾している。よって、 G に自明でない自己同型写像が存在する場合には、 E_G の固有ベクトルのうち、その要素の総和が 0 となるものが存在するはずである。

(証明終)

例

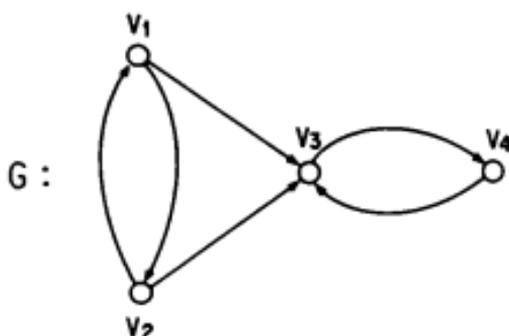


図 2.3: 自明でない自己同型写像を持つグラフ

図 2.3 の (通常の数値) グラフ G を考える。 G は明らかに対称性をもち、自明でない自己同型写像が存在する。 G から E_G を構成し、その固有値、固有ベクトル行列を求めると次のようになる。

$$L_G = \text{diag}(1.802, 0.445, -1.000, -1.247) \quad (2.81)$$

$$U_G = \begin{pmatrix} 0.368 & 0.296 & 0.500 & -0.164 \\ 0.368 & 0.296 & -0.500 & -0.164 \\ 0.591 & -0.328 & 0.000 & 0.737 \\ 0.328 & -0.737 & -0.000 & -0.591 \end{pmatrix} + i \begin{pmatrix} 0.368 & 0.296 & 0.500 & -0.164 \\ 0.368 & 0.296 & -0.500 & -0.164 \\ -0.000 & -0.000 & -0.000 & -0.000 \\ 0.000 & -0.000 & -0.000 & -0.000 \end{pmatrix} \quad (2.82)$$

これから、

$$g = U_G^* e = \begin{pmatrix} 1.813e^{-0.419i} \\ 0.758e^{-2.247i} \\ 0 \\ 0.375e^{2.077i} \end{pmatrix} \quad (2.83)$$

であり、確かに E_G の固有ベクトルで、その要素の総和が 0 となるものが存在することがわかる。

定理 2.4 で、 G が無向グラフの場合にも同様な性質が成立することは明らかである。

2.6 計算機実験と考察

提案する重みつきグラフ間の最適対応手法の有効性を確かめるため、計算機実験を行なった。二つの重みつきグラフを人工的に生成し、それらに対して提案する手法を適用し、評価関数 $J(P)$ およびその計算時間を測定した。また、比較のため木探索と簡単な山登り法についても実験を行なった。

2.6.1 実験方法

実験の対象となる人工的なデータとして大きさ n の重みつきグラフ G 、 H を次のように生成した。

重みつきグラフ G

0 から 1.0 の範囲の疑似一様乱数を各アークの重みとして与えることにより生成する。

重みつきグラフ H

- (1) 上記のように生成された G の各アークに対し、さらに $-\epsilon$ から ϵ の範囲の疑似一様乱数をノイズとして与え、さらにその後、ノードのラベルを適当に入れ替えることによって生成する。あるいは、
- (2) G と同様に 0 から 1.0 の範囲の疑似乱数を各アークの重みとして与えることにより生成する。この場合には $\epsilon = \text{"random"}$ と記すことにする。

さまざまな大きさ、ノイズレベルのグラフ対をアルゴリズムの入力として生成し、それらに対して次のような 6 種類の手法を適用した。

B B You [73] により提案されている分枝限定法を用いた木探索アルゴリズム。これは真の解を求めるために用いた。

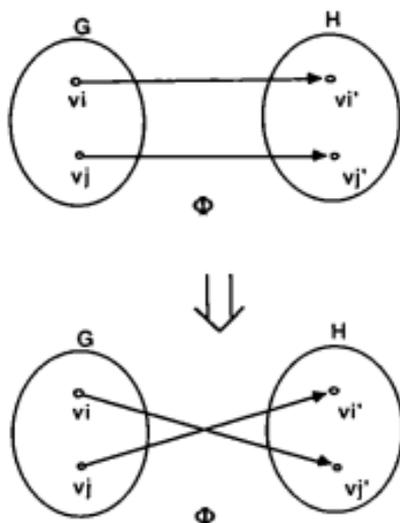


図 2.4: ノード対応の局所的入れ替え

HC 山登り法。つまり、適当に選ばれた初期対応から、常に評価関数 $J(P)$ を最も大きく減少させるように選ばれた二つのノード対応間の入れ替え（例えば図 2.4 に示すように、 $v_i \rightarrow v_i', v_j \rightarrow v_j'$ を $v_i \rightarrow v_j', v_j \rightarrow v_i'$ ）を行なう。評価関数をさらに減少させることができなくなれば停止する。

AX1 2.3.2節で提案した準最適対応の決定法 I。

AX2 2.3.3節で提案した準最適対応の決定法 II。

AX1+HC 準最適対応の決定法 I で得られた対応を初期対応として、山登り法を用いてさらに改良を行なう。

AX2+HC 準最適対応の決定法 II で得られた対応を初期対応として、山登り法を用いてさらに改良を行なう。

グラフの大きさは $n = 6$ から $n = 30$ まで二つおきを選び、ノイズレベルについては $\epsilon = 0.0, 0.05, 0.10, 0.15, 0.20$ および $\epsilon = \text{"random"}$ の場合それぞれについて実験を行なった。ただし計算時間の制約から木探索の手法については $n > 12$ の場合については行っていない。実験は各パラメータの組 (n, ϵ) について 100 回を行ない、それぞれに対して上記の 6 種の手法を適用し、評価関数の値の平均値、実行時間の平均値、および正解個数 (100 回の試行の内、最適解を得るのに成功した場合の数) を求めた。実験は SUN4/280 を用いて行なった。(他の章も含めて、本論文で行なっている実験はすべて SUN4/280 上で行なっている。)

2.8.2 結果および考察

実験結果の一部を表 2.1、表 2.2、表 2.3、表 2.4 に示す。

表 2.1、表 2.2 は、ノイズの大きさが $\epsilon = 0.0, 0.05, 0.10, 0.15, 0.20$ および $\epsilon = \text{"random"}$ の場合について、それぞれ各手法で得られた評価関数の平均値および 100 回の試行内の正解個数である。表 2.1 が有向グラフの場合、表 2.2 が無向グラフの場合であり、グラフの大きさが $n = 10, 20, 30$ の場合について示してある。ただし、 $n = 20$ および $n = 30$ の場合には計算時間の制約から木探索による最適対応 (BB) を行なっておらず、このため真の解が不明なため、これらの場合には正解個数が示されていない。また、それぞれの表の最後の行は評価関数の理論的な期待値を示している。ノード間の対応関係がノイズの付加によって変化しないと仮定すると、グラフの大きさが n 、ノイズの大きさが ϵ の時の評価関数の期待値 $E(n, \epsilon)$ は

$$E(n, \epsilon) = \frac{1}{3}\epsilon^2 n(n-1) \quad (2.84)$$

で与えられる。これは、 $-\epsilon$ から ϵ の範囲に一樣に分布するノイズの分散が $\frac{1}{3}\epsilon^2$ となるためである。

表 2.3、表 2.4 は、グラフの大きさが $n = 6$ から $n = 30$ の場合について、それぞれ各手法を実行するのに要した計算時間の平均値を示している。表 2.1 が有向グラフの場合、表 2.2 が無向グラフの場合であり、ノイズの大きさが $\epsilon = 0.10$ および $\epsilon = 0.20$ の場合を示してある。

これらの表から以下のことがわかる。

- 分枝限定法 (BB) のような厳密な解法は、グラフの大きさがある程度大きくなると計算時間の制約からその実行が非常に困難となる。
- 山登り法 (HC) はそれほど計算時間が必要ではなく、かなり大きなグラフに対しても適用可能であり、平均評価関数値も比較的良好な値が得られる。しかし、局所的な最適化手法であるというその本質的な特性から、結果的に真の最適対応が得られることは少なく、特に G 、 H が同型に非常に近いような場合でも最適対応を発見することには失敗することが多い。ノイズの大きさ ϵ の値に係わらず同じような平均評価関数値が得られている点からみても、このような単純な山登り法で大域的な最適対応を得ることの困難さは明らかである。
- 提案している準最適対応決定法 (AX1, AX2) は共に実験を行なった手法内で最適の手法であり、 ϵ が 0.1 以下の範囲、特に有向グラフ間のマッチングの場合にはほとんどの場合に真の最適対応を見出すことに成功している。計算時間については AX1, AX2 両者間でほとんど差は無い。ほぼ、グラフの大きさ n の三乗に比例した計算時間を要している。どちらの手法についても、平均評価関数値については、有向グラフ間の最適対応の方が無向グラフの場合よりも総じてより良い結果を与えているが、これは有向グラフの方が最適対応に利用できる情報量が多いという意味で自然な結果と思われる。また、有向グラフの場合には AX1 が、無向グラフの場合には AX2 がより良い平均評価関数値を与えている。無向グラフの場合に、AX1 が AX2 に比較してはかばかしくない結果となっている原因であるが、例えば無向グラフは有向グラフに比較して対称性の高いグラフになりやすいなどの理由により、(2.73) 式の $h_i^* g_i \approx 0$ となる場合が多く、 s_i を 1、-1 にいずれかに決定してしまうことが困難となるためであると思われる。 s_i を 1、-1 のどちらかに決定するかで、最終的に得られる対応関係は大きく異なってくる。この意味で AX2 の方がより安定な手法と言える。最後に ϵ が 0.15 以上の時、特に無向グラフの場合にはこれらの手法はどちらもそれほど良い結果を与えない。
- 準最適対応決定法で得られた対応を初期対応として、山登り法により改良する手法 (AX1+HC, AX2+HC) は、両者共に実験を行なった手法内では

最良の平均評価関数値を与えており、ノイズがかなり大きい場合にもほとんどの場合に真の最適対応を与えることに成功している。特に注目すべき点は、 $AX1$ 、 $AX2$ のみでは十分に良好な評価関数値が得られないようなノイズの大きな場合 ($0.20 \geq \epsilon \geq 0.15$) についても、かなり良好な結果が得られていることであり、提案している準最適対応決定法を用いて得られる初期対応が、大局的なノード間の対応関係を良く反映したものとなっていることがわかる。

- G 、 H 両グラフとも完全にランダムに構成した場合 ($\epsilon = \text{"random"}$) には、たとえ準最適対応決定法を山登り法の初期対応の決定に用いたとしても、ランダムに初期対応を選んだ場合とその平均評価関数値はほとんど変わらない。これは提案する準最適対応決定法が二つのグラフ間の十分な同型性を前提として成立しているものであることから考えて、自然な結果であると思われる。

2.7 むすび

無向、有向それぞれの場合について、重みつきグラフ間の最適対応問題に対する近似解法を二種与えた。いずれもグラフの隣接行列、あるいは隣接行列から導かれるエルミート行列の固有値展開を利用した解析的手法であり、木探索法のような“組み合わせ的爆発”を伴わず、また大局的な最適化手法となっている。処理に必要とする計算時間は、ほぼグラフの大きさの三乗に比例しており、計算機実験の結果、グラフが完全に同型の場合および同型に非常に近い場合には、ほとんどの場合に正しい最適対応を与えることが判った。また、ある程度同型から外れている場合でも、これらの近似解法に山登り法を併用することにより、良好なマッチング結果が得られることが判った。これは、提案した近似解法を用いて得られる初期対応が、ノード間の大局的な対応関係を良く反映したものとなっていることを示している。さらに、グラフの自己同型写像に関する幾つかの結果も示した。

表 2.1: 有向グラフ間の最適対応実験結果 (平均評価関数値および正解個数)

| n = 10 | | | | | | | | |
|------------|------------|------------|------------|------------|------------|------------|------------------|--|
| ϵ | BB | HC | AX1 | AX2 | AX1+HC | AX2+HC | $E(n, \epsilon)$ | |
| 0.00 | 0.00 (100) | 5.51 (35) | 0.00 (100) | 0.00 (100) | 0.00 (100) | 0.00 (100) | 0.00 | |
| 0.05 | 0.08 (100) | 5.66 (34) | 0.08 (100) | 0.26 (98) | 0.08 (100) | 0.08 (100) | 0.075 | |
| 0.10 | 0.30 (100) | 6.12 (30) | 0.56 (96) | 1.36 (84) | 0.30 (100) | 0.34 (99) | 0.300 | |
| 0.15 | 0.68 (100) | 6.48 (29) | 1.74 (83) | 4.11 (59) | 0.88 (100) | 0.68 (100) | 0.675 | |
| 0.20 | 1.21 (100) | 7.01 (28) | 3.71 (64) | 7.98 (31) | 1.46 (97) | 1.65 (94) | 1.200 | |
| random | - (-) | 9.03 (-) | 12.48 (-) | 14.44 (-) | 8.90 (-) | 8.91 (-) | - | |
| n = 20 | | | | | | | | |
| ϵ | BB | HC | AX1 | AX2 | AX1+HC | AX2+HC | $E(n, \epsilon)$ | |
| 0.00 | - (-) | 31.38 (-) | 0.00 (-) | 0.00 (-) | 0.00 (-) | 0.00 (-) | 0.000 | |
| 0.05 | - (-) | 31.96 (-) | 0.31 (-) | 0.31 (-) | 0.31 (-) | 0.31 (-) | 0.317 | |
| 0.10 | - (-) | 33.19 (-) | 1.44 (-) | 3.39 (-) | 1.25 (-) | 1.25 (-) | 1.267 | |
| 0.15 | - (-) | 35.70 (-) | 7.53 (-) | 23.39 (-) | 2.82 (-) | 2.82 (-) | 2.850 | |
| 0.20 | - (-) | 37.99 (-) | 22.96 (-) | 51.11 (-) | 5.43 (-) | 6.72 (-) | 5.067 | |
| random | - (-) | 42.23 (-) | 57.03 (-) | 62.90 (-) | 41.63 (-) | 41.93 (-) | - | |
| n = 30 | | | | | | | | |
| ϵ | BB | HC | AX1 | AX2 | AX1+HC | AX2+HC | $E(n, \epsilon)$ | |
| 0.00 | - (-) | 91.21 (-) | 0.00 (-) | 0.00 (-) | 0.00 (-) | 0.00 (-) | 0.000 | |
| 0.05 | - (-) | 94.08 (-) | 0.72 (-) | 0.72 (-) | 0.72 (-) | 0.72 (-) | 0.725 | |
| 0.10 | - (-) | 94.24 (-) | 3.39 (-) | 7.64 (-) | 2.89 (-) | 2.89 (-) | 2.900 | |
| 0.15 | - (-) | 96.68 (-) | 29.28 (-) | 77.73 (-) | 6.51 (-) | 6.51 (-) | 6.525 | |
| 0.20 | - (-) | 105.02 (-) | 90.89 (-) | 138.66 (-) | 12.47 (-) | 16.40 (-) | 11.600 | |
| random | - (-) | 101.96 (-) | 133.08 (-) | 144.24 (-) | 101.47 (-) | 102.24 (-) | - | |

BB : 分枝限定法による未探索

HC : 山登り法

AX1 : 最適対応決定法 I

AX2 : 最適対応決定法 II

AX1+HC : 最適対応決定法 I の結果を初期対応として山登り

AX2+HC : 最適対応決定法 II の結果を初期対応として山登り

n : 実験を行なうグラフの大きさ

 ϵ : グラフ H を構成する際に加えたノイズの大きさ $E(n, \epsilon)$: 平均評価関数値の理論的な期待値

各項目の左側は 100 回試行を行なったことによる平均評価関数値、右側はそのうち其の最適対応が得られた試行の回数を表す。

表 2.2: 無向グラフ間の最適対応実験結果 (平均評価関数値および正解個数)

| n = 10 | | | | | | | | |
|------------|------------|-----------|------------|------------|------------|------------|------------------|-------|
| ϵ | BB | HC | AX1 | AX2 | AX1+HC | AX2+HC | $E(n, \epsilon)$ | |
| 0.00 | 0.00 (100) | 4.51 (16) | 0.00 (100) | 0.00 (100) | 0.00 (100) | 0.00 (100) | 0.00 (100) | 0.000 |
| 0.05 | 0.07 (100) | 4.46 (16) | 2.23 (60) | 0.29 (95) | 0.16 (98) | 0.07 (100) | 0.07 (100) | 0.075 |
| 0.10 | 0.30 (100) | 4.75 (13) | 4.08 (41) | 1.79 (78) | 0.61 (94) | 0.33 (99) | 0.30 (100) | 0.300 |
| 0.15 | 0.67 (100) | 5.10 (11) | 5.98 (22) | 5.58 (37) | 1.68 (76) | 0.83 (96) | 0.67 (100) | 0.675 |
| 0.20 | 1.19 (100) | 5.41 (11) | 7.94 (8) | 9.68 (13) | 2.58 (66) | 2.22 (76) | 1.20 (100) | 1.200 |
| random | - (-) | 6.78 (-) | 10.46 (-) | 13.62 (-) | 6.69 (-) | 6.74 (-) | - | - |

| n = 20 | | | | | | | | |
|------------|-------|-----------|-----------|-----------|-----------|-----------|------------------|-------|
| ϵ | BB | HC | AX1 | AX2 | AX1+HC | AX2+HC | $E(n, \epsilon)$ | |
| 0.00 | - (-) | 30.61 (-) | 0.00 (-) | 0.00 (-) | 0.00 (-) | 0.00 (-) | 0.00 (-) | 0.000 |
| 0.05 | - (-) | 30.41 (-) | 4.88 (-) | 0.60 (-) | 0.31 (-) | 0.31 (-) | 0.31 (-) | 0.317 |
| 0.10 | - (-) | 32.13 (-) | 19.10 (-) | 9.61 (-) | 1.59 (-) | 1.25 (-) | 1.25 (-) | 1.267 |
| 0.15 | - (-) | 32.20 (-) | 33.16 (-) | 36.18 (-) | 7.24 (-) | 3.41 (-) | 3.41 (-) | 2.850 |
| 0.20 | - (-) | 34.37 (-) | 44.92 (-) | 55.24 (-) | 15.95 (-) | 11.97 (-) | 11.97 (-) | 5.067 |
| random | - (-) | 33.86 (-) | 50.60 (-) | 62.40 (-) | 33.32 (-) | 33.62 (-) | - | - |

| n = 30 | | | | | | | | |
|------------|-------|-----------|------------|------------|-----------|-----------|------------------|--------|
| ϵ | BB | HC | AX1 | AX2 | AX1+HC | AX2+HC | $E(n, \epsilon)$ | |
| 0.00 | - (-) | 82.17 (-) | 0.00 (-) | 0.00 (-) | 0.00 (-) | 0.00 (-) | 0.00 (-) | 0.000 |
| 0.05 | - (-) | 81.11 (-) | 7.09 (-) | 1.77 (-) | 0.72 (-) | 0.72 (-) | 0.72 (-) | 0.725 |
| 0.10 | - (-) | 82.88 (-) | 51.31 (-) | 29.47 (-) | 2.88 (-) | 2.88 (-) | 2.90 (-) | 2.900 |
| 0.15 | - (-) | 86.01 (-) | 97.45 (-) | 104.38 (-) | 17.15 (-) | 9.09 (-) | 9.09 (-) | 6.525 |
| 0.20 | - (-) | 90.94 (-) | 123.81 (-) | 141.71 (-) | 47.95 (-) | 34.82 (-) | 34.82 (-) | 11.600 |
| random | - (-) | 84.99 (-) | 123.97 (-) | 143.39 (-) | 84.07 (-) | 84.79 (-) | - | - |

- BB : 分枝限定法による本探索
 HC : 山登り法
 AX1 : 最適対応決定法 I
 AX2 : 最適対応決定法 II
 AX1+HC : 最適対応決定法 I の結果を初期対応として山登り
 AX2+HC : 最適対応決定法 II の結果を初期対応として山登り
 n : 実験を行なうグラフの大きさ
 ϵ : グラフ H を構成する際に加えたノイズの大きさ
 $E(n, \epsilon)$: 平均評価関数値の理論的な期待値

各項目の左側は 100 回試行を行なったことによる平均評価関数値、右側はそのうち真の最適対応が得られた試行の回数である。

表 2.3: 有向グラフ間の最適対応実験結果 (平均実行時間)

| $\epsilon = 0.10$ | | | | | | |
|-------------------|------|------|------|------|--------|--------|
| n | BB | HC | AX1 | AX2 | AX1+HC | AX2+HC |
| 6 | 0.03 | 0.01 | 0.02 | 0.02 | 0.02 | 0.02 |
| 8 | 0.19 | 0.04 | 0.04 | 0.04 | 0.05 | 0.05 |
| 10 | 0.82 | 0.11 | 0.07 | 0.07 | 0.09 | 0.09 |
| 12 | 4.07 | 0.23 | 0.11 | 0.10 | 0.14 | 0.14 |
| 14 | - | 0.45 | 0.16 | 0.15 | 0.21 | 0.21 |
| 16 | - | 0.78 | 0.23 | 0.21 | 0.30 | 0.30 |
| 18 | - | 1.23 | 0.31 | 0.29 | 0.41 | 0.42 |
| 20 | - | 1.91 | 0.41 | 0.39 | 0.56 | 0.56 |
| 22 | - | 2.77 | 0.52 | 0.49 | 0.72 | 0.72 |
| 24 | - | 3.89 | 0.66 | 0.62 | 0.91 | 0.94 |
| 26 | - | 5.42 | 0.83 | 0.78 | 1.16 | 1.15 |
| 28 | - | 7.01 | 1.02 | 0.94 | 1.45 | 1.47 |
| 30 | - | 9.24 | 1.20 | 1.10 | 1.71 | 1.73 |

| $\epsilon = 0.20$ | | | | | | |
|-------------------|------|------|------|------|--------|--------|
| n | BB | HC | AX1 | AX2 | AX1+HC | AX2+HC |
| 6 | 0.04 | 0.01 | 0.02 | 0.02 | 0.03 | 0.03 |
| 8 | 0.49 | 0.04 | 0.04 | 0.04 | 0.05 | 0.06 |
| 10 | 2.70 | 0.10 | 0.07 | 0.07 | 0.10 | 0.12 |
| 12 | - | 0.22 | 0.11 | 0.10 | 0.16 | 0.22 |
| 14 | - | 0.46 | 0.16 | 0.15 | 0.28 | 0.39 |
| 16 | - | 0.74 | 0.22 | 0.22 | 0.43 | 0.68 |
| 18 | - | 1.19 | 0.30 | 0.29 | 0.72 | 1.04 |
| 20 | - | 1.79 | 0.40 | 0.40 | 0.98 | 1.63 |
| 22 | - | 2.62 | 0.52 | 0.51 | 1.49 | 2.48 |
| 24 | - | 3.82 | 0.66 | 0.63 | 2.44 | 3.69 |
| 26 | - | 5.18 | 0.82 | 0.78 | 3.10 | 5.40 |
| 28 | - | 6.47 | 1.01 | 0.95 | 4.67 | 7.32 |
| 30 | - | 9.16 | 1.22 | 1.15 | 6.54 | 10.27 |

- BB : 分枝限定法による木探索
 HC : 山登り法
 AX1 : 階層最適対応決定法 I
 AX2 : 階層最適対応決定法 II
 AX1+HC : 階層最適対応決定法 I の結果を初期対応として山登り
 AX2+HC : 階層最適対応決定法 II の結果を初期対応として山登り
 n : 実験を行なうグラフの大きさ
 ϵ : グラフ H を構成する際に加えたノイズの大きさ

各項目は 100 回試行を行なったことによる平均計算時間 (SEC.)。

表 2.4: 無向グラフ間の最適対応実験結果 (平均実行時間)

| $\epsilon = 0.10$ | | | | | | |
|-------------------|------|------|------|------|--------|--------|
| n | BB | HC | AX1 | AX2 | AX1+HC | AX2+HC |
| 6 | 0.04 | 0.01 | 0.01 | 0.01 | 0.01 | 0.01 |
| 8 | 0.28 | 0.02 | 0.02 | 0.02 | 0.03 | 0.03 |
| 10 | 2.06 | 0.06 | 0.04 | 0.03 | 0.06 | 0.05 |
| 12 | 8.86 | 0.13 | 0.05 | 0.05 | 0.09 | 0.08 |
| 14 | - | 0.25 | 0.08 | 0.08 | 0.15 | 0.13 |
| 16 | - | 0.39 | 0.11 | 0.11 | 0.24 | 0.18 |
| 18 | - | 0.61 | 0.15 | 0.15 | 0.33 | 0.26 |
| 20 | - | 0.93 | 0.19 | 0.19 | 0.47 | 0.34 |
| 22 | - | 1.40 | 0.24 | 0.24 | 0.70 | 0.47 |
| 24 | - | 1.93 | 0.30 | 0.30 | 0.82 | 0.57 |
| 26 | - | 2.61 | 0.37 | 0.37 | 1.20 | 0.75 |
| 28 | - | 3.62 | 0.45 | 0.45 | 1.61 | 0.95 |
| 30 | - | 4.75 | 0.55 | 0.55 | 2.10 | 1.34 |

| $\epsilon = 0.20$ | | | | | | |
|-------------------|------|------|------|------|--------|--------|
| n | BB | HC | AX1 | AX2 | AX1+HC | AX2+HC |
| 6 | 0.07 | 0.01 | 0.01 | 0.01 | 0.01 | 0.01 |
| 8 | 0.84 | 0.02 | 0.02 | 0.02 | 0.03 | 0.04 |
| 10 | 8.92 | 0.06 | 0.04 | 0.04 | 0.07 | 0.07 |
| 12 | - | 0.13 | 0.05 | 0.05 | 0.12 | 0.13 |
| 14 | - | 0.23 | 0.08 | 0.08 | 0.21 | 0.23 |
| 16 | - | 0.38 | 0.11 | 0.11 | 0.38 | 0.39 |
| 18 | - | 0.63 | 0.15 | 0.15 | 0.62 | 0.67 |
| 20 | - | 0.96 | 0.19 | 0.20 | 0.87 | 0.97 |
| 22 | - | 1.40 | 0.25 | 0.26 | 1.39 | 1.55 |
| 24 | - | 1.94 | 0.31 | 0.33 | 1.94 | 2.14 |
| 26 | - | 2.65 | 0.38 | 0.39 | 2.66 | 2.94 |
| 28 | - | 3.40 | 0.46 | 0.48 | 3.79 | 4.16 |
| 30 | - | 4.70 | 0.56 | 0.59 | 4.82 | 5.84 |

- BB : 分枝限定法による木探索
 HC : 山登り法
 AX1 : 準最適対応決定法 I
 AX2 : 準最適対応決定法 II
 AX1+HC : 準最適対応決定法 I の結果を初期対応として山登り
 AX2+HC : 準最適対応決定法 II の結果を初期対応として山登り
 n : 実験を行なうグラフの大きさ
 ϵ : グラフ H を構成する節に加えたノイズの大きさ

各項目は 100 回試行を行なったことによる平均計算時間 (SEC.)。

第 3 章

空間記述されたパターン間のマッチング I

— 空間内の点パターン間のマッチング

3.1 はじめに

与えられた二つの画像間、あるいはあるオブジェクトとあらかじめ知られているモデルとのあいだの最適なマッチングを見つけることは、コンピュータビジョンや画像解析において最も重要な課題の一つである。たとえば、動画処理においては、連続するフレーム間のマッチングを行なうことにより動き情報のパラメータが推定されている。本章では、空間内に与えられた幾何的パターン間のマッチング問題について考察する。

マッチングに対する一般的なアプローチの一つは、各画像から特徴点を抽出し、それらを関連づける最適な対応を見つけることである。実際の応用の場面においては、特徴点は通常、位置、意味ラベル、順序（たとえば境界に沿っての）などさまざまな特性を持ち、これらの特性をてがかりとしてマッチングが行なわれることになる。しかし、ここでは問題をより抽象的にとらえ、意味ラベル等の付加情報は考えず、最も基本的な情報と考えられる位置情報のみをマッチングのための情報として使用する問題を考える。このことにより、各特徴点はユークリッド空間内に座標で表現された点となり、問題は m 次元空間内に与えられた二つの点の集合、 X 、 Y 間のマッチングの問題となる。空間内の点集合はしばしば点パターン (point pattern) と呼ばれるた

め [49, 33]、この問題を点パターンマッチング問題 (point pattern matching problem) と呼ぶ。

点パターンマッチング問題に対しては、従来からさまざまな解法が試みられてきている。例えば、Faugeras and Price [18]、Kitchen [36]、Ogawa [41]、Ranade and Rosenfeld [49] らはこの問題に対して緩和法を適用しており、そこでは点パターン X 内のある点と、点パターン Y 内のある点の間の対応の強さが、それぞれそのまわりの点への影響の大きさから決定され、その値を局所的な適合関係から更新していく (緩和過程) ことにより、全体の点の最終的な対応関係が決定される。Ranade らの手法においては、点パターン間の関係として回転は考慮されておらず、平行移動だけが許されているが、[18, 36, 41] においては回転も考慮されている。また、Baird [5] は、同数の点を含む点パターン間のマッチング問題に対して、線形計画法における Soviet ellipsoid algorithm [44] を探索木の枝刈りに巧妙に利用した、組合せ的な点パターンマッチングアルゴリズムを提案している。Baird の手法は非常に高速であり、アルゴリズムの解析も詳しく行なわれているが、その本質において、二次元空間内の相似変換が変換のパラメータに対して線形で表現されるという性質を利用しているため、適用できるのは二次元内の点パターンに限られており、一般に三次元以上へ拡張するのは困難である。三次元空間においては、もはや相似変換を変換のパラメータに対して線形で表現することはできない。一方 Wong [72] は点パターンのグラフ表現 (点の間の距離をグラフのアークの重みとして表現する) を用いた木探索によるマッチング手法を提案している。Wong の手法は当然どのような距離空間に対しても適用可能であり、また非常に効率的と思われる。しかし、点パターン間の関係がスケール変換を伴っている場合には適用できず、あるパターンと鏡像関係にあるパターンはそのグラフ表現がまったく同一となるため、それらを識別することも不可能である。また最近では計算幾何学 (Computational Geometry) 的な手法も Alt [1] によって提案されているが、やはり二次元を対象としておりスケール変換も考慮されていない。この他にも、Delaunay triangulation を用いた手法 [42] や convex hull を利用したもの [25]、あるいは全ての点間の距離のソートされたリストを用いるもの [55, 38] など、さまざまな手法が提案されている [13, 33]。

本章では m 次元空間内に与えられた点パターン間の、座標情報のみに基づく新し

いマッチング手法を提案する。従来の手法の中には、緩和法のように最終的に得られる解がどのようなものとなるか、直観的に判りづらいものも少なくないが、ここではこれを避けるため、まず求める解に対する厳密な定義を与える。このため、マッチングの許容誤差を定義するものとして δ 制限という概念を導入し、二つの点パターンが与えられた時に、この制限を満足する最大のマッチングすべてを効率的に生成するアルゴリズムを考察する。

点パターンマッチングの問題は、一般に、二つのパターン内の点の数が同じでそのすべての点の間の対応を決定する問題と、二つのパターン内の点の数が異なり、対応する相手を持たないような点が存在する場合の二通りが考えられる。本章ではまず最初により一般的な後者の問題を扱い、その後、前者の問題に特有な手法について考察する。また、点パターンマッチング問題は別の見方をすれば、 X を Y の上へできるだけうまく重ね合わせるような変換を求める問題であるが、ここでは X と Y を関連づける変換として相似変換（回転、平行移動、スケール変換）の範囲を考えることとする。提案する手法は木探索アルゴリズムとして定式化され、 δ 制限という解の定義から導かれた条件にもとづいて枝刈りが行なわれる [65, 66, 68]。手法は高速であり、またあらかじめ与えられる変換パラメータに対する制約（スケール、回転角度の許容範囲など）を探索に利用することも容易である。パターンの次元 m は任意であるが、実際に意味があるのは当然 $m = 2$ あるいは $m = 3$ の場合である。

3.2 点パターンマッチング問題

点パターン X とは、 m 次元ユークリッド空間内の点の集合である。 X 内の点の数を $|X|$ であらわす。二つの点パターン X, Y 間の大きさ n ($n \leq |X|$ かつ $n \leq |Y|$) のマッチング ϕ とは、 X の大きさ n の部分集合 X_ϕ と Y の同じ大きさの部分集合 Y_ϕ の間の 1 対 1 対応のことである。 X_ϕ, Y_ϕ に含まれない点は対応点を持たないとする。 $\phi(x)$ によって $x \in X_\phi$ に対応する Y 内の点を表し、 $X_\phi = \{x_1, x_2, \dots, x_n\}$ 、 $Y_\phi = \{\phi(x_1), \phi(x_2), \dots, \phi(x_n)\} = \{y_1, y_2, \dots, y_n\}$ の時、 $\phi = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ と表現する。またマッチング ϕ のサブマッチング ϕ' とは、 $X_{\phi'}$ ($X_{\phi'} \subseteq X_\phi$) から $Y_{\phi'}$ ($Y_{\phi'} \subseteq Y_\phi$) へのマッチングであ

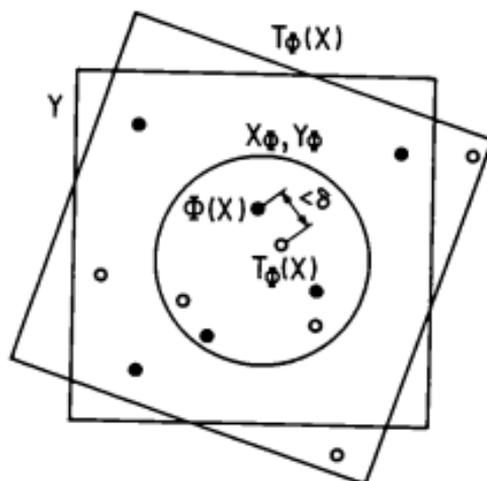


図 3.1: δ 制限されたマッチング

り、 $\Phi'(x) = \Phi(x)$ if $x \in X_\Phi$ を満足するものとする。この時、 $\Phi \supseteq \Phi'$ と書く。次に x に対する相似変換 T を次のように定義する。

$$T(x) = cRx + t \quad (3.1)$$

ここで R は $m \times m$ の回転行列、 t は平行移動ベクトル、 c はスケール ($c > 0$) である。

最後に許容できるマッチング誤差を定式化するため “ δ 制限” (δ -bounded) の概念を定義する。大きさ n のマッチング Φ が次の条件を満足する時、 Φ を δ 制限であるとよぶ。

$$d = \max_{x \in X_\Phi} \|\Phi(x) - T_\Phi(x)\| \leq \delta \quad (3.2)$$

ここで $\|\cdot\|$ はユークリッドノルムであり ($\|x\| = (\sum_{i=1}^m x_i^2)^{1/2}$)、 δ はあらかじめ定められた許容誤差、 T_Φ は二乗平均誤差の意味で最適な X_Φ から Y_Φ への相似変

換、つまり次式で表現される誤差 e^2 を最小にする変換 T とする。

$$e^2 = \frac{1}{n} \sum_{x \in X_\phi(x)} \|\phi(x) - T(x)\|^2 \quad (3.3)$$

(3.2) 式中の d を ϕ の最大誤差とよび、 e^2 の最小値、つまり最小二乗平均誤差を e^2 と書くことにする。ここで誤差は Y のスケールで測られていることに注意する。 δ 制限の定義が意味していることは、点パターン X が点パターン Y の上に最適変換 T_ϕ で重ね合わされた時、対応する点間の最大距離は δ 以下でなければならないということである (図 3.1)。

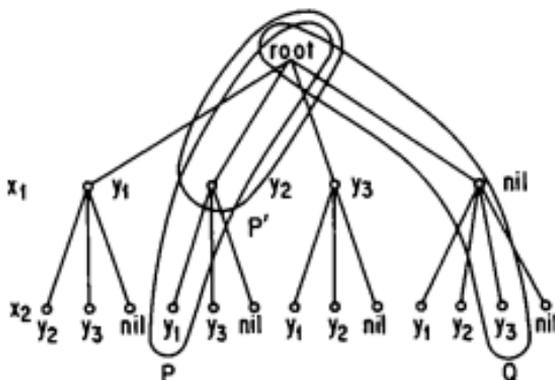
これらの定義のもとで、点パターンマッチング問題を次のように定める。点パターン X 、 Y 間の点パターンマッチング問題とは、 X 、 Y 間の δ 制限された最大の大きさのマッチングをすべて見つけ出す問題である。ここで最大の大きさとは、 δ 制限の条件のもとでマッチング可能な点の最大数である。

3.3 点パターンマッチングアルゴリズム

点パターンマッチングのアルゴリズムは木探索のアルゴリズムとして定式化される。本節ではこの木探索アルゴリズムについて述べる。

3.3.1 探索木と深さ優先探索

探索は、すべての可能なマッチングをルートからリーフへのパスとして表現した木を探索することにより行なわれる。 X を与えられた点パターンとすると、 X 内の点の比較順序をまず適当に定める。例えば $x_1, x_2, \dots, x_{|X|}$ とする。この順序はマッチングの間固定され、 X 内の点はこの順序で Y 内の点と比較される。図 3.2 は $|X| = 2$ 、 $|Y| = 3$ の場合の探索木を示している。図中の各ノードは X 内のある点 (ノードの深さで示される) と Y 内のある点 (ノードのラベルで示される) の対応を示している。すべての可能なマッチングはルートからリーフへのパス (あるいは、リーフノードによってこのパスを代表させるなら、リーフノードそのもの) によって表現され、ルートからこのパス上にある任意のノードへの部分パス (あるいは、同様にこのノードそのもの) はこのマッチングの一部 (部分マッチングと呼ぶ) を定義すること



nil : no corresponding point in Y.

図 3.2: マッチングの探索木 ($|X| = 2$, $|Y| = 3$)

になる。部分マッチングは、マッチングの無定義の部分、つまりその部分パスの先端のノードからその先リーフまでの部分を無視して考えれば、パス全体が表すマッチングのサブマッチングと考えることができる。例えば図 3.2 の例で見ると、パス $P = ((x_1, y_2), (x_2, y_1))$ はマッチング $\{(x_1, y_2), (x_2, y_1)\}$ を表現しており、 P の部分パス $P' = ((x_1, y_2))$ はこのマッチングのサブマッチング $\{(x_1, y_2)\}$ を定めている。図 3.2 では多くのノードが "nil" をそのラベルとして持っているが、これはその点が対応点を Y 内に持たないことを示している。例えば、パス $Q = ((x_1, nil), (x_2, y_3))$ はマッチング $\{(x_2, y_3)\}$ を表現している。このように、探索木中のパス

$$P = ((x_1, y_{j_1}), (x_2, y_{j_2}), \dots, (x_{|X|}, y_{j_{|X|}})), y_{j_i} \in Y \text{ or } y_{j_i} = \text{nil} \quad (3.4)$$

は、マッチング

$$\Phi_P = \{(x_i, y_{j_i}) \mid (x_i, y_{j_i}) \in P \text{ and } y_{j_i} \neq \text{nil}\} \quad (3.5)$$

を表現していることになる。アルゴリズムの中で木の探索は深さ優先で行なう。そこで任意のパス

$$P = ((x_1, y_{j_1}), (x_2, y_{j_2}), \dots, (x_n, y_{j_n})) \quad (3.6)$$

に対してこれのすべての部分パス

$$P^k = ((x_1, y_{j_1}), (x_2, y_{j_2}), \dots, (x_k, y_{j_k})) \quad (k < n) \quad (3.7)$$

は P より先に調べられることになる。

δ 制限された最大のマッチングを見つけるためには、このように木の中のルートからリーフへのパスとして表現されたすべてのマッチングについてその δ 制限性を調べなければならない。しかし、マッチングの可能な組合せの数は莫大なものであり、またパターンが大きくなるにつれて急速に増大するため、これを実際に行なうのは不可能である。そこで、実用的なマッチングアルゴリズムにおいては、効果的な枝刈り手法を用いることが必須となる。

ここでは二種の枝刈り手法を採用する。このうち前者は、探索中に得られた既発見のマッチングの大きさに基づく枝刈り手法であり、後者は点パターン X 、 Y 間の相似変換のスケールに関する制約から導かれる枝刈り手法である。マッチングの計算時間は、実際に探索を行なう木の大きさと条件をテストするのに要する時間という二つの要素で決定される。このうち木の大きさは、深さ優先探索とこれから述べる二つの枝刈り条件を採用するならば、点パターンの幾何学的構造と X 内の点の比較順序だけで決定されるものであり、二つの点パターンが与えられた時点で決ってしまう。このことから、採用する枝刈り条件が高速にテスト可能であることは非常に重要である。後で述べるように、スケールに関する制約から導かれる枝刈り条件は、探索が進むにつれて漸進的に計算可能であり、効率的に計算可能である。

以下にこれら二つの枝刈り手法を説明する。

3.3.2 既発見のマッチングの大きさによる枝刈り

探索木中のあるノードにおいて、そのノードを展開することによって既に得られている δ 制限マッチングよりも大きなマッチングを得られる可能性がまったく無い場合には、そのノードの展開を行なわない。つまり、そのノードにおいて、対応点が既

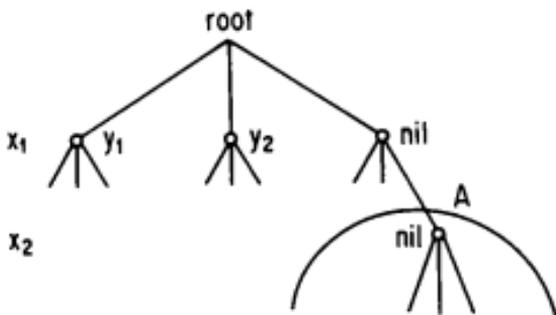


図 3.3: 既発見のマッチングの大きさによる枝刈り ($|X| = 4$, $N = 3$)

に見つかっている点の数とまだ対応を調べていない点の数の和が、既に見られている最大のマッチングの大きさより小さい場合には、以降の対応がすべて成功したとしても、必ず既発見のマッチングより小さいマッチングしか得られないわけであり、このような場合には枝刈りを行なう。例えば、 $|X| = 4$ の時、図 3.3 の “A” で示される部分木が表すマッチングの大きさは明らかにすべて 2 以下であるから、大きさが 3 のマッチングが既に見つかっている場合には、この先を探索することは明らかに無駄であり、この部分木 “A” は枝刈り可能である。

3.3.3 スケールの適格範囲による枝刈り

あるマッチング ϕ が δ 制限である、あるいはより大きな δ 制限マッチングのサブマッチングであるためには、点パターン間の相似変換のスケールについてある制約が課せられる。ここでは、そのような制約に基づく探索木の枝刈り手法について説明する。

まず、あるマッチング ϕ 中の点パターン X_{ϕ} 、 Y_{ϕ} 間の相似変換のスケール c に

対する条件 $EI(\Phi)$ を次のように定義し

$$EI(\Phi) = \{c \mid \exists t, R \max_{(x,y) \in \Phi} \|y - (cRx + t)\| \leq \delta\} \quad (3.8)$$

これを、マッチング Φ のスケールに関する適格範囲 (eligible interval of scale) と呼ぶ。この定義から、 $EI(\Phi)$ は次のような性質を持つことがわかる。

性質 1 任意のマッチング Φ, Φ' について、 $\Phi \subseteq \Phi'$ ならば、 $EI(\Phi) \supseteq EI(\Phi')$ である。

証明： 定義から明らかである。(証明終)

性質 2 マッチング Φ について、 $EI(\Phi) = \emptyset$ (\emptyset は空集合を示す) ならば、 Φ は δ 制限ではない。また、 Φ を含むどのようなマッチング Φ' 、($\Phi' \supseteq \Phi$) も δ 制限ではない。

証明： $EI(\Phi) = \emptyset$ の時には、定義より Φ 中の全ての対応点を δ 以下の距離内に移すような相似変換は存在しないため、 Φ は δ 制限ではありえない。また性質 1 から、 $EI(\Phi) = \emptyset$ の時、 $\Phi' \supseteq \Phi$ となるような全ての Φ' について $EI(\Phi') = \emptyset$ となり、 Φ' も明らかに δ 制限ではない。(証明終)

性質 3 マッチング Φ について $EI(\Phi) \neq \emptyset$ とする。 Φ を拡大することにより、十分 δ 制限に近いマッチング Φ' を構成することができる。

証明： $c \in EI(\Phi)$ であるとする。この時、適格範囲の定義により、任意の対応点対 $(x, y) \in \Phi$ について

$$\|y - (cRx + t)\| \leq \delta \quad (3.9)$$

が成立する回転行列 R 、平行移動ベクトル t が存在する。さて、 Φ に対して、

$$y' = cRx' + t \quad (3.10)$$

を満足するような対応点対 (x', y') を次々と追加し、マッチング ϕ' を構成する。もし、このように新たに付加された対応点対の数が、 ϕ にもともと含まれていた対応点対の数に比較して十分多ければ、 ϕ' から定まる最適相似変換 $T_{\phi'}$ はさきほどの (c, R, t) に十分近くなることが予想され、そのような相似変換に対しては (3.2) 式が近似的に成立するであろうと考えられる。よって、 ϕ を拡大することにより、十分 δ 制限に近いマッチング ϕ' を構成することが可能である。(証明終)

上記の性質から、探索木中の各ノードが示す部分マッチングの適格範囲 $EI(\phi)$ が空集合でないことは、その部分マッチング自身およびそれを拡大して得られるマッチングが δ 制限となるための必要条件(性質3を考慮すればほとんど必要十分条件)であることがわかる。つまり、探索木中の各ノードについて、そのノードが示すマッチングの適格範囲を求め、それが空集合ならば、そのノードをルートとする部分木は枝刈りすることができる。

しかし、適格範囲を一般の m 次元のパターンに対して効率的に求めることは容易ではない。そこでここでは、 $EI(\phi)$ の近似 $EI'(\phi)$ を高速に計算することを考える。任意のマッチング ϕ について $EI'(\phi) \supseteq EI(\phi)$ が成立するならば、 $EI'(\phi) = \phi$ の時、 $EI(\phi) = \phi$ であり、効率はある程度落ちるかもしれないが、必要な枝を誤って刈り取ることなく、枝刈りを行なうことが可能である。このような近似 $EI'(\phi)$ を、 $EI(\phi)$ の安全な近似と呼ぶ。以下では適格範囲 $EI(\phi)$ の安全な近似を行なう手法を考察する。

最小平均二乗誤差を利用したスケールの適格範囲の近似

スケールの適格範囲の近似は、マッチングの最小平均二乗誤差の値を利用して行なうことができる。このことを行なうために、まずマッチング ϕ ($\text{size}(\phi) = |X_{\phi}| = |Y_{\phi}| = n$ とする) に対して、次のようなスケールの範囲 $I(\phi)$ を定義する。

$$I(\phi) = \{c \mid \min_{t, R} \frac{1}{n} \sum_{(x, y) \in \phi} \|y - (cRx + t)\|^2 \leq \delta^2\} \quad (3.11)$$

つまり、平行移動および回転 (t, R) に関する最小平均二乗誤差が δ^2 以下であるようなスケール c の範囲である。明らかに、 ϕ の大きさが 1 以下の場合には、 $I(\phi) =$

$[0, \infty)$ である。このような $I(\Phi)$ に対して、次の定理が成立する。

定理 3.1 任意のマッチング Φ 、 Φ' について、 $\Phi' \subseteq \Phi$ の時、 $I(\Phi') \supseteq EI(\Phi)$ である。

証明： まず $\Phi = \Phi'$ とする。 $c \in EI(\Phi)$ の時、定義から、ある t 、 R が存在して

$$\max_{(x,y) \in \Phi} \|y - (cRx + t)\| \leq \delta \quad (3.12)$$

よって、この t 、 R に対して、

$$\frac{1}{n} \sum_{(x,y) \in \Phi} \|y - (cRx + t)\|^2 \leq \delta^2 \quad (3.13)$$

が成立する。明らかに、 t 、 R に関する最小平均二乗誤差は式左辺の値より小さくなるため、 $c \in I(\Phi)$ である。よって、 $I(\Phi') \supseteq EI(\Phi)$ が成立する。次に、 $\Phi' \subset \Phi$ の時であるが、この時、性質 1 から $EI(\Phi') \supseteq EI(\Phi)$ であり、また、上記の結果を使うと $I(\Phi') \supseteq EI(\Phi')$ であるから、この場合にも $I(\Phi') \supseteq EI(\Phi)$ が成立する。(証明終)

この定理から、マッチング Φ の任意のサブマッチング Φ' ($\Phi' \subseteq \Phi$) について、 $I(\Phi')$ はスケールの適格範囲 $EI(\Phi)$ の安全な近似になっていることがわかる。

$$I(\Phi') \supseteq EI(\Phi) \quad (3.14)$$

また適格範囲と異なり、 $I(\Phi)$ は $\Phi' \subseteq \Phi$ であっても、 $I(\Phi') \supseteq I(\Phi)$ であるとは限らない。これらの点から、 $EI(\Phi)$ のより精密な近似として、マッチング Φ のサブマッチング Φ' に対する $I(\Phi')$ すべての共通部分を採用することが考えられる。

$$EI'(\Phi) = \bigcap_{\Phi' \subseteq \Phi} I(\Phi') \quad (3.15)$$

明らかに

$$EI'(\Phi) \supseteq EI(\Phi) \quad (3.16)$$

であり、 $EI'(\Phi) = \phi$ の時には、 Φ に対応するノード以下の探索木を枝刈りすることが可能である。

しかし、 Φ のサブマッチング Φ' は、 Φ の大きさを n として、 2^n 個存在する。これらすべてについて $I(\Phi')$ を求めることは、あまりにも計算の負担が大きすぎ、またそれらのうちには冗長な部分も多いであろうことが想像される。このため、ここでは上記の $EI'(\Phi)$ ではなく、次に定義するスケールの範囲 $EI''(\Phi)$ を Φ の適格範囲の近似として採用することとする。 $EI''(\Phi)$ は $EI_1(\Phi)$ と $EI_2(\Phi)$ という二つの部分から構成されており、 $EI_1(\Phi)$ は Φ の大きさ 2 のサブマッチングすべてから導かれる条件として Φ のスケールに関する局所的な制約を、 $EI_2(\Phi)$ は Φ が全体として満たすべき条件として Φ の大局的な制約を表現したもとなっている。

$$EI''(\Phi) = EI_1(\Phi) \cap EI_2(\Phi) \quad (3.17)$$

$$EI_1(\Phi) = \bigcap_{\Phi' \subset \Phi \text{ and size}(\Phi')=2} I(\Phi') \quad (3.18)$$

$$EI_2(\Phi) = \bigcap_{i=0}^n I(\Phi_i) \quad (3.19)$$

ただし $\Phi_0 (= \phi), \Phi_1, \dots, \Phi_{n-1}, \Phi_n (= \Phi)$ は、木探索につれて次第に拡大されたマッチングの列であるとする。ここで、局所的な制約をテストする単位として、大きさ 3 以上のマッチングではなく、大きさ 2 のマッチングを選ぶのは、後で示すように、大きさ 2 のマッチング Φ に対する $I(\Phi)$ の計算は各点相互間の距離から極めて簡単に計算することができるためである。

この適格範囲の近似 $EI''(\Phi)$ はマッチングを拡大していくにつれて漸進的に計算可能である。 $\Phi_0 (= \phi), \Phi_1, \dots, \Phi_{n-1}, \Phi_n, \dots$ を、木探索につれて次第に拡大されるマッチングの列であるとし、 $\Phi_n = \Phi_{n-1} \cup \{(x, y)\}$ とすると、

$$EI_1(\Phi_n) = EI_1(\Phi_{n-1}) \cap \bigcap_{(x', y') \in \Phi_{n-1}} I(\{(x', y'), (x, y)\}) \quad (3.20)$$

$$EI_2(\Phi_n) = EI_2(\Phi_{n-1}) \cap I(\Phi_n) \quad (3.21)$$

であるから、

$$EI''(\Phi_0) = [0, \infty) \quad (3.22)$$

$$EI''(\Phi_n) = EI''(\Phi_{n-1}) \cap I(\Phi_n) \cap \bigcap_{(x', y') \in \Phi_{n-1}} I(\{(x', y'), (x, y)\}) \quad (3.23)$$

となる。

後で述べるように、(3.23)式右辺の第二項（大局的制約の項）はマッチングの大きさに依存しない定数時間で計算可能であり、また第三項（局所的制約の項）は、明らかに高々マッチングの大きさに比例した時間で計算可能である。 $EI''(\Phi)$ のこのような性質は、実際に木探索を行なう場合には計算を効率的に実行する上で非常に重要である。

$I(\Phi)$ の計算

最小平均二乗誤差によるスケールの適格範囲の近似を枝刈りに利用するためには、マッチング Φ に対して $I(\Phi)$ を計算する必要がある。言い換えれば、あるスケール c が与えられた時の平行移動および回転 (t, R) に関するマッチングの最小平均二乗誤差 $c^2(c)$ を求めねばならない。これは次の定理から導かれる。

定理 3.2 Φ を点パターン X, Y 間の大きさ n ($n \geq 2$) のマッチングとし、 $X_\Phi = \{x_1, x_2, \dots, x_n\}$ 、 $Y_\Phi = \{\Phi(x_1), \Phi(x_2), \dots, \Phi(x_n)\} = \{y_1, y_2, \dots, y_n\}$ とする。あるスケール c が与えられた時、

$$c^2(R, t, c) = \frac{1}{n} \sum_{i=1}^n \|y_i - (cR x_i + t)\|^2 \quad (3.24)$$

の平行移動ベクトル t 、回転行列 R に関する最小値 $c^2(c)$ は

$$c^2(c) = \sigma_y^2 + c^2 \sigma_x^2 - 2 \text{ctr}(DS) \quad (3.25)$$

で与えられる。ここで

$$\mu_x = \frac{1}{n} \sum_{i=1}^n x_i \quad (3.26)$$

$$\mu_y = \frac{1}{n} \sum_{i=1}^n y_i \quad (3.27)$$

$$\sigma_x^2 = \frac{1}{n} \sum_{i=1}^n \|x_i - \mu_x\|^2 = \frac{1}{n} \sum_{i=1}^n \|x_i\|^2 - \|\mu_x\|^2 \quad (3.28)$$

$$\sigma_y^2 = \frac{1}{n} \sum_{i=1}^n \|y_i - \mu_y\|^2 = \frac{1}{n} \sum_{i=1}^n \|y_i\|^2 - \|\mu_y\|^2 \quad (3.29)$$

$$\begin{aligned} \Sigma_{xy} &= \frac{1}{n} \sum_{i=1}^n (y_i - \mu_y)(x_i - \mu_x)^T = \frac{1}{n} \sum_{i=1}^n y_i x_i^T - \mu_y \mu_x^T \\ &= UDV^T : \Sigma_{xy} \text{ の特異値分解} \end{aligned} \quad (3.30)$$

$$D = \text{diag}(d_i), \quad d_1 \geq d_2 \geq \dots \geq d_m \geq 0 \quad (3.31)$$

$$S = \begin{cases} I & \text{if } \det(\Sigma_{xy}) \geq 0 \\ \text{diag}(1, 1, \dots, 1, -1) & \text{if } \det(\Sigma_{xy}) < 0 \end{cases} \quad (3.32)$$

である。 Σ_{xy} は X_Φ , Y_Φ の共分散行列、 μ_x , μ_y は X_Φ , Y_Φ の平均ベクトル、 σ_x^2 , σ_y^2 は X_Φ , Y_Φ の平均のまわりの分散である。また $\text{rank}(\Sigma_{xy}) = m$ の時、最適な回転行列および平行移動ベクトルは次のように与えられる。

$$R = USV^T \quad (3.33)$$

$$t = \mu_y - cR\mu_x \quad (3.34)$$

証明： 付録Bに示す。

この定理を用いると、マッチング ϕ に対して、(3.11) 式の $I(\phi)$ は次のように定まる。 $(\sigma_x^2 > 0)$ とする)

$$I(\phi) = \begin{cases} \phi & \text{if } \mathcal{D} < 0 \\ [c_{\min}, c_{\max}] & \text{if } \mathcal{D} \geq 0 \end{cases} \quad (3.35)$$

ここで、

$$\mathcal{D} = (\text{tr}(DS))^2 - \sigma_x^2(\sigma_y^2 - \delta^2) \quad (3.36)$$

$$c_{\min} = \max\left\{0, \frac{1}{\sigma_x^2}(\text{tr}(DS) - \sqrt{\mathcal{D}})\right\} \quad (3.37)$$

$$c_{\max} = \frac{1}{\sigma_x^2}(\text{tr}(DS) + \sqrt{\mathcal{D}}) \quad (3.38)$$

である。

定理 3.2 の中で定義されたさまざまな統計量は、次に示すようにマッチングを次第に拡大していくにつれて漸進的に計算可能である。 ϕ_{n-1} を点パターン X , Y 間の

きさ $n-1$ のマッチング、 Φ_n を Φ_{n-1} に新しい対応 (x_n, y_n) を加えた大きさ n のマッチングであるとする、 Φ_n の統計量 $(\mu_{x,n}, \mu_{y,n}, \sigma_{x,n}^2, \sigma_{y,n}^2, \Sigma_{xy,n})$ は Φ_{n-1} の統計量 $(\mu_{x,n-1}, \mu_{y,n-1}, \sigma_{x,n-1}^2, \sigma_{y,n-1}^2, \Sigma_{xy,n-1})$ より次のようにして計算される。

$$\mu_{x,n} = \frac{n-1}{n}(\mu_{x,n-1} + \frac{1}{n-1}x_n) \quad (3.39)$$

$$\mu_{y,n} = \frac{n-1}{n}(\mu_{y,n-1} + \frac{1}{n-1}y_n) \quad (3.40)$$

$$\sigma_{x,n}^2 = \frac{n-1}{n}(\sigma_{x,n-1}^2 + \frac{1}{n} \|\mu_{x,n-1} - x_n\|^2) \quad (3.41)$$

$$\sigma_{y,n}^2 = \frac{n-1}{n}(\sigma_{y,n-1}^2 + \frac{1}{n} \|\mu_{y,n-1} - y_n\|^2) \quad (3.42)$$

$$\Sigma_{xy,n} = \frac{n-1}{n} \{ \Sigma_{xy,n-1} + \frac{1}{n} (\mu_{y,n-1} - y_n)(\mu_{x,n-1} - x_n)^T \} \quad (3.43)$$

そこで、これらの統計量を各ノードへ保存しておくことにより、 $I(\Phi_n)$ の計算を、マッチングの対応点数 n に依らない定数時間で計算することが可能である。

さらに、(3.25) 式中の項 $\text{tr}(DS)$ の計算について少しふれておく。この項を計算するためには一般に Σ_{xy} の特異値分解を計算する必要がある。この計算は実際の応用においては行列の次数が小さいため (2 or 3) それほど大変ではない。さらにパターンが 2次元のものである場合には、 Σ_{xy} の特異値分解を行なうことなく次のように直接 $\text{tr}(DS)$ を求めることが可能である。まず、

$$(\text{tr}(DS))^2 = \begin{cases} (d_1 + d_2)^2 & \text{if } \det(\Sigma_{xy}) \geq 0 \\ (d_1 - d_2)^2 & \text{if } \det(\Sigma_{xy}) < 0 \end{cases} \quad (3.44)$$

であるが、

$$\|\Sigma_{xy}\|^2 = d_1^2 + d_2^2 \quad (3.45)$$

$$\det(\Sigma_{xy}) = \begin{cases} d_1 d_2 & \text{if } \det(\Sigma_{xy}) \geq 0 \\ -d_1 d_2 & \text{if } \det(\Sigma_{xy}) < 0 \end{cases} \quad (3.46)$$

であることから、

$$\begin{aligned} (\text{tr}(DS))^2 &= \text{tr}(\Sigma_{xy}\Sigma_{xy}^T) + 2\det(\Sigma_{xy}) \\ &= (s_{11} + s_{22})^2 + (s_{12} - s_{21})^2 \end{aligned} \quad (3.47)$$

となる。よって

$$\text{tr}(DS) = \sqrt{(s_{11} + s_{22})^2 + (s_{12} - s_{21})^2} \quad (3.48)$$

である。ここで

$$\Sigma_{xy} = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix} \quad (3.49)$$

としている。

最後に、定理 3.2からは、直ちに次の定理 3.3も導かれる。この定理は二つの対応する点パターンが与えられた場合に、その間の最適相似変換と最小平均二乗誤差を与える定理であり、最終的にマッチングが δ 制限であるか否かを確認したり、あるいは実際に点パターンを重ね合わせてみる際に必要な定理である。この定理と同様の結果は Arun [2] らによっても与えられているが、常に正しい回転行列（行列式が 1 の直交行列） R ((3.33) 式) を与えるという意味で定理 3.2のほうがより厳密なものとなっている。Arun らの場合には点パターンの対応状態によっては R として行列式が -1 の直交行列を与える場合があり、その場合にはもちろん正しい最小平均二乗誤差は得られない。また、三次元の点パターン間の最適変換を決定する方法としては、この他に最近、四元数 (quaternion) を利用した手法が提案されているが [19, 32]、四元数による手法は三次元の点パターンに対してしか適用できないため、任意の次元の点パターンに対して一般に適用可能という意味で、定理 3.3は優れている。

定理 3.3 Φ を点パターン X, Y 間の大きさ n ($n \geq 2$) のマッチングとし、 $X_\Phi = \{x_1, x_2, \dots, x_n\}$ 、 $Y_\Phi = \{\Phi(x_1), \Phi(x_2), \dots, \Phi(x_n)\} = \{y_1, y_2, \dots, y_n\}$ とする。この時、 X, Y 間の相似変換に対する最小平均二乗誤差 e^2 は

$$e^2 = \sigma_y^2 - \frac{\text{tr}(DS)^2}{\sigma_x^2} \quad (3.50)$$

で与えられる。また $\text{rank}(\Sigma_{xy}) = m$ の時、最適な相似変換は次のように与えられる。

$$R = USV^T \quad (3.51)$$

$$t = \mu_y - cR\mu_x \quad (3.52)$$

$$c = \frac{1}{\sigma_x^2} \text{tr}(DS) \quad (3.53)$$

ここで、式中のさまざまな統計量は、定理 3.2 で定められたものとする。

証明：(3.25) 式を c に関して最小化すれば良いわけであり、明らかである。(証明終)

大きさ 2 のマッチング ϕ に対する $I(\phi)$ の計算

前節の結果を用いると、大きさ 2 のマッチング ϕ に対する $I(\phi)$ の計算は、点パターン X 、 Y 内の各点相互間の距離のみから、特異値の計算などを行なうことなく非常に簡単に求められる。

$\phi_2 = \{(x_i, y_i), (x_j, y_j)\}$ を大きさ 2 のマッチングとする。これに対して、 \mathbf{x} 、 \mathbf{y} 、 μ_x 、 μ_y を次のように定める。

$$\mathbf{x} = \frac{1}{2}(\mathbf{x}_i - \mathbf{x}_j) \quad (3.54)$$

$$\mathbf{y} = \frac{1}{2}(\mathbf{y}_i - \mathbf{y}_j) \quad (3.55)$$

$$\mu_x = \frac{1}{2}(\mathbf{x}_i + \mathbf{x}_j) \quad (3.56)$$

$$\mu_y = \frac{1}{2}(\mathbf{y}_i + \mathbf{y}_j) \quad (3.57)$$

この時、(3.28) 式、(3.29) 式、(3.30) 式から

$$\sigma_x^2 = \|\mathbf{x}\|^2 \quad (3.58)$$

$$\sigma_y^2 = \|\mathbf{y}\|^2 \quad (3.59)$$

$$\Sigma_{xy} = \mathbf{y}\mathbf{x}^T \quad (3.60)$$

となる。ここで Σ_{xy} のランクは明らかに 1 であるから、

$$\begin{aligned} (\text{tr}(DS))^2 &= d_1^2 \\ &= \|\Sigma_{xy}\|^2 \\ &= \|\mathbf{x}\|^2 \|\mathbf{y}\|^2 \end{aligned} \quad (3.61)$$

であり、(3.36) 式の D は次のように求められる。

$$\begin{aligned}
 D &= (\text{tr}(DS))^2 - \sigma_x^2(\sigma_y^2 - \delta^2) \\
 &= \|x\|^2 \|y\|^2 - \|x\|^2 (\|y\|^2 - \delta^2) \\
 &= \|x\|^2 \delta^2
 \end{aligned} \tag{3.62}$$

ゆえに、(3.38) 式から

$$\begin{aligned}
 c_{\max} &= \frac{1}{\sigma_x^2} (\text{tr}(DS) + \sqrt{D}) \\
 &= \frac{1}{\|x\|^2} (\|x\| \|y\| + \|x\| \delta) \\
 &= \frac{1}{\|x\|} (\|y\| + \delta) \\
 &= \frac{\|y_j - y_i\| + 2\delta}{\|x_j - x_i\|}
 \end{aligned} \tag{3.63}$$

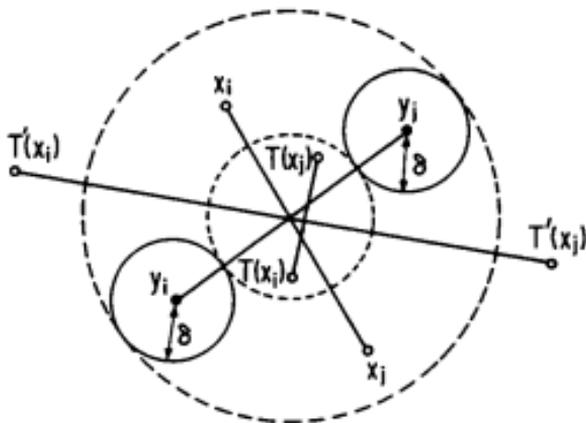
となる。 c_{\min} についても同様である。よって、大きさ 2 のマッチングに対して、 $I(\Phi_2)$ は次のように定められる。

$$I(\Phi_2) = [c_{\min}, c_{\max}] \tag{3.64}$$

$$c_{\min} = \max\left(0, \frac{\|y_j - y_i\| - 2\delta}{\|x_j - x_i\|}\right) \tag{3.65}$$

$$c_{\max} = \frac{\|y_j - y_i\| + 2\delta}{\|x_j - x_i\|} \tag{3.66}$$

図 3.4 において明らかのように、もし X から Y への相似変換のスケールがあまりにも大きすぎたり小さすぎたりしている場合には、この変換によって x_i, x_j を、 y_i, y_j を中心とした半径 δ の球の中へ移すことは、どのように回転および平行移動を選んでも明らかに不可能である。もっと正確に言えば、どのような変換も、そのスケール c が (3.64) 式で定義される範囲 $I(\Phi_2)$ の外側にある場合には、 x_i, x_j を同時にこのような球の中に移すことはできない。つまり、大きさが 2 のマッチングに対しては $EJ(\Phi_2) = I(\Phi_2)$ が成立している。



scaling factor of $T < C_{min}$
 scaling factor of $T' > C_{max}$

図 3.4: 大きさ 2 のマッチングの適格範囲

3.3.4 探索アルゴリズム

以上の準備により、点パターンマッチングのための木探索アルゴリズムは次のようになる (図 3.5)。二点から成るパターンは常にマッチング可能であるから、まず δ 制限マッチングの最大の大きさ N を 2 と仮定し、探索の各時点での最大サイズ (つまり N) のマッチングの複製を登録しておくリスト (MMリストと呼ぶ) を空に初期化する。大きさ 2 のマッチングはすべて δ 制限であるから、これらは MMリストには登録しない。 N はより大きなマッチングが発見されれば、そのマッチングの大きさに更新する。以上の準備の下で、大きさ N 以上の δ 制限マッチングの探索を深さ優先で行なう。

```

N := 2; ONlist := {P0}; MMlist := nil;
/* P0 is a root node, Φ0 = matching(P0) = φ, EIm(Φ0) = [0, ∞) */
/* ONlist is a open node list, i.e., a list of nodes waiting for expansion */
/* MMlist is a list of nodes representing currently maximum matchings */
while ONlist ≠ nil do
begin
  Pn-1 := getNode(ONlist); /* Pn-1 = ((x1, y1), ..., (xn-1, yn-1)) */
  Φ := matching(Pn-1); /* Φ is a matching represented by Pn-1 */
  k := size(Φ) + |X| - (n - 1);
  /* k is the upper bound of size of matchings obtainable by the expansion of Pn-1 */
  if k > N then
  begin
    Pn := Pn-1 ∪ (xn, nil);
    insert(Pn, ONlist); /* xn has no corresponding point in Y */
  end
  if k ≥ N then
  begin
    for all points y in Y that do not appear in Pn-1 do
    begin
      Pn := Pn-1 ∪ (xn, y);
      Φ' = matching(Pn);
      compute the eligible interval EIm(Φ');
      if EIm(Φ') = φ then discard(Pn); /* pruning of the search tree */
      else begin
        case size(Φ') of
          size(Φ') > N : begin
            compute the maximum error distance d of Φ';
            if d ≤ δ then
              begin discard(MMlist); copy(Pn, P'n);
                MMlist := {P'n}; N := N + 1 end;
            end;
          size(Φ') = N : begin copy(Pn, P'n); insert(P'n, MMlist) end;
          end;
        insert(Pn, ONlist);
      end;
    end
  end
end;
end;
end;

```

図 3.5: 点パターンマッチングアルゴリズム

```

for all  $P$  in  $MMlist$  do /* output of obtained matchings */
begin
 $\Phi := \text{matching}(P)$ ;
compute the maximum error distance  $d$  of  $\Phi$ ;
if  $d < \delta$  then output( $\Phi$ ) else discard( $P$ );
end.

```

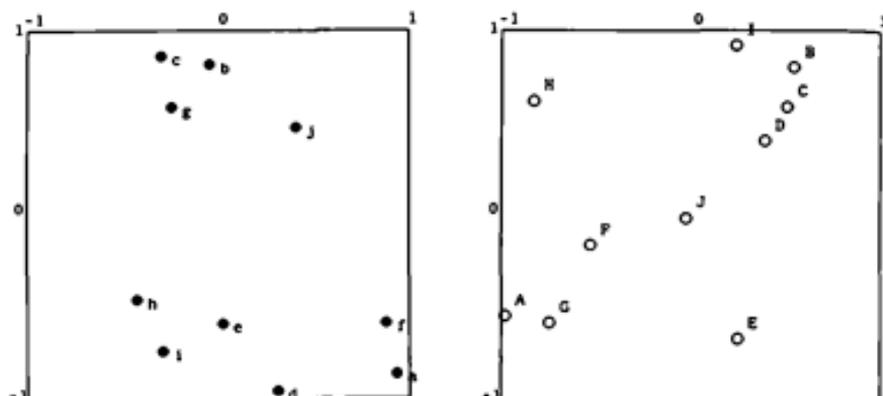
図 3.5: (続き)

探索木の各ノードによって表現される部分マッチング Φ について、そのスケールの適格範囲の近似 $EI^m(\Phi)$ を前節の手法により計算し、もしこれが空になれば、そのマッチングも含めてすべての子孫マッチングを枝刈りする。ここで、このノードの親ノードが表すマッチングを Φ' とし、 $\Phi = \Phi' \cup \{(x, y)\}$ とすると、(3.23) 式

$$EI^m(\Phi) = EI^m(\Phi') \cap I(\Phi) \cap \bigcap_{(x', y') \in \Phi'} I(\{(x', y'), (x, y)\}) \quad (3.67)$$

に従って、漸次的に $EI^m(\Phi)$ の計算を行なう。また、このノードにおいて新たな対応が作られなかった場合、つまり点パターン X 内の点が “nil” に対応づけられた場合には、そのノードが表すマッチングは親ノードが表すマッチングから変化しないため、当然その適格範囲も変化しないことになり、この場合、常にこのような対応は成功することになる。

探索中に大きさ $N + 1$ 以上のマッチング Φ が上記の条件に合格した場合には、このマッチングの最適変換 T_Φ および (3.2) 式で定義される最大誤差 d を求め、このマッチングが実際に δ 制限であるか否かをテストする。この際、最適変換 T_Φ のスケール c が $EI^m(\Phi)$ に含まれているとは限らないことに注意する。もし、 c が $EI^m(\Phi)$ に含まれていないならば、最大誤差を求めるまでもなく、当然このマッチングは δ 制限ではありえない。もし Φ がこのテストに合格した場合には、 Φ は MMリストに現在登録されているマッチングより大きな δ 制限されたマッチングであるから、MMリストの現在の内容を廃棄し、 Φ を複製して MMリストの一番目の要素として登録する。 N をそのマッチングの大きさに更新する。また一方、もし大きさ N のマッチングが上記



(a) Point pattern X

(b) Point pattern Y

図 3.6: マッチングに使用された点パターン

の条件に合格した場合には、大きさ $N + 1$ 以上の場合のように最大誤差 d を用いた δ 制限性のテストを行なうことなく、ただちにMMリストに登録しておく。これは、もしこのマッチングより大きなマッチングが後の探索で見つされた場合に、このマッチングが廃棄され、テストの手間が無駄になるのを防ぐためである。

最後にすべてのノードの探索を終え、もはやより大きなマッチングが見つかりなかった段階で、MMリストに登録されているマッチングの δ 制限性をすべてテストし、この最終テストに合格したマッチングを問題の解として出力する。MMリストに登録されているマッチングの最初の要素はそこに登録される時点でその δ 制限性をテストされているため、少なくとも一つは解が存在する。

3.3.5 実行例

提案するマッチングアルゴリズムの理解を深めるために簡単な計算機実験を示す。乱数を用いて二つの二次元点パターン X 、 Y を生成し、それらを提案するマッチング手法を用いて互いにマッチングさせる。図 3.6(a)(b) が生成された点パターンである。点の数はそれぞれ 10 個づつであり、点の座標は X も Y も -1.0 から 1.0 の範囲にある。

図 3.7はこれらを提案するアルゴリズムで互いにマッチングさせた結果である。図では、最適交換で移動させたパターン X をパターン Y の上へ重ねて示してある。重ね合わされた結果 -1.0 と 1.0 の範囲からはみだした点については表示していない。図に示されているのはいくつか得られた δ 制限マッチングのうち、最大誤差 d が最小のものである。図の左側には、得られたマッチングの大きさ、個数、および探索木の大きさ（図 3.5のアルゴリズムにおいて、ONlist に保存されたノードの延べ数、言い換えれば、 $EJ^m(\Phi) \neq \Phi$ となったノード数）とすべてのマッチングを生成するのに要した時間を示してあり、また、図の下側には得られた解（ δ 制限マッチング）のリスト（平均最小二乗誤差 e^2 , 最大誤差 d , 最適交換のパラメータ、および対応結果）を示してある。解は最大誤差の大きさの順序に並べられている。ここで対応結果は、点パターン X 内の点に対応する点パターン Y の点を順に示してあり、記号 "*" はその点 Y に対応点を持たないことを示している。つまり、図 3.7 (a) の最初の解で、A D C * * ... となっているのは、 X の点 a, b, c が Y の点 A, D, C に対応し、点 d, e が対応点を持たないことを示している。

図 3.7(a) は $\delta = 0.1$ の場合、図 3.7(b) は $\delta = 0.09$ の場合のマッチング結果である。どちらの場合も得られるマッチングの大きさは 5 であり、全部で $\delta = 0.1$ の場合 9 個、 $\delta = 0.09$ の場合 3 個のマッチングが得られている。(b) の場合の 3 個の解は (a) の最初の 3 個の解に一致しており、 δ を 0.1 から 0.09 に制限することにより、得られるマッチングが制限されていることがわかる。すべての解を生成するのに要した時間は (a) の場合で 4.12 秒、(b) の場合で 3.71 秒であった。実験は SUN4/280 上で行った。

(a)

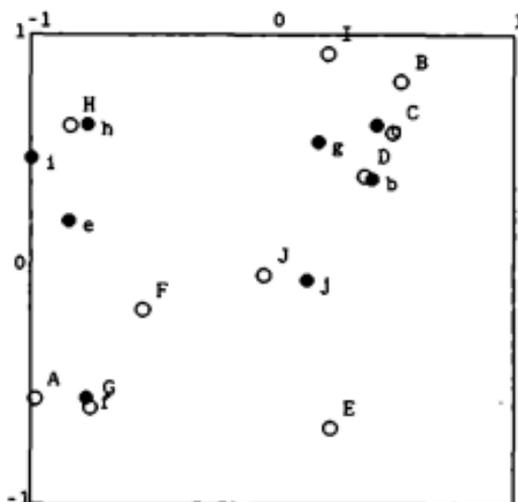
 $\delta = 0.10$

SIZE = 5

No. of solutions = 9

No. of nodes = 9263

time = 4.12 (Sec.)



- (1) $a = 0.004$ $d = 0.074$:: $c = 0.893$ $b = (-0.311, 0.286)$ $R = 84.796$
 A D C * * G * H * * *
 (2) $a = 0.004$ $d = 0.085$:: $c = 1.134$ $b = (-0.279, 0.416)$ $R = -134.639$
 * * G * I * F D B *
 (3) $a = 0.005$ $d = 0.090$:: $c = 0.841$ $b = (-0.161, 0.610)$ $R = 75.981$
 * B * * * F I * E D
 (4) $a = 0.005$ $d = 0.090$:: $c = 0.833$ $b = (-0.178, 0.554)$ $R = 72.690$
 G B * * * I * E D
 (5) $a = 0.004$ $d = 0.091$:: $c = 0.703$ $b = (-0.295, 0.421)$ $R = -70.769$
 B H * D * I * J * *
 (6) $a = 0.004$ $d = 0.093$:: $c = 0.766$ $b = (-0.401, 0.526)$ $R = -60.661$
 B * * D * I E * J *
 (7) $a = 0.004$ $d = 0.094$:: $c = 0.951$ $b = (-0.158, 0.577)$ $R = -131.045$
 * * F * I * * C B E
 (8) $a = 0.005$ $d = 0.097$:: $c = 1.167$ $b = (-0.266, 0.451)$ $R = -135.850$
 * * G * I * * F C B *
 (9) $a = 0.004$ $d = 0.099$:: $c = 0.895$ $b = (-0.205, -0.318)$ $R = -90.748$
 C * A * * D G E * *

図 3.7: マッチング結果

(b)

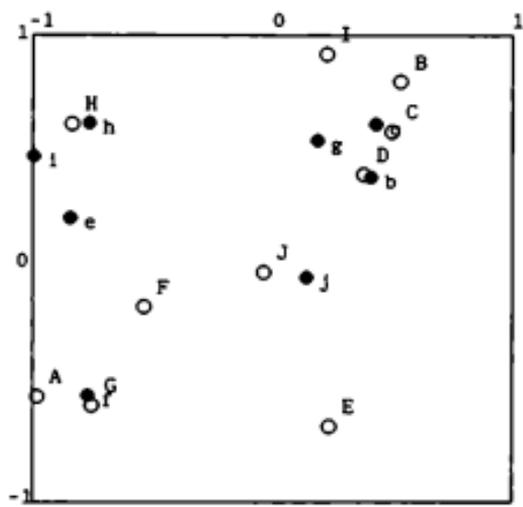
$\delta = 0.09$

SIZE = 5

No. of solutions = 3

No. of nodes = 8354

time = 3.71 (Sec.)



- (1) * = 0.004 d = 0.074 :: c = 0.893 b = (-0.311, 0.286) R = 84.796
 A D C * * G * E * *
 (2) * = 0.004 d = 0.085 :: c = 1.134 b = (-0.279, 0.416) R = -134.639
 * * G * I * F D B *
 (3) * = 0.005 d = 0.090 :: c = 0.841 b = (-0.161, 0.610) R = 75.981
 * B * * * F I * E D

图 3.7: (續前)

3.4 さまざまな条件、制約下でのマッチング

3.3節においては、最も一般的な場合における点パターンマッチング問題について考察した。しかし、現実の問題においては、さまざまな条件、制約等がマッチングに際して存在することも多く、それらに対して柔軟に対処できることも、手法の有効性の点から重要である。この節および次節ではより特殊な条件、制約下での点パターンマッチング問題について考察する。

3.4.1 鏡像パターン

今まで述べたマッチングにおいては、パターン間の関係として相似変換を仮定し、互いに鏡像関係にあるようなパターンについてはまったく考慮してこなかった。この場合には、(裏返すことにより)たとえ完全にぴったりと重なりあうようなパターンがあったとしても、それらは対応しないものとして最初から排除することになる。三次元の物体を裏返すような変換は現実には不可能であるから、三次元のパターンに対しては、これは十分現実的な仮定である。しかし、二次元のパターンを考えると、この場合には同一の物体が裏返しに置かれている場合なども考えられるため、鏡像関係にあるようなパターンも含めて、 δ 制限された最大マッチングを一度に探索可能な手法を考えることは、興味ある問題であると思われる。

鏡像パターンまで考慮してマッチングを行なうということは、両パターン内の点を関係づける (3.1) 式において、 R の定義域を回転行列の範囲から直交行列の範囲まで拡大して考えるということに他ならない。このように R の定義域を拡大したとしても、3.3節での議論は、ほとんど平行に進めることが可能である。この意味で提案したマッチング手法を鏡像パターンを考慮するように変形することは、非常に容易かつ自然であり、図 3.5 に示すアルゴリズムがほとんどそのまま適用可能である。ただ一点、変更を要する点は、 $I(\Phi)$ の計算を行なう際に、(3.32) 式の S を、 $\det(\Sigma_{xy})$ の正負に依らず常に

$$S = I \quad (3.68)$$

とする点だけである。 R の定義域を直交行列の範囲まで拡張して考える場合に、 S がこのように定められることは、定理 3.2 の証明から明らかである。

3.4.2 変換パラメータに対する制約

3.3節で提案したマッチングにおいては、パターン間の変換パラメータについてどのような制限も置いていなかった。しかし、実際の応用場面ではこれについて自然な制限が存在する場合がある。例えば、物体のステレオ視においては（二つのカメラを水平に並べるため）二つの画像間の垂直方向のずれは考慮しなくとも良いことが多いし、またモデルに基づいた3D物体認識においても、その認識対象物が単独で平面上に置かれているような場合には、対象物の形状からその安定位置が幾つかに制限される場合が考えられる。このような場合、変換の平行移動パラメータ、回転パラメータはなんらかの制約を受けることになる。

本論文のマッチング手法は、容易にこれらの制約を利用し、探索木の枝刈りに利用することができる。あらかじめ変換パラメータの基準値、あるいはそのなんらかの拘束条件が与えられた場合、探索中の各部分マッチングの変換パラメータを(3.51) (3.52) (3.53)式から求め、このパラメータを与えられた条件と比較して、条件を満足しないマッチングを排除することにすればよい。これにより拘束条件を満足するマッチングのみを生成することが可能となる。

このような変換パラメータに対する制約による探索の制御は、マッチングを高速化するためにも極めて有用である。ただし、このように探索中の各部分マッチングの変換パラメータをそのつど求め、このパラメータを与えられた条件と比較して、条件を満足しないマッチングは排除するという手法から最終的に得られるマッチングは、もはや δ 制限の意味で最大のマッチングとは限らないことに注意しなければならない。またこの場合、最大のマッチングがすべて数え上げられることも保証できない。これは、与えられた幾何学的制約を満足するようなマッチングのすべての部分マッチングが、やはりその制約を満足するとは限らないからである。ある部分マッチングが与えられた制約を満足さないために枝刈りが行なわれ、その部分マッチングを拡大することによって得られるはずであった解が得られなくなるということは起りえる。

次に、与えられた条件がスケールに関するものである場合には、適格範囲という条件がすでに利用されているため、特別な取扱いが可能である。例えば、もし変換のスケールが区間

$$E I_{init} = [c_{min}, c_{max}] \quad (3.69)$$

の内部になければならないとすると、(3.17) 式の $EI''(\Phi)$ は探索の最初から次のように制限されることになる。

$$EI''(\Phi) = EI_l(\Phi) \cap EI_g(\Phi) \cap EI_{int} \quad (3.70)$$

このためには、単に (3.22) 式において、

$$EI''(\Phi_0) = EI_{init} \quad (3.71)$$

と定めるだけで良く、非常に簡単であり、後の計算にはまったく負担がかからない。また、この場合には、スケールに関する条件が、スケールの適格範囲という条件にうまく埋め込まれているため、先の場合のように最大のマッチングを取りこぼすということも無い。ただしこの場合も、たとえマッチング Φ について $EI''(\Phi) \neq \emptyset$ であったとしても、その Φ から導かれる最適相似変換のスケールが初期制約 EI_{init} の内部にある保証は無いため、アルゴリズム内で実際に Φ の δ 制限性をテストする際には、このスケールが確かに区間 EI_{init} 内に入っていることを調べておく必要がある。

3.4.3 実行例

3.3.5節で用いたのと同じ点パターンの組 (図 3.6) に対して、さまざまな条件、制約を与えてマッチングを行なう。

図 3.8(a) は $\delta = 0.09$ として、鏡像となるパターンも考慮してマッチングを行なった結果である。大きさ 5 の解が 11 個得られている。そのうち鏡像パターン間のマッチングとなっているのは、8 個であり (解の回転パラメータの項に "*" が付けられている)、残り 3 個は図 3.7 (b) でも得られている通常のマッチングである。すべての解を生成するのに要した時間は 4.75 秒であった。通常のマッチングを行なうのに必要な計算時間は 3.71 秒であり、一方の点パターンを裏返してマッチングを行なうのに要する時間は、別の実験を行なった結果 3.42 秒であったので、それぞれを個別に行なった場合 ($3.71 + 3.42 = 7.13$ 秒) に比較して、鏡像パターンを考慮したマッチングにおいては、効率的な処理が実現されていることがわかる。

図 3.8(b)(c) は変換パラメータに何らかの制約が課せられた場合の例である。いずれの場合にも $\delta = 0.10$ とし、図 3.8(b) の場合には、スケールの範囲をあらか

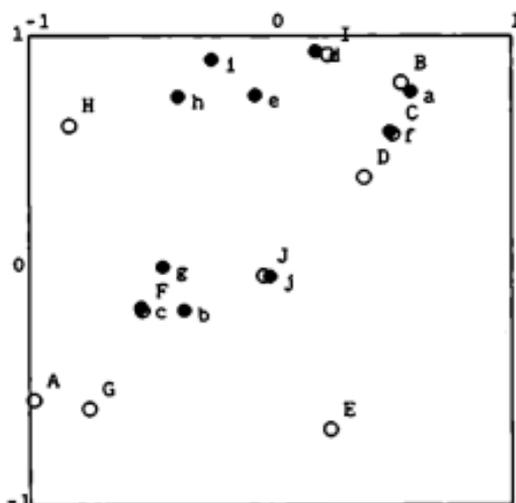
じめ $[c_{min}, c_{max}] = [0.7, 0.8]$ と与え、一方 図 3.8(c) の場合には回転の角度範囲を $[R_{min}, R_{max}] = [30.0, 50.0]$ (degrees) と定めた。明らかに図 3.8(a) の場合には、図 3.7(a) で得られた解のうち、スケールに対する制約を満足する解 2 個が選択的に得られている。一方、図 3.8 (b) の場合には、与えられた回転に関する制約を満足する大きさ 5 の解が存在しないため、大きさ 4 の解が 15 個得られている。それぞれ、計算時間は 1.22、10.91 秒であった。

(a)

$\delta = 0.09$
Mirror = 1

SIZE = 5
No. of solutions = 11

No. of nodes = 10646
time = 4.75 (Sec.)



- (1) * = 0.001 d = 0.052 :: c = 0.678 b = (-0.191, 0.362) R = -14.999
B * F I * C * * * J
- (2) * = 0.001 d = 0.055 :: c = 0.895 b = (-0.529, 0.105) R = -15.933
C G A * * D * H * *
- (3) * = 0.001 d = 0.060 :: c = 0.727 b = (0.071, 0.461) R = +126.740
H B * * * C J * I
- (4) * = 0.003 d = 0.067 :: c = 0.718 b = (-0.260, 0.389) R = -18.971
B F * I * C * * * J
- (5) * = 0.004 d = 0.074 :: c = 0.893 b = (-0.311, 0.286) R = 84.796
A D C * * G * H * *
- (6) * = 0.003 d = 0.080 :: c = 0.968 b = (-0.515, 0.172) R = -12.161
B G A * * C * H * *
- (7) * = 0.005 d = 0.081 :: c = 1.060 b = (-0.153, 0.413) R = -78.744
* * * C * H I B F
- (8) * = 0.004 d = 0.085 :: c = 1.134 b = (-0.279, 0.416) R = -134.639
* * G * I * F D B *
- (9) * = 0.003 d = 0.086 :: c = 0.932 b = (-0.530, 0.140) R = -13.479
B G A * * D * H * *
- (10) * = 0.003 d = 0.086 :: c = 0.977 b = (0.092, 0.088) R = +168.549
A I B * * * C * E *
- (11) * = 0.005 d = 0.090 :: c = 0.841 b = (-0.161, 0.610) R = 75.981
* B * * * F I * H D

図 3.8: さまざまな条件、制約下でのマッチング結果

(b)

$\delta = 0.10$

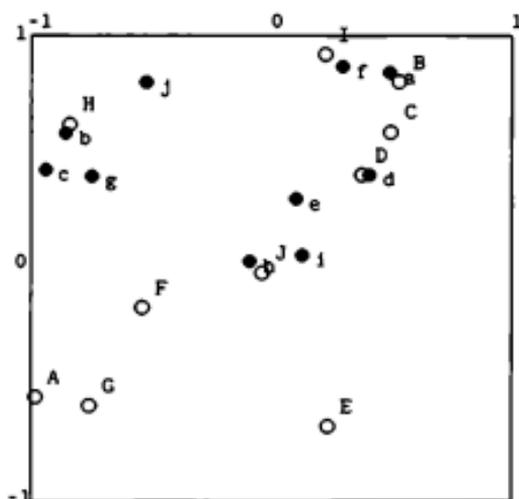
$[c_{min}, c_{max}] = [0.7, 0.8]$

SIZE = 5

No. of solutions = 2

No. of nodes = 2764

time = 1.22 (Sec.)



(1) * = 0.004 d = 0.091 :: c = 0.703 b = (-0.295, 0.421) R = -70.769

B H * D * I * J * *

(2) * = 0.004 d = 0.093 :: c = 0.755 b = (-0.401, 0.526) R = -60.661

B * * D * I H * J *

図 3.8: (続々)

(c)

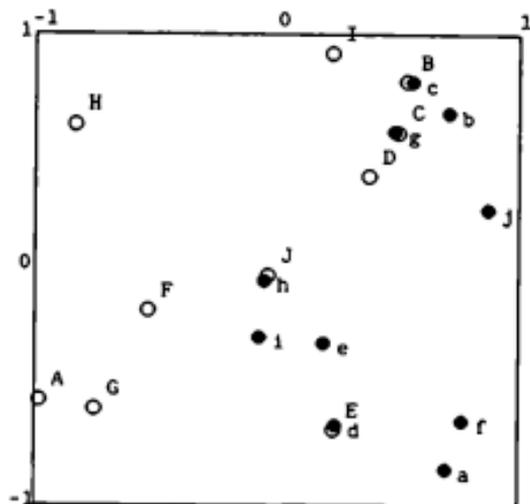
 $\delta = 0.10$ $[R_{min}, R_{max}] = [30.0, 50.0]$

SIZE = 4

No. of solutions = 15

No. of nodes = 8831

time = 10.91 (Sec.)



| | | | | | |
|------|-----------------------|-----------|-----------|-----------------------|------------|
| (1) | e = 0.000 | d = 0.027 | c = 0.777 | b = (0.417, 0.104) | R = 30.646 |
| | * * B E | * * C J | * * | | |
| (2) | e = 0.002 | d = 0.055 | c = 0.693 | b = (0.264, 0.230) | R = 48.906 |
| | E * B * J * C * | * * | | | |
| (3) | e = 0.002 | d = 0.060 | c = 0.953 | b = (-0.183, 0.272) | R = 34.747 |
| | * I * * F E * * D | * * | | | |
| (4) | e = 0.003 | d = 0.076 | c = 0.763 | b = (-0.014, 0.242) | R = 32.751 |
| | E * I * * * F D | * * | | | |
| (5) | e = 0.004 | d = 0.081 | c = 1.050 | b = (0.074, -0.226) | R = 47.639 |
| | * * C * * * D F G * | | | | |
| (6) | e = 0.005 | d = 0.081 | c = 1.182 | b = (0.157, -0.148) | R = 43.619 |
| | * * B * * * C F G * | | | | |
| (7) | e = 0.005 | d = 0.083 | c = 1.124 | b = (0.111, -0.182) | R = 44.617 |
| | * * B * * * D F G * | | | | |
| (8) | e = 0.004 | d = 0.089 | c = 0.578 | b = (0.263, 0.059) | R = 44.864 |
| | E * C * * * D J * | | | | |
| (9) | e = 0.005 | d = 0.092 | c = 0.811 | b = (0.100, 0.125) | R = 46.874 |
| | * C B * * E * * F * | | | | |
| (10) | e = 0.004 | d = 0.092 | c = 0.670 | b = (0.212, 0.222) | R = 45.717 |
| | E C B * * J * * * | | | | |
| (11) | e = 0.006 | d = 0.096 | c = 0.860 | b = (-0.324, 0.754) | R = 32.863 |
| | * * * F * J * B * I | | | | |
| (12) | e = 0.006 | d = 0.096 | c = 1.188 | b = (-0.194, -0.105) | R = 44.215 |
| | * C I * * G * * A * | | | | |
| (13) | e = 0.005 | d = 0.097 | c = 0.861 | b = (0.029, 0.196) | R = 39.387 |
| | * B I * * * E * * F * | | | | |
| (14) | e = 0.004 | d = 0.098 | c = 0.585 | b = (0.233, 0.122) | R = 49.366 |
| | E * C * J * D * * * | | | | |
| (15) | e = 0.008 | d = 0.100 | c = 0.886 | b = (-0.115, 0.237) | R = 39.320 |
| | * * I * F E * * * D | | | | |

図 3.8: (続き)

3.5 点パターン間の全体マッチング問題

ここまで扱ってきたマッチング問題は、二つの点パターンの大きさが異なっており、また最終的に得られるマッチングの大きさも事前にはわからないという最も一般的な場合であった。この節では、点パターンマッチング問題の特別な場合として、二つの点パターンの大きさが等しく、かつすべての点が (δ 制限の意味で) 対応点を持つような場合を考える。このようなマッチング問題を全体マッチング問題と呼ぶ。全体マッチング問題においては、点パターンを X 、 Y として、 $|X| = |Y|$ であり、 X を Y 上へ 対応点間の平均二乗誤差が最小となるように重ね合わせた時、対応点間の距離が δ 以下となる X 、 Y のすべての点を対応づけるマッチング (全マッチングと呼ぶ) ϕ を決定することが問題となる。

Baird [5] が取り扱っているのは、解の定義などの点では我々と異なった部分もあるが、大枠としてはこのような問題である。しかし、まえがきでも述べたように、彼の手法を三次元以上の場合に拡張することは困難であり、その意味で、この節でこのような場合を特別に考慮することは重要であると思われる。

全体マッチング問題の場合にもアルゴリズムの大枠に変更は無い。また、前節までに示した一般的なマッチング手法をそのままこの問題に適用することも、もちろん可能である。しかし、全体マッチング問題の場合には、全ての点はその対応点を持つという制約から、木探索に際してより強力な枝刈りを行なうことが可能であり、例えば、 X 内のある点が Y 内に対応点を持たない場合などを考慮しなくても良いという一点からだけでも、マッチングの探索空間は非常に小さなものとなる。この節では、全体マッチング問題に特有の枝刈り法を考察する。

3.5.1 重心点一致戦略

全体マッチング問題の場合には、点パターン内のすべての点が対応づけられるという制約から、対応がどのようにつけられるにせよ、対応点間の平均二乗誤差を最小にするような相似変換を適用した場合、両パターンはその重心が常に一致するように重ね合わされる。

マッチングを行なう点パターンを $X = \{x_1, x_2, \dots, x_N\}$ 、 $Y = \{y_1, y_2, \dots, y_N\}$

とする。 ($|X| = |Y| = N$) これらのパターンを、それぞれの重心 (平均 μ_x, μ_y) が原点となるように平行移動した点パターン $X' = \{x'_1, x'_2, \dots, x'_N\}$ 、 $Y' = \{y'_1, y'_2, \dots, y'_N\}$ を考える。

$$x'_i = x_i - \mu_x \quad (3.72)$$

$$y'_i = y_i - \mu_y \quad (3.73)$$

この時、明らかに X' 、 Y' の重心 (μ'_x, μ'_y) は

$$\mu'_x = 0 \quad (3.74)$$

$$\mu'_y = 0 \quad (3.75)$$

である。そこで、 X' 、 Y' 間の任意の全マッチングに対し、それから導かれる両パターン間の最適相似変換の平行移動ベクトル t' は (3.52) 式から

$$t' = \mu'_y - cR\mu'_x = 0 \quad (3.76)$$

となる。つまり、全マッチングにおいては、 X' 、 Y' の原点、言い換えれば X 、 Y の重心は常に一致するように、相似変換が決定される。このことから、 X 、 Y 間の全マッチングを考えるには、それぞれその重心を原点まで移動した X' 、 Y' を考え、また両パターンを関連づける変換としてスケール変換と回転のみを考えてマッチングを行えば良いことが判る。以下では、点パターン X 、 Y の重心があらかじめ原点に一致させられているとして議論を進める。

3.5.2 スケールの適格範囲による枝刈り

全体マッチング問題においても、スケールの適格範囲に関する議論は、変換において平行移動を考慮しなくとも良いという点を除いて、3.3節とほとんど同様に行なうことができる。ただし、変換としてスケール変換と回転しか考えないという点から、次に述べるような変更が必要である。

まず、マッチング ϕ ($\text{size}(\phi) = n$) のスケールに関する適格範囲 $EI_c(\phi)$ は次にように定められる。

$$EI_c(\phi) = \{c \mid \exists R \max_{(x,y) \in \phi} \|y - cRx\| \leq \delta\} \quad (3.77)$$

一般の場合と同様に、この場合にも 3.3 節における性質 1、性質 2、性質 3 が成立する。さらに、最小平均二乗誤差から導かれるスケールの範囲 $I_1(\Phi)$ を次のように定義する。

$$I_1(\Phi) = \{c \mid \min_R \frac{1}{n} \sum_{(x,y) \in \Phi} \|y - cRx\|^2 \leq \delta^2\} \quad (3.78)$$

これについても、定理 3.1 が同様に成立する。また、この範囲を実際に計算することも容易であり、これは定理 3.2 において $\mu_x = \mu_y = 0$ とすることにより、一般のマッチングの場合と同じく (3.35) 式で与えることができる。ただし、この場合には定理 3.2 の (3.28) 式、(3.29) 式、(3.30) 式はそれぞれ次のように定まる。

$$\sigma_x^2 = \frac{1}{n} \sum_{i=1}^n \|x_i\|^2 \quad (3.79)$$

$$\sigma_y^2 = \frac{1}{n} \sum_{i=1}^n \|y_i\|^2 \quad (3.80)$$

$$\Sigma_{xy} = \frac{1}{n} \sum_{i=1}^n y_i x_i^T \quad (3.81)$$

以上の準備のもとで、全体マッチングの場合の適格範囲の近似 $EJ_1''(\Phi)$ を次のように定義する。

$$EJ_1''(\Phi) = EJ_H(\Phi) \cap EJ_{g1}(\Phi) \cap EJ_{init}(X, Y) \quad (3.82)$$

$$EJ_H(\Phi) = \bigcap_{\Phi' \subset \Phi \text{ and size}(\Phi')=1} I_1(\Phi') \cap \bigcap_{\Phi' \subset \Phi \text{ and size}(\Phi')=2} I(\Phi') \quad (3.83)$$

$$EJ_{g1}(\Phi) = \bigcap_{i=0}^n I_1(\Phi_i) \quad (3.84)$$

一般のマッチングの場合と同様に $EJ_H(\Phi)$ はスケールの局所的な制約を、 $EJ_{g1}(\Phi)$ は大域的な制約を表現している。また、 $EJ_{init}(X, Y)$ は、すべての点が対応点を持つという制約から導かれる初期制約であり、次節で詳述する。

さてここで、 $EJ_H(\Phi)$ の第一項は大きさ 1 のマッチングから導かれる制約である。一般には、大きさ 1 のマッチング $\Phi_1 = \{(x, y)\}$ について $I(\Phi_1) = [0, \infty)$ であり、スケールに対して何の条件も与えないため、これを適格範囲の近似として用いることは無意味であった。ところが、全体マッチングの場合には、パターン間の平行移

動を考慮しなくとも良いという制約から、大きさ 1 のマッチングに対しても $I_t(\Phi_1) \neq [0, \infty)$ となり、これを枝刈りに利用することが可能となる。実際、(3.79)(3.80)(3.81) 式に注意して定理 3.2 から $I_t(\Phi_1)$ を求めると、

$$I_t(\{(x, y)\}) = [c_{\min}, c_{\max}] \quad (3.85)$$

$$c_{\min} = \max(0, \frac{\|y\| - \delta}{\|x\|}) \quad (3.86)$$

$$c_{\max} = \frac{\|y\| + \delta}{\|x\|} \quad (3.87)$$

となる。

次に $EJ_t(\Phi)$ の第二項は大きさ 2 のマッチングから導かれる制約である。一般のマッチングの場合、大きさ 2 のマッチング Φ_2 に対する $I(\Phi_2)$ の計算は各点相互間の距離から簡単に行なうことができたが、全体マッチングの場合には（平行移動を考慮しないという制約から）、 $I_t(\Phi_2)$ を求めるには大きさ 3 以上のマッチングと同様な特異値計算を必要とする。 $I_t(\Phi_2)$ の計算は、たとえ漸進的な計算法を採用したとしても、マッチングの大きさに比例した回数だけ各ノードにおいて行なう必要があるため、これでは負担が大きすぎると思われる。そのため、ここでは $I_t(\Phi_2)$ を、平行移動まで考慮した範囲である $I(\Phi_2)$ ((3.64) 式) で代行する。明らかに

$$I(\Phi_2) \supseteq I_t(\Phi_2) \quad (3.88)$$

であるから、このように代行することは安全である。

一般のマッチングの場合と同様に、 $EJ_t''(\Phi)$ も次のようにマッチングを拡大していくにつれて漸進的に計算可能である。 $\Phi_0 (= \phi), \Phi_1, \dots, \Phi_{n-1}, \Phi_n, \dots$ を、木探索につれて次第に拡大されるマッチングの列であるとし、 $\Phi_n = \Phi_{n-1} \cup \{(x, y)\}$ とすると、

$$EJ_t''(\Phi_0) = EJ_{\text{init}}(X, Y) \quad (3.89)$$

$$EJ_t''(\Phi_n) = EJ_t''(\Phi_{n-1}) \cap I_t(\Phi_n) \cap I_t(\{(x, y)\}) \\ \cap \bigcap_{(x', y') \in \Phi_{n-1}} I(\{(x', y'), (x, y)\}) \quad (3.90)$$

となる。

3.5.3 運格範囲に対する初期制約

マッチングを行なう点パターンを $X = \{x_1, x_2, \dots, x_N\}$ 、 $Y = \{y_1, y_2, \dots, y_N\}$ とし、 X 、 Y を $m \times N$ 行列 $X = [x_1, x_2, \dots, x_n]$ 、 $Y = [y_1, y_2, \dots, y_n]$ で表現する。さて、最終的に得られる δ 制限全マッチングを $\hat{\phi}$ とし、これに対応する順列行列を \hat{P} とすると、(3.78) 式は

$$I_t(\hat{\phi}) = \{c \mid \min_R \frac{1}{N} \|Y - cRX\hat{P}\| \leq \delta^2\} \quad (3.91)$$

となる。ところで、 \mathcal{P} を順列行列全体の集合、 \mathcal{R} を回転行列全体の集合、また \mathcal{Q} を直交行列全体の集合とすると、

$$\mathcal{P} \subset \mathcal{Q} \quad (3.92)$$

$$\mathcal{R} \subset \mathcal{Q} \quad (3.93)$$

であるから、

$$\begin{aligned} \min_{R \in \mathcal{R}} \frac{1}{N} \|Y - cRX\hat{P}\| &\geq \min_{P \in \mathcal{P}} \min_{R \in \mathcal{R}} \frac{1}{N} \|Y - cRX\hat{P}\| \\ &\geq \min_{Q_1 \in \mathcal{Q}} \min_{Q_2 \in \mathcal{Q}} \frac{1}{N} \|Y - cQ_1XQ_2\| \end{aligned} \quad (3.94)$$

となる。よって、スケールの範囲 $EJ_{init}(X, Y)$ を次のように定義する時

$$EJ_{init}(X, Y) = \{c \mid \min_{Q_1 \in \mathcal{Q}} \min_{Q_2 \in \mathcal{Q}} \frac{1}{N} \|Y - cQ_1XQ_2\| \leq \delta^2\} \quad (3.95)$$

任意の δ 制限全マッチング $\hat{\phi}$ について

$$EJ_{init}(X, Y) \supseteq I_t(\hat{\phi}) \quad (3.96)$$

が成立する。あるマッチング ϕ が将来 δ 制限全マッチング $\hat{\phi}$ に拡大されるとすると、当然

$$I_t(\hat{\phi}) \cap EJ_t''(\phi) \neq \emptyset \quad (3.97)$$

つまり

$$EJ_{init}(X, Y) \cap EJ_t''(\phi) \neq \emptyset \quad (3.98)$$

でなければならない。このことから、 $EI_{init}(X, Y)$ を、適格範囲に対する初期制約として用いることができる。

$EI_{init}(X, Y)$ の実際の計算は次のように行なう。まず定理 3.4 [9] を与える。

定理 3.4 (von Neumann) $m \times n$ ($m \leq n$) 複素行列 A, B がそれぞれ特異値 $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_m \geq 0, \beta_1 \geq \beta_2 \geq \dots \geq \beta_m \geq 0$ を持つとする。この時、任意のユニタリ行列 V ($n \times n$)、 W ($m \times m$) に対して

$$\operatorname{Re}\{\operatorname{tr}(AVB^TW)\} \leq \sum_{i=1}^m \alpha_i \beta_i \quad (3.99)$$

が成立する。

この定理から次の定理が得られる。

定理 3.5 X, Y を $m \times N$ 行列で、特異値 $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_m \geq 0, \beta_1 \geq \beta_2 \geq \dots \geq \beta_m \geq 0$ を持つとする。この時、

$$e^2(Q_1, Q_2, c) = \frac{1}{N} \|Y - cQ_1 X Q_2\|^2 \quad (3.100)$$

の、直交行列 Q_1, Q_2 に対する最小値 $e^2(c)$ は、

$$e^2(c) = \sigma_y^2 - 2\frac{c}{N} \sum_{i=1}^m \alpha_i \beta_i + c^2 \sigma_x^2 \quad (3.101)$$

で与えられる。ただし、 σ_x^2, σ_y^2 は(3.79)(3.80)式で定義される。

証明： (3.100) 式を展開すると

$$e^2(Q_1, Q_2, c) = \sigma_y^2 - 2\frac{c}{N} \operatorname{tr}(Y Q_2^T X^T Q_1^T) + c^2 \sigma_x^2 \quad (3.102)$$

となる。ここで、定理 3.4 より、任意の直交行列 Q_1, Q_2 に対して

$$\operatorname{tr}(Y Q_2^T X^T Q_1^T) \leq \sum_{i=1}^m \alpha_i \beta_i \quad (3.103)$$

が成立するため

$$e^2(Q_1, Q_2, c) \geq \sigma_y^2 - 2\frac{c}{N} \sum_{i=1}^m \alpha_i \beta_i + c^2 \sigma_x^2 \quad (3.104)$$

である。実際、 X 、 Y の特異値分解を

$$X = U_X D_X V_X^T \quad (3.105)$$

$$Y = U_Y D_Y V_Y^T \quad (3.106)$$

として、

$$\hat{Q}_1 = U_Y U_X^T \quad (3.107)$$

$$\hat{Q}_2 = V_X V_Y^T \quad (3.108)$$

とすると、

$$e^2(\hat{Q}_1, \hat{Q}_2, c) = \sigma_v^2 - 2 \frac{c}{N} \sum_{i=1}^m \alpha_i \beta_i + c^2 \sigma_x^2 \quad (3.109)$$

が成立する。ゆえに、 $e^2(Q_1, Q_2, c)$ の Q_1 、 Q_2 に対する最小値は (3.101) 式で与えられる。(証明終)

定理 3.5 より、 $EJ_{init}(X, Y)$ は次式を満たす c の範囲として求まる。

$$e^2(c) = \sigma_v^2 - 2 \frac{c}{N} \sum_{i=1}^m \alpha_i \beta_i + c^2 \sigma_x^2 \leq \delta^2 \quad (3.110)$$

この式から、

$$EJ_{init}(X, Y) = \begin{cases} \phi & \text{if } \mathcal{D}' < 0 \\ [c_{\min}, c_{\max}] & \text{if } \mathcal{D}' \geq 0 \end{cases} \quad (3.111)$$

となる。ここで、

$$\mathcal{D}' = \left(\frac{1}{N} \sum_{i=1}^m \alpha_i \beta_i \right)^2 - \sigma_x^2 (\sigma_v^2 - \delta^2) \quad (3.112)$$

$$c_{\min} = \max \left\{ 0, \frac{1}{\sigma_x^2} \left(\frac{1}{N} \sum_{i=1}^m \alpha_i \beta_i - \sqrt{\mathcal{D}'} \right) \right\} \quad (3.113)$$

$$c_{\max} = \frac{1}{\sigma_x^2} \left(\frac{1}{N} \sum_{i=1}^m \alpha_i \beta_i + \sqrt{\mathcal{D}'} \right) \quad (3.114)$$

である。

3.6 計算機実験による点パターンマッチングアルゴリズムの評価

提案する点パターンマッチングアルゴリズムの効率の評価を理論的に行なうことは極めて困難であるため、計算機実験によってこれを行なった。二次元および三次元空間内の大きさ(点数)の等しい点パターン対 X, Y を計算機によって人工的に生成し、それらに対して提案しているアルゴリズムを適用してそのマッチングに要する計算時間などを測定した。通常の点パターンマッチングの場合(すべての点に対応点を持つとは限らない場合)および全体マッチングの場合(すべての点とその対応点を持つ場合)について実験を行ない、探索木の大きさや平均計算時間などを調べた。前者の実験を実験1と呼び、後者を実験2と呼ぶ。

3.6.1 実験方法

実験1 (通常のマッチング)

点パターン対 X, Y は、共に半径 $\rho = 1.0$ の円(二次元の場合)あるいは球(三次元の場合)内に一様かつ独立に n 個の点を選ぶことにより作成した。マッチングの許容誤差を $\delta = 0.05, \delta = 0.1, \delta = 0.2$ として、さまざまな大きさの点パターン対 100 組について提案する点パターンマッチングアルゴリズムを適用し、探索木の大きさ、平均計算時間などを求めた。最終的にどれだけの大きさのマッチングが得られるかは、実験に際して仮定していない。

実験2 (全体マッチング)

この場合には Baird [5] に従って次のように点パターン対を作成した。まず点パターン Y として、半径 $\rho = 1.0$ の円(二次元の場合)あるいは球(三次元の場合)内に一様かつ独立に n 個の点を選ぶ。次にこの点パターンの各点を中心にした半径 δ の円(球)内に一個づつ点を選び、点パターン Y に“ノイズ”の加わった点パターンを作成する。この点パターンを回転、平行移動、スケール変換し、最後に適当に点の番号を入れ替えて、それを点パターン X とする。ここで、平行移動ベクトルは原点を中心とした半径 0.5 の円(球)内から一様を選び、回転角は $[0, 2\pi]$ (三次元の場合には x, y, z のそれぞれの回転軸について)のうちから一様を選ぶ。またスケール c は

$\log c$ が $[\log \frac{1}{3}, \log 3]$ のうちで一樣に選ばれるように決定する。この結果、スケール c は $[\frac{1}{3}, 3]$ の範囲に入ることになる。

δ 制限の定義では、パターン間の重ね合わせを最小二乗誤差を最小にするような相似変換で行なうため、このようにして得られた点パターン対が δ 制限な全マッチングを持つとは限らない。このため、このように構成された点パターン対が全体として実際に δ 制限であるか否かを、あらかじめ判っている点の対応関係に基づいて調べ、 δ 制限な全マッチングが存在するような点パターン対のみを実験に用いる。

最後にノイズの大きさ δ であるが、[5] に従って、次のように決定する。二次元の場合について説明する。まず点パターン Y 内の二点でその間の距離が 2δ 以下であるようなものを互いに “confusable” であるとよぶ。これはこのような二点を中心とする半径 δ の円が互いに重なりあうことになるため、マッチングさせた時その重なった部分に来るような点をどちらに対応するか判断できかねる状況が生じるためである。 ρ が一定とすると、 δ を一定に保ったままパターンのサイズ n を大きくしていった場合、このような “confusable” な点対も次第に増えていくことになる。そこで、ノイズの影響を n によらず一定とするため、パターン Y のある点と互いに “confusable” な点の数の期待値をパターンのサイズ n によらずに一定になるように δ を決定する。これには δ を n に応じて次のように決定する。まずパターン Y のある点 y_i がある点 y_j と “confusable” である確率は、 y_j が y_i を中心とする半径 2δ の円の中にある確率であり、点は半径 ρ の円内で一樣に分布しているため、(y_i が円の端のほうにある場合を無視して考えれば) これはこの円の面積と y_j が選ばれている円 (半径 ρ) の面積の比、

$$\frac{\pi(2\delta)^2}{\pi\rho^2} = \frac{4\delta^2}{\rho^2} \quad (3.115)$$

に等しい。パターン Y 内の点はそれぞれ独立に選ばれるため、 y_i と “confusable” な点の数の期待値 E_c はこの比の値を $n-1$ 倍したものである。

$$E_c = \frac{4\delta^2(n-1)}{\rho^2} \quad (3.116)$$

さてここで

$$\delta = \frac{\rho}{2} \sqrt{\frac{k}{n}} \quad (3.117)$$

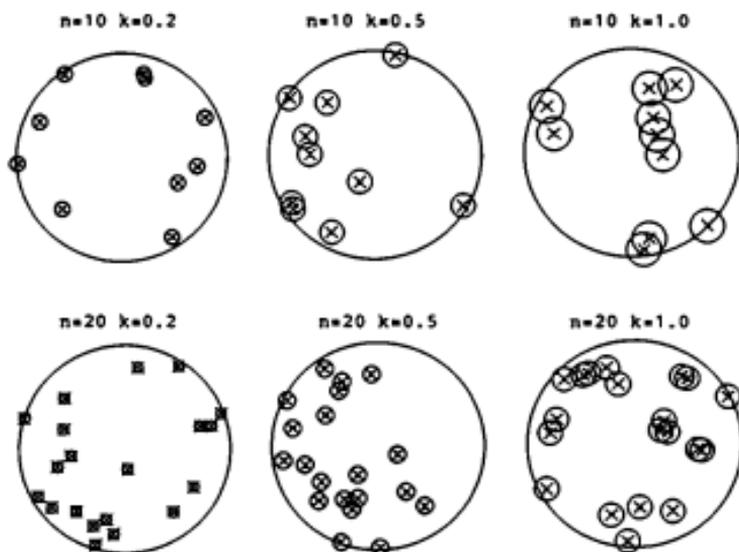


図 3.9: さまざまなパラメータに対するノイズの大きさ

とすると、

$$E_c = \frac{k(n-1)}{n} < k \quad (3.118)$$

となり、 δ をこのように定める時、 y_i と “confusable” な点の数の期待値は k 以下となるのがわかる。そこで k をパラメータとして、 δ を (3.117) 式によって決定する。三次元の場合にも同様の議論から、

$$\delta = \frac{\rho}{2} \sqrt{\frac{k}{n}} \quad (3.119)$$

を用いて η を決定する。二次元の場合のさまざまな n, k についてのパターン Y とノイズの円を図 3.9 に示す。Baird は $k = 1.0$ の場合を “high” noise、 $k = 0.5$ の場合を “moderate” noise、 $k = 0.2$ の場合を “low” noise と呼んでいる。

$k = 0.2$ 、 $k = 0.5$ 、 $k = 1.0$ の場合について、さまざまな大きさの点パターン対をこのような手続によって作成し、提案する全体マッチングアルゴリズムを適用し

た。実験1と同様に、それぞれのパラメータについてマッチングを100回行ない、探索木の大きさ、平均計算時間を求めた。

3.6.2 結果および考察

実験1の二次元パターンの場合の結果を図3.10に、三次元パターンの場合の結果を図3.11に示す。それぞれ、左側の図がさまざまな大きさの点パターンのマッチングに対する探索木の平均の大きさ、右側の図が平均計算時間である。計算時間はプログラムの最適化などによって変動するため、これは単なる目安である。またスケールの適格範囲による枝刈りに於いて、局所的な制約による枝刈り (EJ_l)、大局的な制約による枝刈り (EJ_g) のそれぞれが、どれほどの効果を持つかを見るため、 $\delta = 0.1$ の場合について、(1) 局所的な制約のみ (EJ_l) から枝刈りを行なった場合、(2) 大局的な制約のみ (EJ_g) から枝刈りを行なった場合、そして(3) その両者を併用した場合、それぞれについて、平均の探索木の大きさがどれほどになるかを調べた結果が表3.1である。この際、既発見のマッチングの大きさによる枝刈りは、いずれの場合も行なっている。

一方、図3.12、図3.13はそれぞれ二次元、三次元の場合の実験2の結果である。実験1の場合と同様に、左側の図がさまざまな大きさの点パターンのマッチングに対する探索木の平均の大きさ、右側の図が平均計算時間である。

これらの実験結果から以下のことがわかる。

- 実験1の場合、 δ の大きさによっていくらかの違いがあるが、二次元の場合も三次元の場合も、マッチングを行なうパターンの大きさを1点増やすごとに、ほぼ2倍の探索木の大きさ、計算時間を必要としている。実際の応用においては、完全にランダムなパターンをマッチングさせることは考えにくいので、これはほぼ最悪の場合の結果であると思われる。ちなみにそれぞれの場合について、得られた δ 制限最大マッチングの大きさとその個数は表3.2の通りとなった。いずれの場合も、ほぼマッチングを行なうパターンの大きさに比例した大きさのマッチングが得られていることが判る。
- 探索木の大きさは、二次元の場合と三次元の場合を比較してそれほど大きな差

は無い。このため計算時間も高々 2 - 3 倍程度の差となっている。三次元の場合は二次元の場合と比べて自由度が高いため、そこでのマッチングは格段に困難なような印象を持ちがちであるが、この結果を見る限りでは、両者にそれほど差は無い。

- 表 3.1 から、大局的な制約 (EJ_g) のみを用いて得られる探索木の大きさは、大局的、局所的制約 (EJ_l) 共に用いることによって得られる探索木の大きさにほぼ等しい。つまり EJ_l を用いた枝刈りのよって刈られるような枝は、 EJ_g を用いた枝刈りによっても刈られてしまい、枝刈り能力の点からは、 EJ_g は EJ_l をほとんど含むことが判る。 EJ_l の効果は、その計算の容易さに依っている。
- 全体マッチング (実験 2) は、通常のマッチングに比較して非常に高速である。パラメータなどにもよるが、同じ大きさの問題を比較した場合、ほぼ 1/100 程度の時間で行なうことができる。探索木の大きさ、計算時間は k の値に大きく依存するが、 $k \leq 0.5$ の場合には、二次元の場合も三次元の場合も、実験を行なった範囲では、ほぼパターンの大きさの 3 乗に比例した程度の時間でマッチング可能であった。

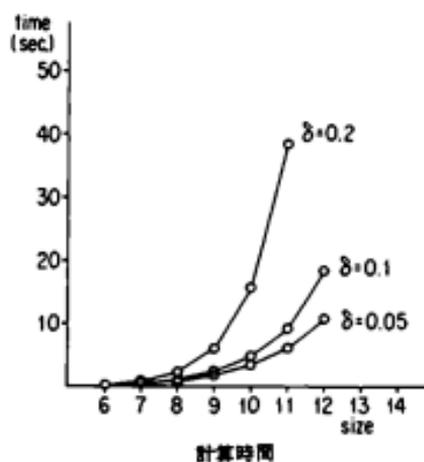
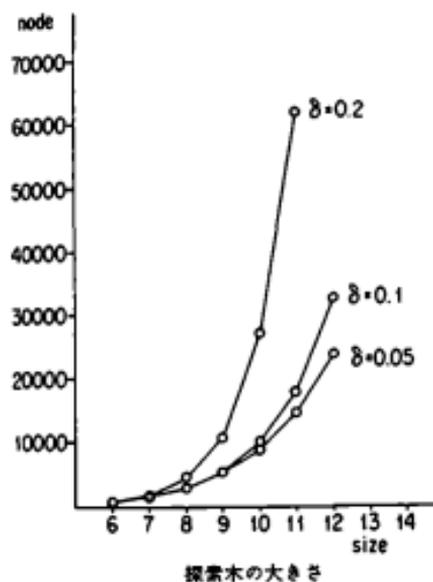


図 3.10: 実験 1 の結果 (二次元)

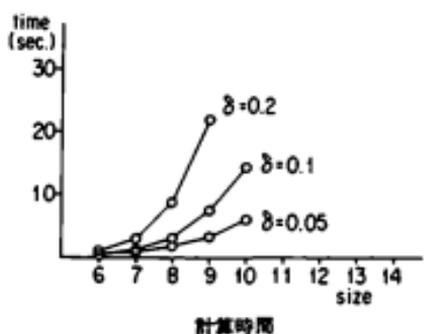
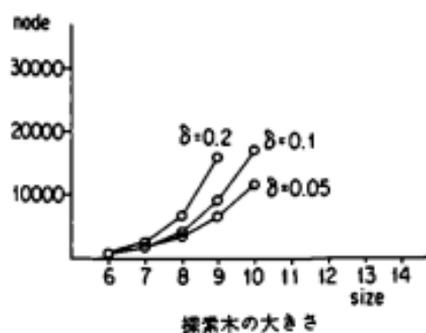


図 3.11: 実験 1 の結果 (三次元)

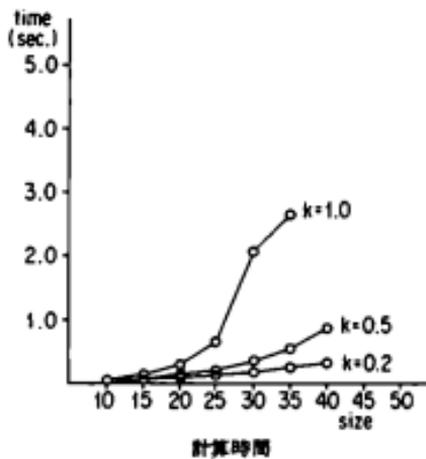
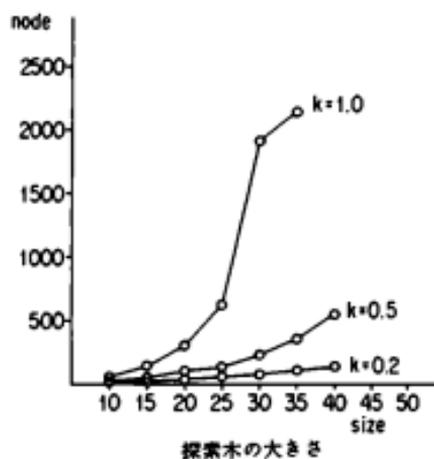


図 3.12: 実験2の結果 (二次元)

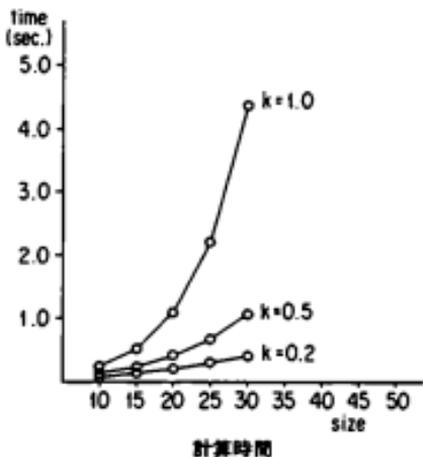
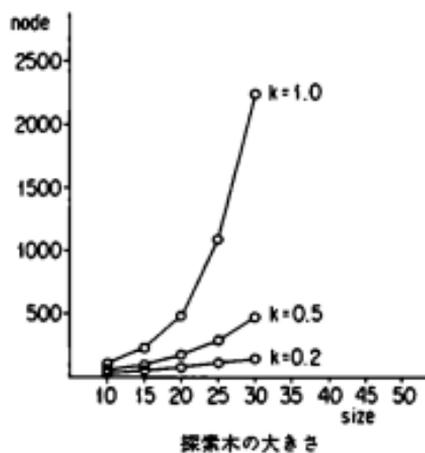


図 3.13: 実験2の結果 (三次元)

表 3.1: 適格範囲による枝刈り手法の効果の比較

2 dim $\delta = 0.1$

| size | El_t | El_p | $El_t + El_p$ |
|------|--------|--------|---------------|
| 6 | 1013 | 634 | 632 |
| 7 | 2411 | 1267 | 1259 |
| 8 | 6230 | 2765 | 2738 |
| 9 | 14271 | 5404 | 5325 |
| 10 | 31633 | 10224 | 10012 |

3 dim $\delta = 0.1$

| size | El_t | El_p | $El_t + El_p$ |
|------|--------|--------|---------------|
| 6 | 1111 | 807 | 805 |
| 7 | 2854 | 1816 | 1811 |
| 8 | 7721 | 4157 | 4143 |

表 3.2: 実験 1 により得られるマッチングの大きさとその個数

2 dim

| size | $\delta = 0.05$ | $\delta = 0.1$ | $\delta = 0.2$ |
|------|-----------------|----------------|----------------|
| 6 | 3.1 (15.4) | 3.7 (19.2) | 4.4 (15.4) |
| 7 | 3.5 (26.2) | 4.3 (11.1) | 5.2 (14.9) |
| 8 | 3.9 (16.6) | 4.6 (19.1) | 5.8 (21.7) |
| 9 | 4.2 (9.4) | 5.1 (16.1) | 6.5 (23.4) |
| 10 | 4.5 (14.6) | 5.6 (14.2) | 7.3 (29.3) |
| 11 | 4.8 (14.6) | 6.1 (11.8) | 8.0 (29.2) |
| 12 | 5.1 (11.4) | 6.5 (15.5) | - (-) |

3 dim

| size | $\delta = 0.05$ | $\delta = 0.1$ | $\delta = 0.2$ |
|------|-----------------|----------------|----------------|
| 6 | 3.0 (21.8) | 3.6 (28.6) | 4.3 (19.6) |
| 7 | 3.2 (48.1) | 4.0 (16.8) | 4.8 (22.6) |
| 8 | 3.6 (71.2) | 4.2 (18.8) | 5.3 (23.8) |
| 9 | 3.9 (46.1) | 4.4 (45.3) | 5.9 (20.7) |
| 10 | 4.0 (8.1) | 4.8 (34.5) | - (-) |

各項目の左側は得られたマッチングの平均の大きさであり、右側（括弧の中）は一試行あたり得られたマッチングの平均個数である。上の表が二次元の場合、下の表が三次元の場合である。

3.7 むすび

本探索を用いた点パターンマッチングアルゴリズムを提案した。許容できるマッチング誤差を定義するため“ δ 制限”という条件を用い、この条件を満足する二つの点パターン間のすべてのマッチングを、このアルゴリズムは生成する。

提案した手法の特徴・長所としては、次のような点が挙げられる。

- a) 従来から提案されているマッチング手法の中には、対象が二次元の場合に限定されるものも少なくないが [5, 1]、提案した点パターンマッチング手法は一般の m 次元空間内の点パターンに対して適用可能である。また、点パターン間の関係についても、相似変換（回転・平行移動・スケール変換）の範囲まで考慮されている。
- b) 得られる解の意味が明確に定義されており、また定義を満足する解をすべて数え上げることができる。これに対して、緩和法やヒューリスティックなマッチング手法においては、マッチングの結果最終的に得られた解がどのようなものなのか（どのような条件を満足するのか）、直観的に判りづらいことが多い。
- c) 手法が柔軟であり、相似変換のパラメータに制約のある場合や鏡像パターンを考慮する場合など、さまざまな拡張が可能である。
- d) 従来の手法においては、マッチングに際して数多くのパラメータを設定する必要がある場合が少なくない。実際にアルゴリズムを適用する場合には、これらのパラメータをどのように設定するかが大きな問題となる。一方、提案したマッチング手法においては、マッチングを行なう際に設定しなければならないパラメータは許容誤差 δ だけであり、これは得られる解の意味そのものであるから、設定することは容易である。

また、マッチングに要する時間については、計算機実験により次のように求められた。

- e) ランダムに生成された点パターンをマッチングさせる場合には、ほぼ点数を 1 増やすごとに 2 倍の処理時間を必要とする。

- f) 全体マッチングの場合には、実験した範囲ではほぼ点数の 3 乗に比例した時間程度でマッチングが可能である（中位のノイズの場合）。

第 4 章

空間記述されたパターン間のマッチング II

— パラメータ化された点パターン間のマッチング

4.1 はじめに

前章においては、点パターンの座標情報のみに基づく非常に一般的なマッチング手法を提案した。本章ではさらにこれを拡張し、平面上のパターンに限られるが、パラメータ化された点パターン間のマッチング手法を考察する。

例えば“はさみ”を考えてみる。はさみは支点を中心にして2つの刃が互いに交差し回転する。このような対象をその交差角に独立に、また交差角それぞれについて全てのモデルを準備することなく認識したいとする。このためには、対象のモデルとして交差角をパラメータとして持つようなモデルを用意し、そのようなモデルとのマッチングを考えなければならない。また、一方、“壺”のような言葉は、単に大小ばかりでなく、例えば縦横のプロポーションなどの多様性も含めた、多種多様な壺を差し示すために用いられる。このように多様な“壺”を、やはりすべてのモデルを用意することなく認識するためには、例えば、壺の縦横の比をパラメータとして持つようなモデルを準備しておけばよい。

このようにパラメータ化されたモデルを用いることにより巧く表現可能な対象は、現実の世界において決して少なくない。これらは、例えば、次のように分類されよう。

- “はさみ”のように、互いに動くことが可能な幾つかのパーツから構成される物体。
- “壺”のように、多種多様な対象をパラメータ化によって一括りにして表現可能な場合。これには生物の成長による形態の変化なども含まれる。例えば、植物の葉は成長するにつれて単に相似的に大きくなるだけでなく、その特徴を保ったまま、その形態を少しずつ変化させることが多い。そのような“葉”を、その年齢に係わらず認識するためには、パラメータ化が有効であろう。

本章では、このようなパラメータ化された対象をモデル化するために、点パターンの定義を拡張し、パラメータ化された点パターン (parameterized point pattern) の概念を導入する。そして、そのような点パターン間のマッチング手法としてパラメータ化された点パターン間のマッチングアルゴリズム (parameterized point pattern matching algorithm) を提案する。パラメータ化された点パターンは非常に柔軟な概念であり、上で二つに分類したようなパラメータ化された対象のモデルを、共に非常に広い範囲でカバー、表現することが可能である。

パラメータ化されたモデルによる物体認識についての研究はそれほど多くない。Grimson [27] はパラメータ化された対象の例として、上で挙げた“はさみ”の例の他に、物体の一部がある軸に沿って引きのばされるような変形 (さまざまな長さの柄を持つハンマー) を取り上げ、解釈木 (interpretation tree) を探索していく際に、局所的に定義された制約を用いて枝刈りを行なうことによって、パラメータ化されたモデルと対象とのマッチングを行なう手法を提案している。その手法は本章で提案する手法に類似したものであるが、Grimson の場合には解の定義が随に与えられていないため、最終的に得られる解の意味はそれほど明確ではない。この他に ACRONYM SYSTEM [11] などもこのような物体認識の例となっている。

以下、次節では、パラメータ化された点パターンの定義を与え、それに基づく点パターンマッチング問題を定義する。次に、4.3節でその問題を木探索で解くための枝刈り手法を示し、4.4節では簡単な実行例を示す。本章では一般的にパラメータ化された点パターン間のマッチングアルゴリズムを示すにとどめ、上で考えているような形状認識に対する応用などは、後の第5章にまとめて示す。

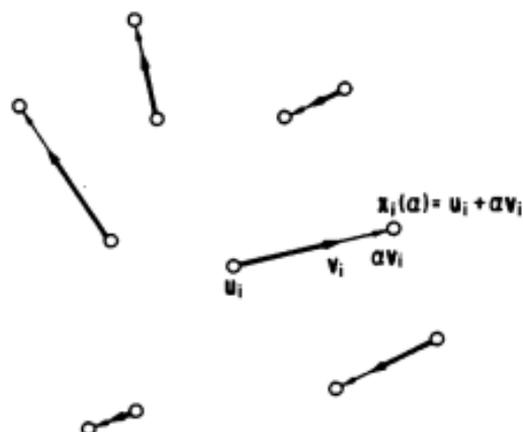


図 4.1: パラメータ化された点パターン

4.2 パラメータ化された点パターン間のマッチング

平面上のパラメータ化された対象を表現するためのモデルとして、本章では、モデル内の各点がそれぞれその点ごとに定まるある方向へパラメータの値だけ移動するようなモデルを考える。そして、そのようなモデルを表現するものとして、二次元のパラメータ化された点パターン $X(\alpha) = \{x_1(\alpha), x_2(\alpha), \dots\}$ を次のように定義する。

$$x_i(\alpha) = u_i + \alpha v_i \quad (4.1)$$

ここで、 α がパラメータであり、 u_i は $\alpha = 0$ における点 x_i の位置を、 v_i は α が変化した時に点 x_i が動いて行く方向および移動の大きさを表現している。 α を変化させることにより、点パターン X は $X(\alpha_{\min}) = \{u_1 + \alpha_{\min} v_1, u_2 + \alpha_{\min} v_2, \dots\}$ から $X(\alpha_{\max}) = \{u_1 + \alpha_{\max} v_1, u_2 + \alpha_{\max} v_2, \dots\}$ まで α に応じて連続的に変化することになる (図 4.1)。ここで、 α を自由に選べるとすると、自由度が大きくなり

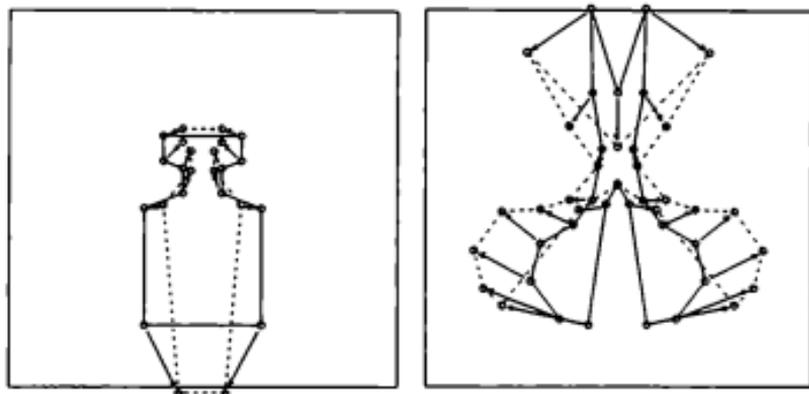


図 4.2: パラメータ化された点パターンによって表現された壺とはさみ

すぎるため、 α は次の範囲を動くものと仮定する。

$$\alpha \in EI_{\alpha, \text{init}} = [\alpha_{\text{min}}, \alpha_{\text{max}}] \quad (4.2)$$

外形上の特徴点集合を用いて物体を表現するとして、このような点パターンをモデルとして用いることにより、非常に広い範囲の対象をモデル化することが可能となる。例えば、物体の一部が引きのばされるような変形がこのモデルを用いて表現できることは明らかであるし、またはさみのように回転を伴う対象についても、変形の範囲がそれほど大きくない場合にはそれを近似的に表現可能である。さまざまなプロポーションを持つ壺のシルエットや、口を開いたはさみをこのような点パターンで表現したものが図 4.2 である。また、それぞれパラメータ $\alpha = 0, 0.4, 0.7, 1.0$ におけるパターンを示したものが図 4.3 である。

さて、このようにパラメータ化された点パターン X と通常の点パターン Y 間のマッチングを考える。両パターン間の関係として、通常の点パターンマッチングの場合と同様に相似変換

$$T(x(\alpha)) = cRz(\alpha) + t \quad (4.3)$$

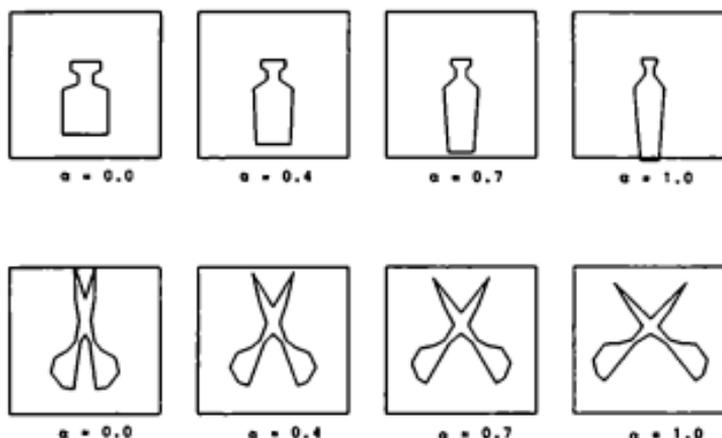


図 4.3: さまざまなパラメータに対する変とはさみ

を考え、さまざまな用語についても第3章と同様に定めることとする。ただし、パラメータ化された点パターン間のマッチングに対する δ 制限の概念については、次のように拡張する。大きさ n のマッチング $\Phi = \{(x_1(\alpha), y_1), (x_2(\alpha), y_2), \dots, (x_n(\alpha), y_n))\}$ が次の条件を満足する時、 Φ を δ 制限であると呼ぶ。

$$d = \max_i \|y_i - T_\Phi(x_i(\alpha^*))\| \leq \delta \quad (4.4)$$

ここで、 T_Φ 、 α^* は次式を最小とする相似変換 T およびパラメータ α である。

$$\begin{aligned} e^2 &= \frac{1}{n} \sum_{i=1}^n \|y_i - T(x_i(\alpha))\|^2 \\ &= \frac{1}{n} \sum_{i=1}^n \|y_i - cR(u_i + \alpha v_i) - t\|^2 \end{aligned} \quad (4.5)$$

これらの定義のもとで、パラメータ化された点パターン間のマッチング問題は、点パターン X 、 Y 間の上記の意味で δ 制限された最大のマッチングをすべて見つける問題となる。

4.3 探索木の枝刈り

パラメータ化された点パターン間のマッチングにおいても、探索木の構成、深さ優先探索、また既発見のマッチングの大きさによる枝刈りを行なう事などは、通常のマッチングの場合とまったく同じである。この場合異なってくるのは、枝刈り手法のうちの適格範囲による枝刈りの部分であり、この節では、このようなパラメータ化された点パターンマッチングに特有の枝刈り手法について説明する。

通常のマッチングの場合には、そのマッチング自身およびそれを拡大して得られるマッチングが δ 制限であるための必要条件として、スケールに関する適格範囲 $EI(\Phi)$ を定義し、その安全な近似 $EI''(\Phi)$ を探索木の各ノードに対して計算することによって枝刈りを行なった。パラメータ化された点パターンマッチングにおいても同様に、このような適格範囲を定義し、枝刈りを行なう。ただし、この場合にはスケール c ばかりではなく、次のように、パラメータ α に関する適格範囲も導入する。

$$EI_{\alpha}(\Phi) = \{ \alpha \mid \exists c, t, R \max_i \| y_i - (cRx_i(\alpha) + t) \| \leq \delta \} \cap EI_{\alpha, \text{init}} \quad (4.6)$$

$$EI_c(\Phi) = \{ c \mid \exists \alpha \in EI_{\alpha, \text{init}}, t, R \max_i \| y_i - (cRx_i(\alpha) + t) \| \leq \delta \} \quad (4.7)$$

これらについて、3.3 節における性質 1、性質 2、性質 3 が同様に成立する。このため、 $EI_{\alpha}(\Phi)$ 、 $EI_c(\Phi)$ が共に空でないことは、 Φ およびそれを拡大して得られるマッチングが δ 制限であるための必要条件となる。しかし、通常のマッチングの場合と同じく、これらを直接求めることは容易ではない。このため、前章と同じく、これらの近似、 $EI'_{\alpha}(\Phi)$ 、 $EI'_c(\Phi)$ を計算することによって枝刈りを行なう。以下ではそれぞれについての安全な近似手法を述べる。

4.3.1 $EI_{\alpha}(\Phi)$ の近似

通常のマッチングの場合と同様に、マッチング Φ に対して、次のようなパラメータの範囲 $I_{\alpha}(\Phi)$ を導入する。

$$I_{\alpha}(\Phi) = \{ \alpha \mid \min_{c, t, R} \frac{1}{n} \sum_{i=1}^n \| y_i - (cRx_i(\alpha) + t) \|^2 \leq \delta^2 \} \quad (4.8)$$

明らかに、 $n \leq 2$ の時、 $I_{\alpha}(\Phi) = (-\infty, \infty)$ である。このような $I_{\alpha}(\Phi)$ について、次の定理が成立する。証明は、定理 3.1 とまったく平行に行なえるため省略する。

定理 4.1 任意のマッチング ϕ, ϕ' について、 $\phi' \subseteq \phi$ の時、 $I_o(\phi') \supseteq EI_o(\phi)$ である。

この定理から、 $EI_o(\phi)$ の安全な近似 $EI'_o(\phi)$ を次のように定義する。

$$EI'_o(\phi) = EI_{o,init} \cap \bigcap_{i=0}^n I_o(\phi_i) \quad (4.9)$$

ここで、 $\phi_0 (= \phi), \phi_1, \dots, \phi_{n-1}, \phi_n (= \phi)$ は、木探索につれて次第に拡大されたマッチングの列であり、 $I_o(\phi_i) = (-\infty, \infty)$ ($i \leq 2$) とする。

$EI'_o(\phi)$ を枝刈りに用いるためには $I_o(\phi)$ が計算可能である必要があるが、これは次のように求められる。

ϕ を点パターン $X(\alpha), Y$ 間の大きさ n ($n \geq 3$) のマッチングとし、 $X_\phi(\alpha) = \{x_1(\alpha), x_2(\alpha), \dots, x_n(\alpha)\}$ 、 $Y_\phi = \{\phi(x_1), \phi(x_2), \dots, \phi(x_n)\} = \{y_1, y_2, \dots, y_n\}$ とする。さて、

$$c^2(\alpha) = \min_{c,t,R} \frac{1}{n} \sum_{i=1}^n \|y_i - (cRx_i(\alpha) + t)\|^2 \quad (4.10)$$

とすると、定理 3.3 から

$$c^2(\alpha) = \sigma_y^2 - \frac{(\text{tr}(DS))^2}{\sigma_x^2} \quad (4.11)$$

となる。(簡単のため $\sigma_x^2 > 0$ とする)。ここで σ_x^2, σ_y^2 は $X_\phi(\alpha), Y_\phi$ の平均のまわりの分散、また $\Sigma_{xy} = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix}$ を $X_\phi(\alpha), Y_\phi$ 間の共分散行列として $(\text{tr}(DS))^2 = (s_{11} + s_{22})^2 + (s_{12} - s_{21})^2$ である。さらに、 $x_i(\alpha) = u_i + \alpha v_i$ であるから、 σ_u^2, σ_v^2 をそれぞれ $U_\phi = \{u_1, u_2, \dots, u_n\}, V_\phi = \{v_1, v_2, \dots, v_n\}$ の平均のまわりの分散、 Σ_{uv} を U_ϕ, V_ϕ 間の共分散行列、 $\Sigma_{xy} = \begin{pmatrix} s'_{11} & s'_{12} \\ s'_{21} & s'_{22} \end{pmatrix}$ 、 $\Sigma_{yy} = \begin{pmatrix} s''_{11} & s''_{12} \\ s''_{21} & s''_{22} \end{pmatrix}$ を U_ϕ と Y_ϕ および V_ϕ と Y_ϕ 間の共分散行列であるとする、

$$\Sigma_{xy} = \Sigma_{uy} + \alpha \Sigma_{vy} \quad (4.12)$$

$$\sigma_x^2 = \sigma_u^2 + 2\alpha \text{tr} \Sigma_{uv} + \alpha^2 \sigma_v^2 \quad (4.13)$$

であり、また

$$(\text{tr}(DS))^2 = (s'_{11} + s'_{22})^2 + (s'_{12} - s'_{21})^2 \quad (4.14)$$

$$+2\alpha\{(s'_{11} + s'_{22})(s''_{11} + s''_{22}) + (s'_{12} - s'_{21})(s''_{12} - s''_{21})\} \\ +\alpha^2\{(s''_{11} + s''_{22})^2 + (s''_{12} - s''_{21})^2\}$$

となる。そこで

$$\sigma_x^2 = C + 2B\alpha + A\alpha^2 \quad (4.15)$$

$$(\text{tr}(DS))^2 = F + 2E\alpha + D\alpha^2 \quad (4.16)$$

$$A = \sigma_v^2$$

$$B = \text{tr}\Sigma_{uv}$$

$$C = \sigma_w^2$$

$$D = (s''_{11} + s''_{22})^2 + (s''_{12} - s''_{21})^2$$

$$E = (s'_{11} + s'_{22})(s''_{11} + s''_{22}) + (s'_{12} - s'_{21})(s''_{12} - s''_{21})$$

$$F = (s'_{11} + s'_{22})^2 + (s'_{12} - s'_{21})^2$$

として、これらを

$$\varepsilon^2(\alpha) \leq \delta^2 \quad (4.17)$$

に対して代入し整理すると、次のような不等式が得られる。

$$\{(\sigma_v^2 - \delta^2)A - D\}\alpha^2 + 2\{(\sigma_v^2 - \delta^2)B - E\}\alpha + \{(\sigma_v^2 - \delta^2)C - F\} \leq 0 \quad (4.18)$$

この式は α に関する二次の不等式であるから、(4.18)式が成立する α の範囲、つまり $\mathcal{I}_\alpha(\Phi)$ は容易に決定することができる。 $(\sigma_v^2 - \delta^2)A - D \neq 0$ とすると、この式の解は、

(1) $\sigma_v^2 - \frac{D}{A} < \delta^2$ の時

$$(1.1) \text{ (4.18) 式の左辺} = 0 \text{ が実根を持たない時} \quad \mathcal{I}_\alpha(\Phi) = (-\infty, \infty)$$

$$(1.2) \text{ 実根 } \alpha_1, \alpha_2 \text{ (} \alpha_1 \geq \alpha_2 \text{) を持つ時} \quad \mathcal{I}_\alpha(\Phi) = (-\infty, \alpha_2] \cup [\alpha_1, \infty)$$

(2) $\sigma_v^2 - \frac{D}{A} > \delta^2$ の時

$$(2.1) \text{ (4.18) 式の左辺} = 0 \text{ が実根を持たない時} \quad \mathcal{I}_\alpha(\Phi) = \emptyset$$

$$(2.2) \text{ 実根 } \alpha_1, \alpha_2 \quad (\alpha_1 \geq \alpha_2) \text{ を持つ時} \quad I_\alpha(\Phi) = [\alpha_2, \alpha_1]$$

となる。

このように条件 (1.2) の場合には $I_\alpha(\Phi)$ が二つの部分に分裂することになる。複数の $I_\alpha(\Phi)$ の共通部分をとるような操作を考える場合、このような対象が存在することは、その操作を複雑化するため好ましくない。ところで、(1.2) が成立する条件、 $\sigma_V^2 - \frac{D}{\lambda} < \delta^2$ というのは、 V_Φ のパターンが Y_Φ のパターンに類似しているということであり ($\sigma_V^2 - \frac{D}{\lambda}$ は V_Φ を Y_Φ へマッピングさせた時の平均二乗誤差)、 V_Φ がパラメータ化された点パターンの変化部分を表しているという意味からは、このような式が成立するのは不自然に思われる。実際、幾つかの例に対して実験してみた結果でも、このような場合が成立するのは極く少数のノードについてのみであった。このため、このような場合に α の範囲を制限するのはあきらめ、 $((\sigma_V^2 - \delta^2)A - D = 0$ が成立するような場合も含めて) $I'_\alpha(\Phi)$ を次のように決定することとする。

$$(1) \sigma_V^2 - \frac{D}{\lambda} \leq \delta^2 \text{ の時} \quad I'_\alpha(\Phi) = (-\infty, \infty)$$

$$(2) \sigma_V^2 - \frac{D}{\lambda} > \delta^2 \text{ の時}$$

$$(2.1) (4.18) \text{ 式の左辺} = 0 \text{ が実根を持たない時} \quad I'_\alpha(\Phi) = \phi$$

$$(2.2) \text{ 実根 } \alpha_1, \alpha_2 \quad (\alpha_1 \geq \alpha_2) \text{ を持つ時} \quad I'_\alpha(\Phi) = [\alpha_2, \alpha_1]$$

このように定められた $I'_\alpha(\Phi)$ から定義される $EJ'_\alpha(\Phi)$ は、適格範囲 $EI_\alpha(\Phi)$ の安全な近似となる。また、 Σ_{uv} などのさまざまな統計量をマッピングが進むにつれて (3.3節と同様に) 漸進的に計算することにより、この計算を、探索木の各ノードにおいて、点の数に依らない定数時間で行なうことができることも明らかである。

さて次に、二つの点パターンを重ね合わせた時の、平均二乗誤差を最小にするパラメータ α 、つまり (4.4) 式の最適パラメータ α^* であるが、これは次のように計算できる。この結果は、最終的にマッピングが δ 制限であるか否かを確認めたり、あるいは実際に点パターン間の重ね合わせを行なう際に必要となる。

相似変換に対する平均二乗誤差 $e^2(\alpha)$ の最小値を与える α を決定するため、 $e^2(\alpha)$ を α について微分する。簡単のため、以降の議論では $AE - BD \neq 0$ と仮定する。

$$\frac{de^2(\alpha)}{d\alpha} = -\frac{1}{\sigma_x^2} \frac{d(\text{tr}(DS))^2}{d\alpha} + \frac{(\text{tr}(DS))^2}{\sigma_x^4} \frac{d\sigma_x^2}{d\alpha}$$

$$= \frac{1}{\sigma_x^2} \left\{ (\text{tr}(DS))^2 \frac{d\sigma_x^2}{d\alpha} - \sigma_x^2 \frac{d(\text{tr}(DS))^2}{d\alpha} \right\} \quad (4.19)$$

この式に、(4.15)式、(4.16)式を代入して整理すると、次のようになる。

$$\frac{d\epsilon^2(\alpha)}{d\alpha} = \frac{1}{\sigma_x^2} \{ (AE - BD)\alpha^2 + (AF - CD)\alpha + (BF - CE) \} \quad (4.20)$$

この式右辺の二次式の判別式を D とすると、

$$\begin{aligned} A^2 D^2 D &= A^2 D^2 \{ (AF - CD)^2 - 4(AE - BD)(BF - CE) \} \\ &= (AE - BD)^4 + 2(AE - BD)^2 \{ (AC - B^2)D^2 + (DF - E^2)A^2 \} \\ &\quad + \{ (AC - B^2)D^2 - (DF - E^2)A^2 \}^2 \end{aligned} \quad (4.21)$$

となる。 $\sigma_x^2 > 0$ 、 $(\text{tr}(DS))^2 \geq 0$ であるから、(4.15)(4.16)式の右辺二次式の判別式を D_1 、 D_2 とすると、

$$D_1 = B^2 - AC < 0 \quad (4.22)$$

$$D_2 = E^2 - DF \leq 0 \quad (4.23)$$

が成立し、また $AE - BD \neq 0$ であるから、

$$D > 0 \quad (4.24)$$

となる。ゆえに、

$$\frac{d\epsilon^2(\alpha)}{d\alpha} = 0 \quad (4.25)$$

は二実根 α_1 、 α_2 ($\alpha_1 > \alpha_2$) を持つ。このことと、 $\alpha \rightarrow \pm\infty$ の時、 $\epsilon^2(\alpha) \rightarrow \sigma_y^2 - \frac{D}{A}$ であることから、最適パラメータ α^* は

$$(1) \quad AE - BD > 0 \text{ の時} \quad \alpha^* = \alpha_1$$

$$(2) \quad AE - BD < 0 \text{ の時} \quad \alpha^* = \alpha_2$$

となる。

4.3.2 $EI_c(\Phi)$ の近似

通常のマッチングにおいては、大局的および局所的制約の二つの面からスケールに関する適格範囲の近似を行なった。しかし、パラメータ化された点パターン間のマッチングにおいては、大局的な制約をこの近似に利用することは極めて困難であるため、この場合には局所的制約のみから適格範囲の近似を行なう。

$\Phi_2 = \{(x_i(\alpha), y_i), (x_j(\alpha), y_j)\}$ を大きさ 2 のマッチングとし、

$$l_{\min} = \min_{\alpha \in EI_{\alpha, \text{local}}} \|x_j(\alpha) - x_i(\alpha)\| \quad (4.26)$$

$$l_{\max} = \max_{\alpha \in EI_{\alpha, \text{local}}} \|x_j(\alpha) - x_i(\alpha)\| \quad (4.27)$$

と定める。通常のマッチングの場合の大きさ 2 のマッチングに対する $I(\Phi)$ の計算での議論と同様に (図 3.4 参照)、相似変換のスケールがあまりに大きすぎたり小さすぎたりしている場合には、この変換およびパラメータ α をどのように選んでも、 $x_i(\alpha)$ 、 $x_j(\alpha)$ を y_i 、 y_j を中心とした半径 δ の円内へ移すことは不可能である。つまり、大きさ 2 のマッチング Φ_2 に対して

$$EI_c(\Phi_2) = [c_{\min}, c_{\max}] \quad (4.28)$$

$$c_{\min} = \max(0, \frac{1}{l_{\max}} (\|y_j - y_i\| - 2\delta)) \quad (4.29)$$

$$c_{\max} = \frac{1}{l_{\min}} (\|y_j - y_i\| + 2\delta) \quad (4.30)$$

となる。これから、一般のマッチング Φ に対するスケールの適格範囲 $EI_c(\Phi)$ を次式で近似する。

$$EI'_c(\Phi) = \bigcap_{\Phi' \subset \Phi \text{ and } \text{size}(\Phi')=2} EI_c(\Phi') \quad (4.31)$$

この計算も、3.3節の場合と同様に、マッチングが進むにつれて漸進的に実行可能である。

4.4 実行例

パラメータ化された点パターン間のマッチングアルゴリズムに対する理解を深めるため、次のような簡単な計算機実験を示す。乱数を用いて二次元の点パターン Y を

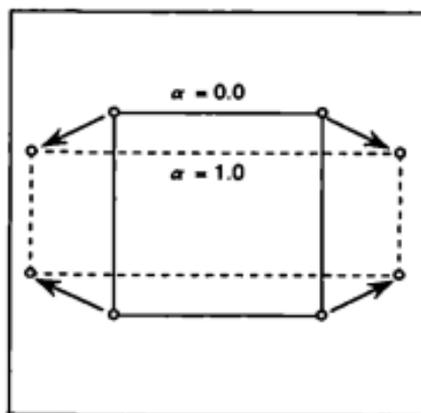


図 4.4: パラメータ表現された長方形

作り、それに対してパラメータ化された点パターン $X(\alpha)$ (図 4.4) をマッチングさせる。

$$X(\alpha) = \{x_1(\alpha), x_2(\alpha), x_3(\alpha), x_4(\alpha)\} \quad (4.32)$$

$$x_1(\alpha) = (0.5, 0.5)^T + \alpha(0.4, -0.2)^T \quad (4.33)$$

$$x_2(\alpha) = (0.5, -0.5)^T + \alpha(0.4, 0.2)^T \quad (4.34)$$

$$x_3(\alpha) = (-0.5, -0.5)^T + \alpha(-0.4, 0.2)^T \quad (4.35)$$

$$x_4(\alpha) = (-0.5, 0.5)^T + \alpha(-0.4, -0.2)^T \quad (4.36)$$

$X(\alpha)$ は様々な縦横比を持つ長方形を一度に表現するパラメータ化された点パターンであり、図 4.4 から判るように、 $\alpha = 0$ の時には高さと同様の比が 1 : 1 の正方形、 $\alpha = 1.0$ の時にはこの比が 1 : 3 の長方形を表現している。Y には 13 個の点が含まれており、それぞれその座標が -1.0 から 1.0 の範囲に入るように一様乱数を用いて生成されている。このマッチングにより、点パターン Y 内に含まれる様々な縦横比を持つ長方形パターンが抽出されることになる。

マッチングの許容誤差 $\delta = 0.1$ とし、 $EJ_{\alpha, \text{init}} = [0.0, 1.0]$ としてマッチングを行った結果を図 4.5 に示す。図 4.5 では、得られた解をすべて Y の上へ重ね合わせて

示している。この場合、 $\alpha = 0.090$ から $\alpha = 0.728$ まで、全部で 10 個の解が得られるが、長方形の対称性から、そのうち半分は結果として同一の解であり、図からは見分けることができない。すべての解を生成するのに要した時間は、1.03 秒、探索木の大きさ（ノード数）は 978 であった。

この実験で重要な点は、パラメータ化された点パターンマッチングアルゴリズムを用いることにより、さまざまな縦横比を持つ長方形を、それぞれに対して別々のモデルを用意することなく、同時に、一度で、点パターンから抽出可能であることである。

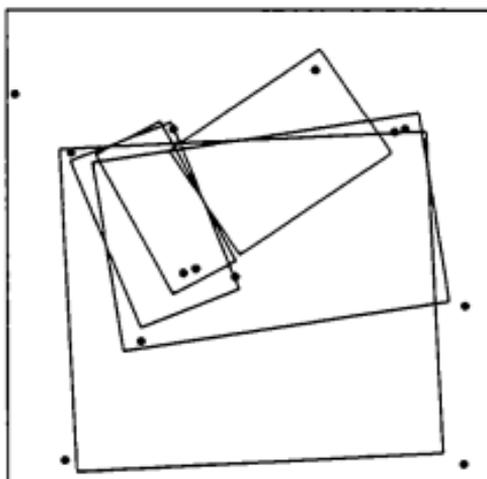
4.5 むすび

“はさみ”やさまざまなプロポーシオンを持つ“壺”のように、パラメータ化されたモデルによって表現することが自然な対象を取り扱うため、パラメータ化された点パターンの概念を導入し、前章で述べたマッチングアルゴリズムを拡張して、パラメータ化された点パターン間の一般的なマッチングアルゴリズムを提案した。

Grimson は [27] の中で、このような対象物の認識を行なう手法として、(1) マッチングの過程でパラメータに関する制約を更新・伝播させ、その制約（パラメータの許容範囲）が空とならないマッチングを解として探索する、あるいは(2) モデルを変形しない幾つかの部分に分割し（例えば“はさみ”を二つの刃に分ける）、それぞれを通常の手法を用いて認識して、得られたそれぞれの解を寄せ集め、最終的に互いに無矛盾で、対象を巧く説明する解釈を構成する、の二種の手法をあげ、その中で、(1) の手法の欠点として、パラメータに対する制約を自然で容易に計算・更新可能な形に表現することが困難であることを挙げている。しかし、本章で提案するマッチングアルゴリズムは、まさしく(1)に属する手法であり、この章は、このような原理に基づく非常に自然で一般的なマッチング手法が、パラメータ化された点パターンという概念を導入することにより、実際に構成可能であることを示すものとなっている。

$\delta = 0.10$
 $EJ_{a, out} = [0.0, 1.0]$

SIZE = 4
 No. of solutions = 10
 No. of nodes = 978
 time = 1.05



- (1) $e = 0.002123$ $d = 0.057231$ $\alpha = 0.478$
 $c = 0.649$ $b = (-0.392, 0.097)$ $R = -111.799$
 0 10 7 11
- (2) $e = 0.002123$ $d = 0.057231$ $\alpha = 0.478$
 $c = 0.649$ $b = (-0.392, 0.097)$ $R = 68.201$
 7 11 0 10
- (3) $e = 0.008645$ $d = 0.094803$ $\alpha = 0.321$
 $c = 0.607$ $b = (0.125, 0.404)$ $R = 146.989$
 7 10 1 3
- (4) $e = 0.008645$ $d = 0.094803$ $\alpha = 0.321$
 $c = 0.607$ $b = (0.125, 0.404)$ $R = -34.011$
 1 3 7 10
- (5) $e = 0.007067$ $d = 0.095118$ $\alpha = 0.471$
 $c = 0.991$ $b = (0.087, 0.066)$ $R = 171.234$
 11 0 9 2
- (6) $e = 0.007067$ $d = 0.095118$ $\alpha = 0.471$
 $c = 0.991$ $b = (0.087, 0.066)$ $R = -8.766$
 9 2 11 0
- (7) $e = 0.006181$ $d = 0.099559$ $\alpha = 0.090$
 $c = 1.412$ $b = (0.008, -0.226)$ $R = -2.812$
 9 4 5 0
- (8) $e = 0.006181$ $d = 0.099559$ $\alpha = 0.090$
 $c = 1.412$ $b = (0.008, -0.226)$ $R = 177.188$
 5 0 9 4
- (9) $e = 0.006923$ $d = 0.099802$ $\alpha = 0.728$
 $c = 0.421$ $b = (-0.343, 0.189)$ $R = -117.987$
 0 10 7 6
- (10) $e = 0.006923$ $d = 0.099802$ $\alpha = 0.728$
 $c = 0.421$ $b = (-0.343, 0.189)$ $R = 82.013$
 7 6 0 10

図 4.5: パラメータ化された点パターンのマッチング結果

第 5 章

点パターンマッチングの形状認識への応用

5.1 はじめに

物体の二次元形状（輪郭）からその物体を認識する、あるいはその物体のシーン内での位置を決定する問題はコンピュータビジョン研究における重要な課題の一つである。

このような問題は大きく二つの場合に分けられる。一つは対象物が分離され、その全体が観測されるような場合であり、もう一つの場合は、対象物の一部を他の物体が隠していたり、あるいはカメラの視野の関係から対象物の全体ではなくその一部しか観測できないような場合である（部分形状認識の問題）。前者の場合には、その全体的な形状からモーメントやフーリエ記述などの大局的な形状特徴を計算し、それらの値をもとにして、特徴空間内であらかじめ求められている物体モデルとの間の識別を行なうような手法が多く採用されるが [74, 17, 16]、後者の部分形状認識の場合には、このような大局的特徴量を用いることはできないため、対象物体とモデル間の部分的な特徴の対応をボトムアップ的に積み上げることにより認識を行なうことが普通である。

部分形状認識の従来の手法としては、弛緩法によるもの、一般化ハフ変換を用いるもの、あるいは構造的アプローチ (syntactic method) によるものなどさまざまな手法が提案されている。例えば Davis [14] は “association graph”、“line graph” と呼ぶ対象物とモデルとの局所的な対応関係を表現するグラフに対して弛緩法を適用

し、正しい対応づけにあたる部分グラフを抽出することによってマッチングを行なう方法を示しており、Bhanu [10]は階層的な確率的弛緩法を用いる、より精密な手法を提案している。また一般化ハフ変換によるものとしてはBallardによるもの[7]が有名であり、構造的なアプローチとしてはPavlidisらの提案する“syntactic shape analyzer”を用いたマッチング法などがある[45, 46]。“syntactic shape analyser”とは対象物境界の多角形近似を入力として、直線、コーナー、二次曲線など境界形状に対するより高次の記述列を出力するシステムであり、解析と同時に出力されるさまざまな属性をもとにして、このような記述間の対応づけを行なうことによりマッチングを行なう。またこの他にも、対象物境界の部分的特徴としてセグメントのフーリエ記述子を用い、それらの間の対応を動的計画法を用いて行なうものや[24]、対象物の境界を“concurve”と呼ぶ直線と円弧の並びで表現し、それらの間の対応を $\theta-S$ 表現（接線の角度と境界に沿ってのその点までの距離）のマッチングによって行なうものなど[48]が提案されている。また最近では三次元での物体認識手法としてセグメント間の空間的な相互関係を直接用いることによりマッチングを行なう方法が数多く提案されており[4, 26, 28]、これらは二次元での形状認識へも同様に適用可能なものが多い。この他にも数多くの手法が提案されている[75, 39]。

本章では、第3章、第4章で提案した点パターンマッチングアルゴリズムの応用として、このような部分形状による認識、位置決め問題を取りあげる[69]。認識対象が通常の（単独の）モデルで表現可能な場合、および第4章で述べたようなパラメータ化されたモデルで表現される場合それぞれについて、第3章および第4章の点パターンマッチングアルゴリズムを適用し、認識、位置決めを行なう。認識対象は、画像面に平行な台の上に置かれた本質的には二次元的な物体であり、カメラに対して三つの自由度（平行移動について二つの自由度、回転について一つの自由度）を持っている。またいくつかの物体が重なり合っていることを許している。物体のシルエットの多角形近似の各頂点を外形の特徴点として採用し、そのような点の集合を点パターンとして点パターンマッチングアルゴリズムを適用する。この際には、形状認識に特有の条件を用いることにより処理の高速化を行なっている。

以下5.2節では、点パターンマッチングを形状認識に用いる場合に特有の枝刈り条件を示し、続く5.3節において、通常の点パターンマッチングアルゴリズムおよびパ

ラメータ化された点パターンマッチングアルゴリズムを形状認識に適用した結果について示す。

5.2 形状認識に特有の枝刈り法

第3章、第4章で述べたのは、最も一般的に点の座標情報のみを用いてマッチングを行なう場合の枝刈り法であった。しかし、形状認識を行なう場合にはその他にも利用可能な情報があり、マッチングを行なう際にそれらを充分に利用することが効率的なマッチングを行なう上で非常に重要である。ここでは、本章で採用した形状認識に特有の枝刈り法を示す。

5.2.1 対応順序の無矛盾性を用いた枝刈り

認識対象のシルエットの多角形近似の頂点として選ばれた特徴点は、境界に沿って順番にラベルを与えられる。正しいマッチングは当然このラベルの順序に関して無矛盾でなければならないため、対応する点のラベル順序を調べることにより不必要な探索をさけることができる。例えば図 5.1において、点 x_1 が y_2 に、 x_2 が y_4 に、 x_3 が y_6 に対応していたとすると、(対応順序が矛盾するため) 点 x_4 は点 y_3 には対応することができない。そこでこの対応は第3章であげた枝刈り条件をテストすることなく枝刈りすることができる。このような対応は通常の(座標情報による)枝刈り条件によっていずれにしろ枝刈りされる可能性が高いが、ラベル順序の無矛盾性判定はテストが簡単なため、効率的な枝刈りには非常に有効である。

5.2.2 頂点の内角による枝刈り

各点における局所的な特徴量を利用することもマッチングを高速に行なうためには非常に有効である。例えば、多角形近似された各頂点における内角は相似変換において不変であるため、その大きさを比較することにより、モデル側(対応づけを行なう側)の各点の対応候補点(シーン側、つまり対応づけをされる側の点)を絞り込むことが可能である。具体的には、点 x_i での内角 $\theta(x_i)$ と点 y_j での内角 $\theta(y_j)$ の差がある与えられた値 θ_{th} より小さい場合のみ、それらの点が対応可能であると考える。

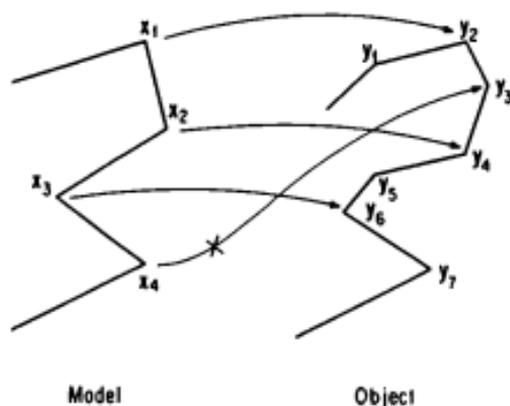


図 5.1: 対応順序の無矛盾性を用いた枝刈り

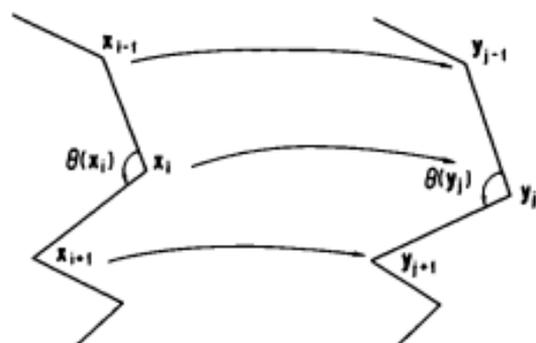
(図 5.2)

$$\theta(x_i) - \theta_{th} \leq \theta(y_j) \leq \theta(x_i) + \theta_{th} \quad (5.1)$$

パラメータ化された点パターンの場合には、 $\theta(x_i)$ の大きさが α の値によって変化するため、あらかじめ α が $EJ_{\alpha, int}$ の範囲で変化した時の $\theta(x_i)$ の最大値、最小値を求めておき、たとえ $\theta(x_i)$ がこの範囲で動いたとしても、 $\theta(y_j)$ との差が常に θ_{th} 以下である場合に、それらの点が対応可能であるとする。

$$\theta(x_i)_{min} - \theta_{th} \leq \theta(y_j) \leq \theta(x_i)_{max} + \theta_{th} \quad (5.2)$$

ただし、このような枝刈りを行なう場合には、得られる解の意味が、“対応する点の内角の大きさがあまり遠くないような δ 制限マッチング” というように変化している点には注意しなければならない。



$$\theta(x_i) - \theta_{th} \leq \theta(y_j) \leq \theta(x_i) + \theta_{th}$$

図 5.2: 頂点の内角による枝刈り

5.3 実験と結果

5.3.1 点パターンマッチングアルゴリズムによる部分形状認識

第3章で述べた点パターンマッチングアルゴリズムを用いて、互いに重なり合い、部分的にしかその形が見えないような機械部品の形状認識を行なった。認識対象物の8ビットグレイレベルの画像をTVカメラから入力し、それを判別分析的手法を用いた二値化アルゴリズム [43] によって二値化する。さらにその二値化画像の境界を抽出し、最後に多角形近似アルゴリズム [47] を用いて境界を線分で近似する。認識のためのモデルは同様な条件下で個別に部品を入力することによって作製する。入力画像の大きさは 256×256 画素である。図 5.3 に認識対象の多角形近似結果、図 5.4(a)(b)(c) にモデルとなる個々の部品の多角形近似結果を示す。多角形近似の際の近似許容誤差としては 2 画素分を認めている。図 5.4 の認識対象は図 5.3 に示す 3 個の機械部品が重なり合うように置かれたものであることがわかる。

最終的に得られた多角形近似の各頂点を点パターン内の点としてマッチングを行う。ただし、各点の座標値は、もとの画像の中心を原点として、画像全体が x 軸方向も y 軸方向もちょうど -1.0 と 1.0 の範囲にはいるように正規化している。マッ

グの許容誤差 $\delta = 0.05$ とし、各頂点の内角に対する許容誤差 θ_{th} を 15 度とする。認識結果を図 5.5(a)(b)(c) に示す。それぞれ図 5.4(a)(b)(c) と図 5.3 をマッチングさせた結果であり、マッチングの結果得られた相似変換を用いて、(a)(b)(c) を図 5.3 の上へ重ね合わせて示している。また対応づけに成功した点についてはそれぞれ図中で小さな円で示している。非常に良好なマッチングが行なわれていることがわかる。図には、得られた解の大きさ、個数、計算時間、および展開されたノードの数も示してある。ここで計算時間はマッチングに必要であった時間のみを含み、画像の二値化、境界抽出、多角形近似などに要する時間は含んでいない。またこれからわかるように、(c) の場合には 2 個の解が得られているが、図 5.5(c) にはそのうち最大誤差の小さな方を示してある。両者は、対応点が少し異なるだけで、大枠としては同一の解である。

ここでは、認識対象物のスケールは不明であるとして、マッチングを行なっている。しかし、本実験の場合、認識のためのモデルも認識対象と同じ条件下で入力することによって作成しているため、モデルと認識対象との間に、実際にはスケール差は存在しない。そこで、このことがあらかじめ判っているとして、スケールの初期制約を

$$EJ_{init} = [0.95, 1.05] \quad (5.3)$$

のように定めてマッチングを行なった結果、得られる解のリストを示しているのが図 5.6 である。最適スケールが EJ_{init} の外側 ($c = 0.942$) にある (c) の二番目の解を除いて、当然、スケールの制約を設けない場合と同一の解が得られている。一方、マッチングに要する計算時間は、このような制約により、(a) の場合で、107.2 秒から 19.0 秒へ、(b) の場合で、23.3 秒から 4.6 秒へ、(c) の場合で、96.0 秒から 44.9 秒へ、大きく減少している。第 3 章で提案したマッチング手法はもともと相似変換を考慮したものであるが、このようにスケールに関する初期制約を設けることにより、二つのパターン間のスケール差が既知の場合のマッチングにも、柔軟に対応可能である。

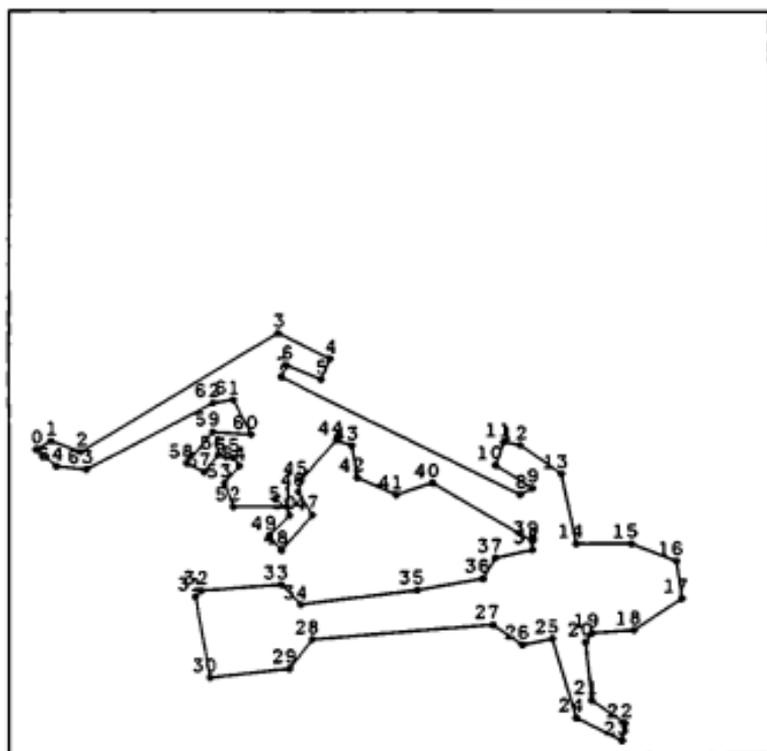


図 5.3: 重なり合った機械部品の多角形近似

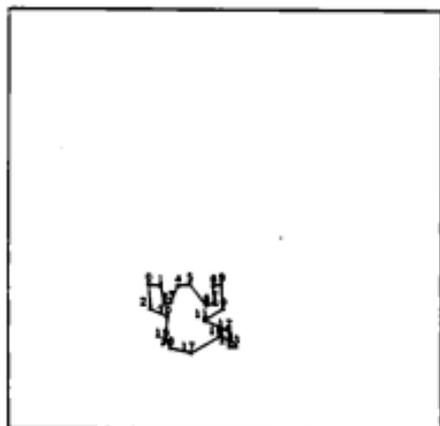
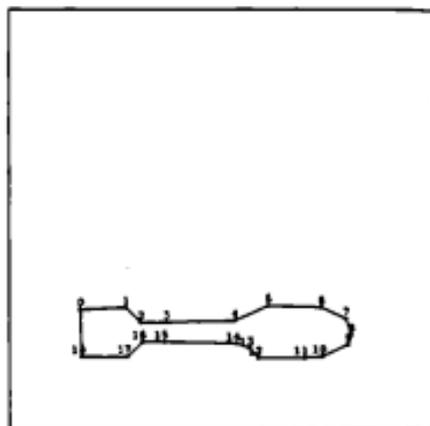
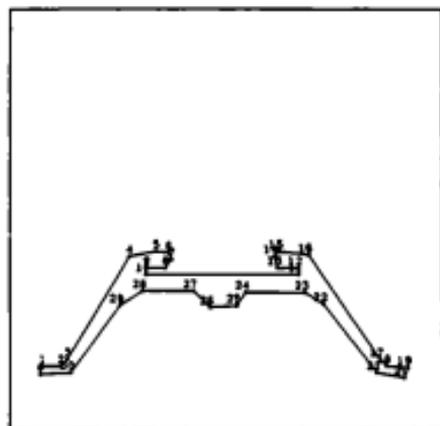


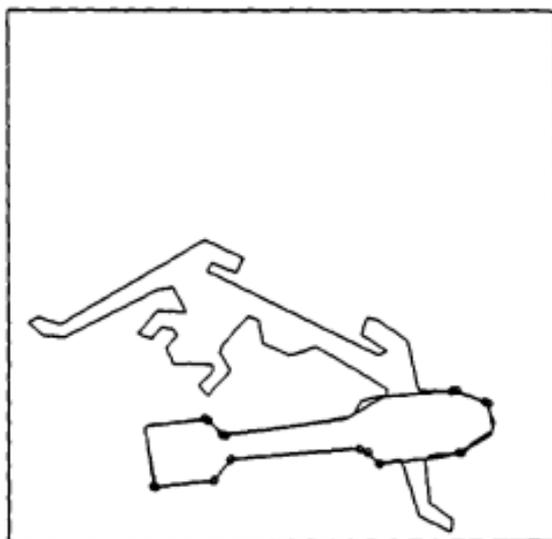
図 5.4: モデルとなる機械部品の多角形近似

(b)

$\delta = 0.05$
 $\theta_{ih} = 15.0$

SIZE = 10
No. of solutions = 1

No. of nodes = 43018
time = 23.34 (Sec.)



(1) $a = 0.000201$ $d = 0.033$:: $c = 0.999$ $b = (0.105, -0.071)$ $R = -6.320$
• 33 34 • • • 15 16 • • 18 • 26 27 • • 28 29 30

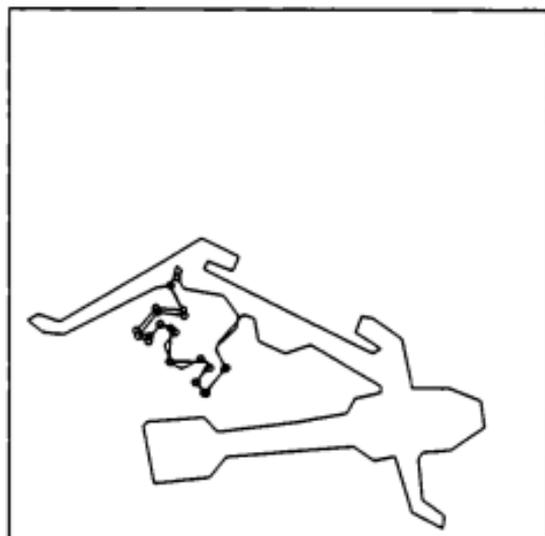
図 5.5: (続き)

(c)

$\delta = 0.05$
 $\theta_{sh} = 15.0$

SIZE = 12
No. of solutions = 2

No. of nodes = 118381
time = 95.99 (Sec.)



```
(1) e = 0.000611 d = 0.047 :: c = 0.988 b = (-0.735, -0.457) R = -139.618
    48 49 * 50 * 52 54 56 57 58 59 60 * * * 61 * * * *
    = 47
(2) e = 0.000646 d = 0.049 :: c = 0.942 b = (-0.731, -0.430) R = -138.174
    48 49 * * 52 53 54 56 57 58 59 60 * * * 61 * * * *
    = 47
```

图 5.5: (续卷)

```

(a) SIZE = 15 No. of solutions = 1 No. of nodes = 29952 time = 19.03
(1) a = 0.000301 mard = 0.031 :: c = 1.014 b = ( 0.102, 0.080) R = 27.145
    0 * 2 * 3 * * * * 6 7 * * 10 * * 13 * 21 23
    * 24 * * 40 41 42 43 * 62 *

(b) SIZE = 10 No. of solutions = 1 No. of nodes = 8129 time = 4.61
(1) a = 0.000201 d = 0.033 :: c = 0.999 b = ( 0.105, -0.071) R = -6.320
    * 33 34 * * * 15 16 * * 18 * 26 27 * * 28 29 30

(c) SIZE = 12 No. of solutions = 1 No. of nodes = 50588 time = 44.91
(1) a = 0.000611 d = 0.047 :: c = 0.988 b = ( -0.735, -0.457) R = -139.518
    48 49 * 50 * 52 54 56 57 58 59 60 * * * 61 * * *
    * 47

```

図 5.6: スケールに関する初期制約を用いたマッチング結果

5.3.2 パラメータ化されたモデルによる形状認識

認識対象がパラメータ化されたモデルで表現される場合について、第4章のマッチングアルゴリズムを適用する。具体的には、互いに重なり合ったり、一部分しか見えないような“はさみ”の形状認識を行う。前節と同様に、TVカメラから入力した画像に対して二値化、境界抽出、多角形近似を行ない、その頂点として点パターンを決定する。ただし二値化に際しては、前節のように判別分析的な手法を用いることなく、人間がその二値化レベルを決定した。入力画像の大きさは 200×320 であり、多角形近似の際の近似許容誤差として3画素分を認めている。また、認識のためのモデルは、第4章の図4.2に示しているパラメータ化された点パターンによって表現された“はさみ”を用いることとする。これも、人間によって作成されたモデルである。図5.7(a)(b)(c)に認識対象の多角形近似結果を示す。(a)では単独の“はさみ”が見えているが、(b)では開き角度の異なる二つの“はさみ”が重なって見えている。また、(c)は、刃の一部がなんらかの原因で観測されなかった場合である。

このようにして得られた多角形近似の各頂点を点パターン内の点としてマッチングを行なう。前節と同様、各点の座標値は、もとの画像の中心を原点として、画像全体が x 軸方向も y 軸方向もちょうど -1.0 と 1.0 の範囲にはいるように正規化している。マッチングの許容誤差 $\delta = 0.05$ とし、各頂点の内角に対する許容誤差 θ_{ih} を 15

度とする。また、パラメータの変化範囲を

$$EI_{\alpha,init} = [0.0, 1.0] \quad (5.4)$$

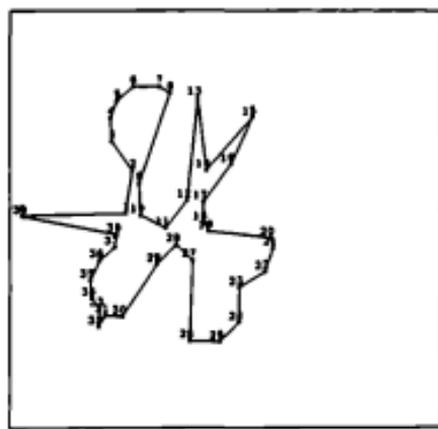
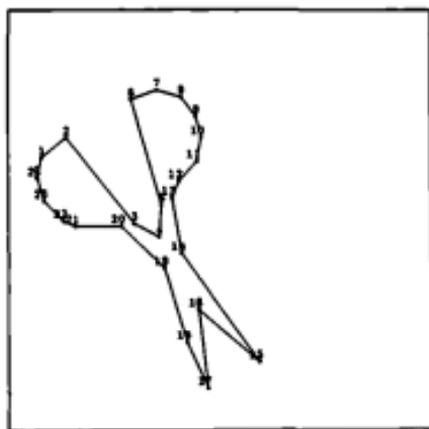
と制限する。認識結果を図 5.8(a)(b)(c) に示す。それぞれ図 5.7(a)(b)(c) を図 4.2 とマッチングさせた結果であり、マッチングの結果得られた相似変換とパラメータ α を用いて、モデルを図の上へ重ね合わせて示している。複数個の解が得られた場合には、そのうち、最大誤差 d が最小の解を示してある。また対応づけに成功した点についてはそれぞれ図中で小さな円で示している。すべての場合についてマッチングは成功し、良好な位置決めも行なわれていることがわかる。パラメータ α の推定値としては、それぞれ 0.13、0.92、0.34 が得られている。マッチングに必要であった計算時間などは図中に示されているが、これは純粋にマッチングに必要であった時間であり、画像処理のための時間は含まれていない。

図 5.8(b) の場合には 3 個の解が得られている。このうち 1 番目と 3 番目の解は対応点が少し異なるだけで、画像中の左側の“はさみ”に対する解であり、2 番目の解だけが右側の“はさみ”に対する解となっている。ここで、パラメータ α の初期制約 $EI_{\alpha,init}$ を適当に選ぶことにより、この 2 番目の解だけを分離・抽出することも可能である。例えば、

$$EI_{\alpha,init} = [0.0, 0.5] \quad (5.5)$$

とすることにより、2 番目の解のみが得られる (図 5.8(d))。このことから、 $EI_{\alpha,init}$ を適当に設定することにより、パラメータ α がある範囲にある対象だけを、容易に分離・抽出可能であることがわかる。

(a)



(c)

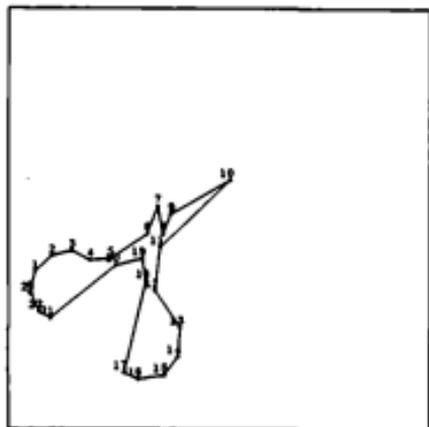
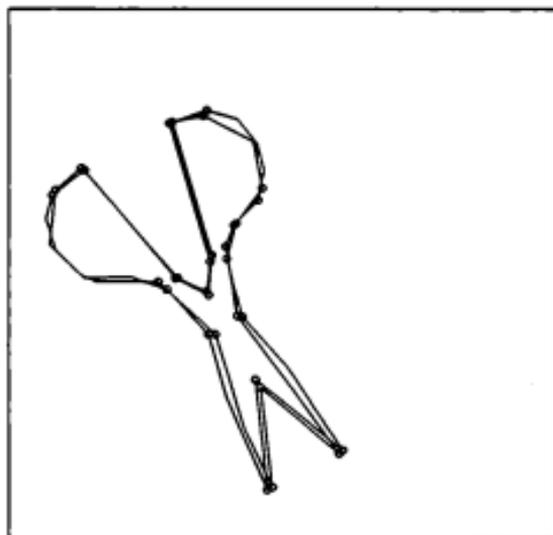


図 5.7: さまざまな“はさみ”の多角形近似

(a)

$\delta = 0.05$
 $\theta_{sk} = 15.0$
 $EI_{\alpha, \text{min}} = [0.0, 1.0]$

SIZE = 16
No. of solutions = 1
No. of nodes = 44712
time = 24.00



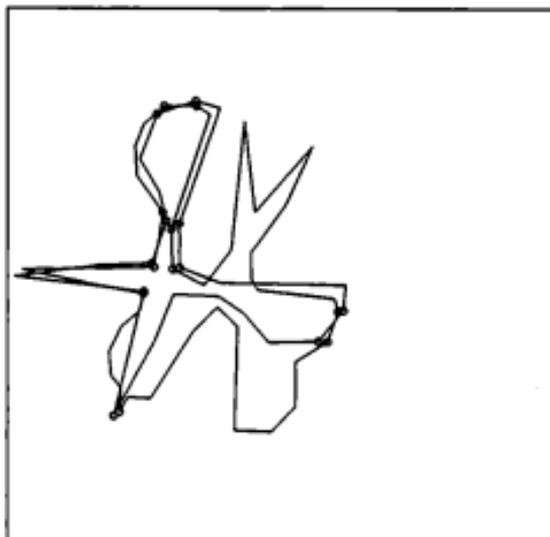
(1) $e = 0.000719$ $d = 0.048793$ $\alpha = 0.134$
 $c = 0.849$ $b = (-0.320, -0.003)$ $R = 152.021$
17 * 19 20 * * * 1 2 3 4 5 6 7 * 11 12 13 14 *
15 16

図 5.8: マッチング結果

(b)

$\delta = 0.05$
 $\theta_{ik} = 15.0$
 $EI_{0, \text{mit}} = [0.0, 1.0]$

SIZE = 10
No. of solutions = 3
No. of nodes = 160056
time = 84.30



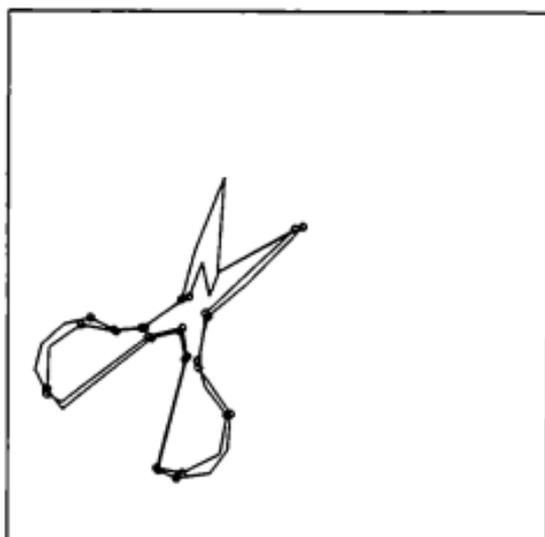
- (1) $e = 0.000786$ $d = 0.038583$ $\alpha = 0.921$
 $c = 0.719$ $b = (-0.323, 0.081)$ $R = -124.796$
* * 1 2 * * * 6 7 * 9 10 * * 21 22 * * * * *
32 38
- (2) $e = 0.000846$ $d = 0.041776$ $\alpha = 0.264$
 $c = 0.707$ $b = (-0.198, -0.085)$ $R = 18.190$
15 16 17 * * * 24 25 26 * * * * 34 * * * * 12 *
13 14
- (3) $e = 0.000817$ $d = 0.045812$ $\alpha = 0.932$
 $c = 0.703$ $b = (-0.327, 0.052)$ $R = -125.893$
* * 1 2 * 3 * 7 * 9 10 * * 21 22 * * * * *
32 38

图 5.8: (続き)

(c)

$\delta = 0.05$
 $\theta_{ik} = 15.0$
 $EJ_{0,inf} = [0.0, 1.0]$

SIZE = 14
No. of solutions = 1
No. of nodes = 84601
time = 45.58



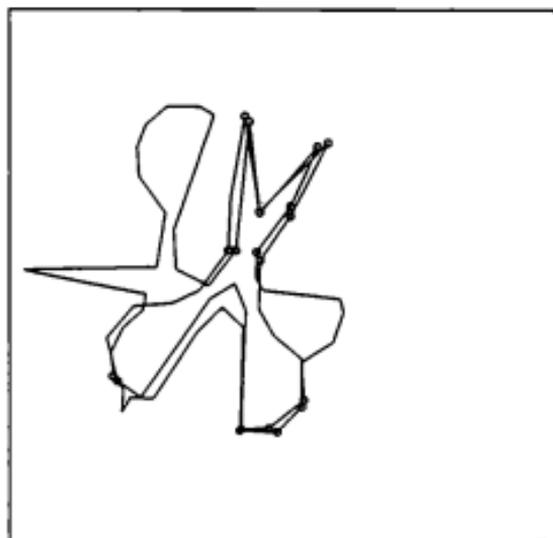
(1) $e = 0.000486$ $d = 0.040985$ $\alpha = 0.344$
 $c = 0.679$ $b = (-0.392, -0.250)$ $k = 32.476$
10 * 11 12 * 13 * 16 17 18 19 20 * 22 * 3 4 5 6 *

図 5.8: (続き)

(d)

$\delta = 0.05$
 $\theta_{ik} = 15.0$
 $EJ_{a,mit} = [0.0, 0.5]$

SIZE = 10
No. of solutions = 1
No. of nodes = 97256
time = 44.45



(1) $a = 0.000845$ $d = 0.041776$ $\alpha = 0.264$
 $c = 0.707$ $b = (-0.198, -0.085)$ $R = 18.190$
15 16 17 * * * 24 25 26 * * * 34 * * * 12 *
13 14

图 5.8: (续)

5.4 むすび

物体の二次元形状の一部からその物体を認識、あるいはシーン内での位置決めを行なう問題に対する、点パターンマッチングアルゴリズムを核とする手法を提案した。木探索によるマッチングは柔軟性が高く、ここで見たようにさまざまな制約を枝刈りの条件として取込むことが非常に容易であり、実験で示したような重なり合ったような物体に対しても、充分精密なマッチングが可能である。

点パターンマッチング手法を、このような形状認識に用いる場合の利点をまとめると、第3章のむすびで述べたことも重複するが、次のような点が挙げられる。

- a) 対象物の一部を他の物体が隠していたり、あるいはカメラの視野の関係から対象物の全体ではなくその一部しか観測できないような場合、つまり部分形状認識に対して適用可能である。
- b) 最小二乗誤差に基づく精密な位置決めが同時に得られる。
- c) 解の意味が明確に定義されており、また定義を満足する解をすべて数え上げることができる。
- d) マッチングを行なう際に設定しなければならないパラメータ数が少ない。最小の場合には許容誤差 δ だけであり、内角による枝刈りを行なったとしても、その閾値 θ_{ik} が加わるだけである。また、それらは解の意味に直接的に関連したものであるため、その設定が比較的容易である。
- e) 相似変換のパラメータの範囲に制約がある場合や、パラメータ化されたモデルの場合など、手法が柔軟で、さまざまな拡張が可能である。

第 6 章

結論

コンピュータビジョンにおける基礎的な課題の一つであるパターンマッチング問題について、対象シーンやモデルが関係・構造記述された場合、および空間記述された場合の両者について、それぞれその問題の持つ固有の特徴・性質を十分に利用したパターンマッチング手法を考察、提案した。その結果を要約すれば次のようになる。

- a) 無向、有向それぞれの場合について、重みつきグラフ間の最適対応問題に対する近似解法を二種与えた。いずれもグラフの隣接行列、あるいは隣接行列から導かれるエルミート行列の固有値展開を利用した解析的手法であり、木探索法のような“組み合わせ的爆発”を伴わず、また大局的な最適化手法となっている。処理に必要とする計算時間は、ほぼグラフの大きさの三乗に比例しており、計算機実験の結果、グラフが完全に同型の場合および同型に非常に近い場合には、ほとんどの場合に正しい最適対応を与えることが判った。また、ある程度同型から外れている場合でも、これらの近似解法に山登り法を併用することにより、良好なマッチング結果が得られることが判った。これは、提案した近似解法を用いて得られる初期対応が、ノード間の大局的な対応関係を良く反映したものとなっていることを示している。さらに、グラフの自己同型写像に関する幾つかの結果も示した。
- b) 点の座標情報のみに基づいた点パターンマッチングアルゴリズムを提案した。提案する手法は木探索を用いており、相似変換（回転、平行移動、スケール変換）

により関連づけられた任意の次元の点パターンに対して適用可能である。許容できるマッチング誤差を定義するため“ δ 制限”という概念を導入し、アルゴリズムは、この条件を満足するすべてのマッチングを生成する。提案する手法の特徴としては、次の点が挙げられる。(1)一般の m 次元空間内の点パターンに対して適用可能である。(2)相似変換のパラメータに制約のある場合や、鏡像パターンを考慮する場合など、手法が柔軟で、さまざまな拡張が可能である。(3)マッチングを行なう際に設定しなければならないパラメータは、許容誤差 δ だけである。また、マッチングに要する計算時間については、計算機実験の結果、次のようなことが言える。(4)ランダムに生成された点パターンをマッチングさせる場合には、ほぼ点数を 1 増やすごとに 2 倍の処理時間を必要とする。(5)全体マッチングの場合には、中位のノイズの場合で、実験した範囲では、ほぼ点数の 3 乗に比例した時間程度でマッチングが可能である。

- c) “はさみ”やさまざまなプロポーシオンを持つ“壺”のように、パラメータ化されたモデルによって表現することが自然な対象を取り扱うため、パラメータ化された点パターンの概念を導入し、点パターンマッチングアルゴリズムを拡張して、パラメータ化された点パターン間の一般的なマッチングアルゴリズムを提案した。
- d) 点パターンマッチングアルゴリズムの応用として、部分形状による認識、位置決め問題を取りあげ、認識対象が通常のモデルで表現可能な場合、およびパラメータ化されたモデルで表現される場合それぞれについて、認識、位置決め実験を行なった。点パターンマッチング手法を、このような形状認識に用いる場合の利点をまとめると、b) で述べたこととも重複するが、次のような点が挙げられる。(1)最小二乗誤差に基づく精密な位置ざめが同時に得られる。(2)解の意味が明確に定義されており、また定義を満足する解をすべて数え上げることができる。(3)マッチングを行なう際に設定しなければならないパラメータ数が少なく、またその設定が比較的容易である。(4)相似変換のパラメータの範囲に制約がある場合や、パラメータ化されたモデルの場合など、手法が柔軟で、さまざまな拡張が可能である。

以上の結果から、コンピュータビジョンにおける様々なパターンマッチング問題に共通すると思われる本質的な問題を取り上げ、それらに対する一般的な解法を数理的な立場から考察するという本論文の目的は、ある程度達成されたものと考えられる。

最後に今後の課題について述べる。まず、構造・関係記述されたパターン間のマッチング問題についてであるが、この場合、本論文で考察した重みつきグラフばかりでなく、より複雑な構造を持つグラフ、例えば、多重ラベルや記号的なラベルを持つグラフなどの最適対応問題を考察する必要がある。また、グラフの大きさが等しい場合ばかりでなく、一方のグラフがもう一方のグラフのサブグラフとなるような場合も重要な問題である。さらに、2.5節ではグラフの隣接行列などの固有値展開を利用してグラフの幾つかの性質についての証明を行なったが、コンピュータビジョンと直接の繋がりはないものの、グラフ理論に対するこのようなアプローチも非常に興味深いものであると思われる [70]。

次に空間記述されたパターン間のマッチングの場合であるが、本論文ではパターンのプリミティブが点の場合のみを考察したが、実際のコンピュータビジョン、特に三次元情報を取り扱う場合には、点パターンばかりでなく、平面や直線などのパターン間のマッチング問題も重要である。三次元の場合の平面（二次元の場合には直線）をプリミティブとしたパターン間のマッチングは、本論文で述べた手法と類似の手法が適用可能であるが [65]、三次元内の直線パターンや、さらに線分のパターン間のマッチング問題は未解決である。

謝辞

本論文は、通商産業省工業技術院電子技術総合研究所に於て筆者が行なってきたコンピュータビジョンに関する研究をまとめたものである。

本研究は電子技術総合研究所の多くの方々の支援のおかげで行なうことができました。特に、横上昭男情報アーキテクチャ部長（前ソフトウェア部長）、田村浩一郎情報科学部長には、終始多大のご支援とご理解を頂きました。ここに心からの謝意を表します。

また、大津農情報数理研究室長はじめ、宮川正弘氏、栗田多喜夫氏、麻生英樹氏、関田巖氏の情報数理研究室の諸氏には、熱心かつ有益なご討論ならびにご協力を頂きました。ここに記して感謝いたします。特に、大津室長には、研究遂行にあたっての心構えと、問題に対する数理的なアプローチの面白さと重要さをご教示いただきました。そのご支援とご指導に深く感謝いたします。

Tony Kasvand 博士（元カナダ国立研究所（NRC）主任研究員、現コンコルディア大学教授）には、筆者がNRC滞在中に行なったコンピュータビジョンの研究に多大なご協力を頂きました。この機会に行なった研究が、本論文の第三章、空間記述されたパターン間のマッチングの基礎となっています。また、このような研究の機会を与えてくださった電総研およびNRCの関係者に感謝いたします。

最後に、本論文をまとめるにあたっては、京都大学工学部克木光彦教授から適切なご指導と、多大なご意見、ご討論を頂きました。ここに、深く感謝の意を表します。

参考文献

- [1] H.Alt, K.Mehlhorn, H.Wagener, and E.Welzl, "Congruence, Similarity and Symmetries of Geometric Objects", ACM 3rd Annual Computational Geometry Conference, pp.308-315, 1987.
- [2] K.S.Arun, T.S.Huang, and S.D.Blostein, "Least-squares fitting of two 3-D points sets", IEEE Trans. Pattern Anal. Machine Intell., vol.PAMI-9, No.5, pp.698-700, 1987.
- [3] M.Athans, "The matrix minimum principle", Information and Control, vol.11, pp.592-606, 1967.
- [4] N. Ayache and O.D. Faugeras, "HYPER: A new approach for the recognition and positioning of two-dimensional objects", IEEE Trans. Pattern Anal. & Mach. Intell., PAMI-8, 1, pp. 44-64, 1986.
- [5] H.S. Baird, *Model-Based Image Matching Using Location*, The MIT Press, 1985.
- [6] D.H. Ballard and C.M. Brown, *Computer Vision*, Prentice-Hall, Inc., 1982.
- [7] D.H. Ballard and D. Sabbah, "Viewer independent shape recognition", IEEE Trans. Pattern Anal. & Mach. Intell., PAMI-5, 6, pp. 653-660, 1983.
- [8] R.Bellman, *Introduction to Matrix Analysis*, McGraw-Hill, p56, 1960.

- [9] A. Ben-Israel and T.N.E. Grevilles, *Generalized Inverses: Theory and Applications*, John Wiley & Sons, 1974.
- [10] B. Bhanu and O.D. Faugeras, "Shape matching of two-dimensional Objects", *IEEE Trans. Pattern Anal. & Mach. Intell.*, PAMI-6, 2, pp.137-156, 1984.
- [11] R. Brooks, "Symbolic reasoning among 3-dimensional models and 2-dimensional images", *Artificial Intelligence*, 17, pp.285-349, 1981.
- [12] C.Y. Chao, "A note on the eigenvalues of a graph", *Journal of Combinatorial Theory*, vol.10, pp.301-302, 1971.
- [13] H.H. Chen and T.S. Huang, "Maximal matching of 3-D points for multiple-object motion estimation", *Pattern Recognition*, vol.21, No.2, pp.75-90, 1988.
- [14] L.S. Davis, "Shape matching using relaxation techniques", *IEEE Trans. Pattern Anal. & Mach. Intell.*, PAMI-1, 1, pp. 60-72, 1979.
- [15] D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*, Academic Press, 1980.
- [16] S.R. Dubois and F. H. Glanz, "An autoregressive model approach to two-dimensional shape classification", *IEEE Trans. Pattern Anal. & Mach. Intell.*, PAMI-8, 1, pp. 55-66, 1986.
- [17] S.A. Dudani, K. J. Breeding, and R. B. McGhee, "Aircraft identification by moment invariants", *IEEE Trans. Comput.*, C-26, 1, pp. 39-45, 1977.
- [18] O.D. Faugeras and K.E. Price, "Semantic description of aerial images using stochastic labeling", *IEEE Trans. Pattern Anal. Machine Intell.*, vol.PAMI-3, No.6, pp.633-642, 1981.
- [19] O.D. Faugeras and M. Hebert, "The representation, recognition, and locating of 3-D objects", *The International Journal of Robotics Research*, vol.4, No.3, pp.27-52, 1986.

- [20] M. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H.Freeman and Company, San Francisco, 1979.
- [21] G. Gati, "Further annotated bibliography on the isomorphism disease", *Journal of Graph Theory*, vol.3, No.2, pp.95-109, 1979.
- [22] D. Ghahraman, A.K.C. Wong, and T. Au, "Graph monomorphism algorithms", *IEEE Trans. on Systems, Man, and Cybernetics*, vol.SMC-10, No.4, pp.189-196, 1980.
- [23] D. Ghahraman, A.K.C. Wong, and T. Au, "Graph optimal monomorphism algorithms", *IEEE Trans. on Systems, Man, and Cybernetics*, vol.SMC-10, No.4, pp.181-188, 1980.
- [24] J.W. Gorman, O. R. Mitchell, and F. P. Kuhl, "Partial shape recognition using dynamic programming", *IEEE Trans. Pattern Anal. & Mach. Intell.*, PAMI-10, 2, pp. 257-266, 1988.
- [25] A. Goshtasby and G. Stockman, "Point pattern matching using convex hull edges", *IEEE Trans. on Systems, Man, and Cybernetics*, vol.SMC-15, No.5, pp.631-637, 1985.
- [26] W.E.L. Grimson and T. Lozano-Pérez, "Localizing overlapping parts by searching the interpretation tree", *IEEE Trans. Pattern Anal. & Mach. Intell.*, PAMI-9, 4, pp.469-482, 1987.
- [27] W.E.L. Grimson, "Recognition of object families using parameterized models", *Proc. of First International Conference on Computer Vision*, pp.93-101, 1987.
- [28] W.E.L. Grimson, "On the recognition of curved objects", *IEEE Trans. Pattern Anal. & Mach. Intell.*, PAMI-11, 6, pp.632-643, 1989.

- [29] F. Harary, "The determinant of the adjacency matrix of a graph", *SIAM Rev.*, vol.4, No.3, pp.202-210, 1962.
- [30] A.J. Hoffman and H.W. Wielandt, "The variation of the spectrum of a normal matrix," *Duke Math. J.* 20, 37-39, 1953.
- [31] B.K.P. Horn, *Robot Vision*, The MIT Press, 1986.
- [32] B.K.P. Horn, "Closed-form solution of absolute orientation using unit quaternions", *J. Opt. Soc. Am. A*, vol.4, No.4, pp.629-642, 1987.
- [33] D.Kahl, A.Rosenfeld, and A.Danker, "Some experiments in point pattern matching", *IEEE Trans. on Systems, Man, and Cybernetics*, vol.SMC-10, pp.105-116, 1980.
- [34] S. Kasif, L. Kitchen, and A. Rosenfeld, "A Hough transform technique for subgraph isomorphism", *Pattern Recognition Letters*, vol.2, No.2, pp.83-88, 1983.
- [35] L. Kitchen and A. Rosenfeld, "Discrete relaxation for matching relational structures," *IEEE Trans. on Systems, Man, and Cybernetics*, vol.SMC-9, No.12, pp.869-874, Dec. 1979.
- [36] L. Kitchen, "Relaxation applied to matching quantitative relational structures," *IEEE Trans. on Systems, Man, and Cybernetics*, vol.SMC-10, No.2, pp.96-101, Feb. 1980.
- [37] 児玉慎三, 須田信英, "システム制御のためのマトリクス理論", 計測自動制御学会, 1978.
- [38] D. Lavine, B.A. Lambird, and L. Kanal, "Recognition of spatial point patterns", *Pattern Recognition*, vol.16, No.3, pp.289-295, 1983.
- [39] 守田了, 川嶋勉夫, 青木由直, "輪郭線形状の階層的記述によるマッチング", 電子情報通信学会パターン認識・理解研究会, PRU88-142, 1988.

- [40] A. Mowshowitz, "The group of a graph whose adjacency matrix has all distinct eigenvalues", in *Proof Techniques in Graph Theory*, pp.109-110, Academic Press, New York, 1969.
- [41] H. Ogawa, "Labeled point pattern matching by fuzzy relaxation", *Pattern Recognition*, vol.17, No.5, pp.569-573, 1984.
- [42] H. Ogawa, "Labeled point pattern matching by Delaunay triangulation and maximal cliques", *Pattern Recognition*, vol.19, No.1, pp.35-40, 1986.
- [43] N. Otsu, "A threshold selection method from grey-level histogram", *IEEE Trans. Syst. Man & Cybern.*, SMC-9, 1, pp.62-66, 1979.
- [44] C.H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithm and Complexity*, Prentice-Hall Inc., 1982.
- [45] T. Pavlidis and F. Ali, "A hierarchical syntactic shape analyzer", *IEEE Trans. Pattern Anal. & Mach. Intell.*, PAMI-1, 1, pp. 2-9, 1979.
- [46] T. Pavlidis, "The use of a syntactic shape analyzer for contour matching", *IEEE Trans. Pattern Anal. & Mach. Intell.*, PAMI-1, 3, pp. 307-310, 1979.
- [47] T. Pavlidis, *Structural Pattern Recognition*, New York: Springer-Verlag, 1977.
- [48] W. A. Perkins, "A model-based vision system for industrial parts", *IEEE Trans. Comput.*, C-27, 2, pp. 126-143, 1978.
- [49] S.Ranade and A.Rosenfeld, "Pattern matching by relaxation", *Pattern Recognition*, No. 12, pp.269-275, 1980.
- [50] R.C. Read and D.G. Corneil, "The graph isomorphism disease", *Journal of Graph Theory*, vol.1, No.4, pp.339-363, 1977.

- [51] A. Rosenfeld, "Rapporteur for sensing systems", Workshop on the Research Needed to Advance the State of Knowledge in Robotics, Newport, RI, pp.198-205, 1980.
- [52] L.G. Shapiro and R.M. Haralick, "Structural descriptions and inexact matching", IEEE Trans. Pattern Anal. Machine Intell., vol.PAMI-3, No.5, pp.504-519, 1981.
- [53] L.G. Shapiro and R.M. Haralick "A metric for comparing relational descriptions", IEEE Trans. Pattern Anal. Machine Intell., vol.PAMI-7, No.1, pp.90-94, 1985.
- [54] 白井良明編, パターン理解, 如工学講座 9, オー・A社, 1987.
- [55] J.C.Simon, A.Chetroun, and C.Roché, "A method of comparing two patterns independent of possible transformations and small distortions", Pattern Recognition, No. 4, pp.73-81, 1972.
- [56] 竹中淑子, 線形代数的グラフ理論, 情報処理シリーズ 13, 培風館, 1989.
- [57] W.H. Tsai and K.S. Fu, "Error-correcting isomorphisms of attributed relational graphs for pattern analysis," IEEE Trans. on Systems, Man, and Cybernetics, vol.SMC-9, No.12, pp. 757-768, 1979.
- [58] W.H. Tsai and K.S. Fu, "Subgraph error-correcting isomorphisms for syntactic pattern recognition," IEEE Trans. on Systems, Man, and Cybernetics, vol.SMC-13, No.1, pp.46-62, 1983.
- [59] J. Turner, "Generalized matrix functions and the graph isomorphism problem", SIAM J. Appl. Math., vol.16, No.3, pp.520-526, 1968.
- [60] J.R. Ullmann, "An algorithm for subgraph isomorphism", J. Assoc. Comput. Mach., vol.23, No.1, pp.31-42, 1976.

- [61] 梅山伸二, “固有値展開を用いた重み付きグラフ間のマッチングについて”, 昭和61年度電子通信学会総合全国大会, 1593, 1986.
- [62] 梅山伸二, “重み付きグラフ間のマッチング問題に対する固有値展開を用いた近似解法”, 電子通信学会パターン認識・理解研究会, PRU87-100, 1986.
- [63] S.Umeyama, “Metrics for comparing structural descriptions”, 電子技術総合研究所彙報, No.50, vol.7, 1986.
- [64] S.Umeyama, “Weighted graph matching algorithms using eigen-decomposition approach”, The Transactions of the IEICE, vol.E 70, No.9, 1987.
- [65] S.Umeyama, T.Kasvand, “A new algorithm for point and plane pattern matching”, NRCC report, NRC/ERB-1009, 1987.
- [66] 梅山伸二, T.Kasvand, “点パターンマッチングアルゴリズム”, 電子情報通信学会パターン認識・理解研究会, PRU87-100, 1988.
- [67] S. Umeyama, “An Eigendecomposition approach to weighted graph matching problems”, IEEE Trans. Pattern Anal. Machine Intell., PAMI-10, 5, pp. 695-703, 1988.
- [68] 梅山伸二, “点パターンマッチングアルゴリズム”, 電子情報通信学会論文誌 D-II Vol.J-72-D-II No.2 pp.218-228, 1989.
- [69] 梅山伸二, “点パターンマッチングアルゴリズムを用いた重なり合った物体の認識”, 電子情報通信学会春期全国大会, D-567, 1989.
- [70] 梅山伸二, “グラフの同型写像の線形代数的手法を用いた表現について”, 電子情報通信学会春期全国大会, 1990.
- [71] J.H. Wilkinson, *The Algebraic Eigenvalue Problem*, Monographs on Numerical Analysis, Oxford University Press, 1965.
- [72] A.K.C.Wong, “An algorithm for constellation matching”, Proc. of the 8th International Conference of Pattern Recognition, vol. 1, pp.546-554, 1986.

- [73] M. You and A.K.C. Wong, "An algorithm for graph optimal isomorphism," Proc. 1984 ICPR, pp.316-319, 1984.
- [74] C. T. Zahn and R. Z. Roskies, "Fourier descriptors for plane closed curves", IEEE Trans. Comput., C-21, 3, pp. 269-281, 1972.
- [75] 張鴻賓, 美濃導彦, 池田克夫, "輪郭線のマッチングに基づく重なり合った物体の認識", 電子情報通信学会パターン認識・理解研究会, PRU88-104, 1988.

付録 A

定理 2.1 の証明

ここでは、第 2 章の定理 2.1 の証明を行なう。

ユークリッドノルムはユニタリ変換に対して不変なため、

$$\begin{aligned}
 f(W) &= \|WAW^* - U_B \Lambda_B U_B^*\|^2 \\
 &= \|U_B(U_B^*WAW^*U_B - \Lambda_B)U_B^*\|^2 \\
 &= \|U_B^*WAW^*U_B - \Lambda_B\|^2
 \end{aligned} \tag{A.1}$$

である。そこでまず、 $f(W)$ の最小値は $X = U_B^*WAW^*U_B$ が対角行列のときに達成されることを示す。 X の p 行 q 列 ($p \neq q$) の要素 $x \neq 0$ であるとする。このとき、 p 行、 q 行、 p 列、 q 列の交差する部分では Λ_B および X は例えば次のようになっている。

| Λ_B | | | | X | | | |
|-------------|-----------|---|-----------|-------|-------|---|-------|
| col.p | col.q | | | col.p | col.q | | |
| ⋮ | ⋮ | | | ⋮ | ⋮ | | |
| ⋯ | β_p | ⋯ | 0 | ⋯ | a | ⋯ | x |
| | | ⋯ | row p | | | ⋯ | row p |
| ⋮ | ⋮ | | | ⋮ | ⋮ | | |
| ⋯ | 0 | ⋯ | β_q | ⋯ | x* | ⋯ | b |
| | | ⋯ | row q | | | ⋯ | row q |
| ⋮ | ⋮ | | | ⋮ | ⋮ | | |

ここで、 X はエルミート行列であるからその対角要素 a 、 b は実数であることに注意する。さて、ユニタリ行列 T を

$$\begin{aligned}
 T &= [t_{ij}] & (A.2) \\
 t_{pp} &= e^{i\phi} \cos \theta, \quad t_{pq} = \sin \theta \\
 t_{qp} &= -\sin \theta, \quad t_{qq} = e^{-i\phi} \cos \theta \\
 t_{ii} &= 1 \quad (i \neq p, q), \\
 t_{ij} &= 0 \quad \text{otherwise}
 \end{aligned}$$

とすると、 ϕ 、 θ を適当に定めることにより

$$g(T) = \|T^*XT - \Lambda_B\|^2 - \|X - \Lambda_B\|^2 < 0 \quad (A.3)$$

とできることを示す。 T^*XT の p 行、 q 行、 p 列、 q 列の交差する部分での要素をそれぞれ a' 、 b' 、 x' 、 x'' (a' 、 b' は実数) とすると

$$\begin{aligned}
 g(T) &= -2a'\beta_p - 2b'\beta_q + 2a\beta_p + 2b\beta_q \\
 &= 2(a - a')\beta_p + 2(b - b')\beta_q
 \end{aligned} \quad (A.4)$$

である。この他の項はすべて相殺される。また (A.2) 式から

$$a' = a \cos^2 \theta - (x^* e^{i\phi} + x e^{-i\phi}) \sin \theta \cos \theta + b \sin^2 \theta \quad (A.5)$$

$$b' = a \sin^2 \theta + (x^* e^{i\phi} + x e^{-i\phi}) \sin \theta \cos \theta + b \cos^2 \theta \quad (A.6)$$

$x = r e^{i\psi}$ ($r \neq 0$) とすると、

$$x^* e^{i\phi} + x e^{-i\phi} = 2r \cos(\phi - \psi) \quad (A.7)$$

$\phi = \psi$ と定めると、これは $2r$ に等しい。よって、

$$a' = a \cos^2 \theta - 2r \sin \theta \cos \theta + b \sin^2 \theta \quad (A.8)$$

$$b' = a \sin^2 \theta + 2r \sin \theta \cos \theta + b \cos^2 \theta \quad (A.9)$$

(A.4) 式に代入して、

$$\begin{aligned} h(\theta) = g(T) &= 2\beta_p\{(a-b)\sin^2\theta + r\sin 2\theta\} + 2\beta_q\{(b-a)\sin^2\theta - r\sin 2\theta\} \\ &= P\sin^2\theta + Q\sin 2\theta \end{aligned} \quad (\text{A.10})$$

ここで

$$P = 2(\beta_p - \beta_q)(a - b) \quad (\text{A.11})$$

$$Q = 2r(\beta_p - \beta_q) \quad (\text{A.12})$$

$r \neq 0$, $\beta_p \neq \beta_q$ から $Q \neq 0$ である。 $\theta = 0$ のとき $h(\theta) = 0$ であり、また

$$\frac{d}{d\theta}h(\theta) = P\sin 2\theta + 2Q\cos 2\theta \quad (\text{A.13})$$

であるから、 $\theta = 0$ の点での $h(\theta)$ の微係数は $2Q \neq 0$ であり、このことから θ を適当に選ぶことにより $g(T) > 0$ にも $g(T) < 0$ にもできることが言える。ゆえに $f(W)$ の最小値は X が対角行列のときに達成されることがわかった。さて、 $X = U_B^* W A W^* U_B$ の固有値は A の固有値と一致するため、 $\alpha_1, \alpha_2, \dots, \alpha_n$ を適当に並べかえたものを $\alpha_{p_1}, \alpha_{p_2}, \dots, \alpha_{p_n}$ とすると、 X が対角行列のときには

$$X = \text{diag}(\alpha_{p_i}) \quad (\text{A.14})$$

である。よって、

$$f(W) = \sum_{i=1}^n (\alpha_{p_i} - \beta_i)^2 \quad (\text{A.15})$$

(A.15) 式は明らかに $\alpha_{p_1} = \alpha_1, \alpha_{p_2} = \alpha_2, \dots, \alpha_{p_n} = \alpha_n$ のときに最小となる。つまり

$$X = U_B^* W U_A \Lambda_A U_A^* W^* U_B = \Lambda_A \quad (\text{A.16})$$

である ($\Lambda = U_A \Lambda_A U_A^*$ を用いた)。この式の両辺に左から $U_A^* W^* U_B$ をかけると、

$$\Lambda_A U_A^* W^* U_B = U_A^* W^* U_B \Lambda_A \quad (\text{A.17})$$

Λ_A は相異なる要素からなる対角行列であり、 $U_A^* W^* U_B$ はその固有ベクトル行列であると考えられることができるから、明らかに

$$U_B^* W U_A = S, S \in S \quad (\text{A.18})$$

$$S = \{\text{diag}(s_i) \mid s_i \text{ is any complex number satisfying } |s_i| = 1\}$$

となる。これから定理が導かれる。また最小値が $\sum_{i=1}^n (\alpha_i - \beta_i)^2$ であることも明らかである。

付録 B

定理 3.2 の証明

ここでは、第3章の定理 3.2の証明を行なう。

まず、定理の証明を行なう前に次の補題をしめす。

[補題] A, B を $m \times n$ の行列、 R を $m \times m$ の回転行列、そして UDV^T ($UU^T = VV^T = I, D = \text{diag}(d_i), d_1 \geq d_2 \geq \dots \geq d_m$) を AB^T の特異値分解であるとす。この時、

$$\min_R \|A - RB\|^2 = \|A\|^2 + \|B\|^2 - 2\text{tr}(DS) \quad (\text{B.1})$$

ここで

$$S = \begin{cases} I & \text{if } \det(AB^T) \geq 0 \\ \text{diag}(1, 1, \dots, 1, -1) & \text{if } \det(AB^T) < 0 \end{cases} \quad (\text{B.2})$$

また、 $\text{rank}(AB^T) = m$ のとき、上記の最小値を与える R は次のように定まる。

$$R = USV^T. \quad (\text{B.3})$$

証明: 目的関数 F を次のように定める。

$$F = \|A - RB\|^2 + \text{tr}(L(R^T R - I)) + g(\det(R) - 1) \quad (\text{B.4})$$

ここで g はラグランジュ乗数、 L は対称ラグランジュ乗数行列であり、それぞれ R が直交でその行列式が1に等しい、つまり回転行列であるという条件を示している。

この F を R, L, g について微分 [3, 37] すると次のような式が得られる。

$$\frac{\partial F}{\partial R} = -2AB^T + 2RBB^T + 2RL + gR = 0 \quad (\text{B.5})$$

$$\frac{\partial F}{\partial L} = R^T R - I = 0 \quad (\text{B.6})$$

$$\frac{\partial F}{\partial g} = \det(R) - 1 = 0 \quad (\text{B.7})$$

ここで、 R は回転行列であるから

$$\frac{\partial}{\partial R} \det(R) = \text{adj}(R^T) = \det(R^T)(R^T)^{-1} = R \quad (\text{B.8})$$

であることを用いている。(B.5)式から

$$RL' = AB^T, \quad \text{where } L' = BB^T + L + \frac{1}{2}gI \quad (\text{B.9})$$

この式の両辺の転置をとると、

$$L'R^T = BA^T \quad (\text{B.10})$$

(B.10)式の各辺を(B.9)式の各辺へそれぞれ左からかけると (L' が対称行列であり、 $R^T R = I$ であることに注意する) 次式となる。

$$L^2 = BA^T AB^T = VD^2V^T \quad (\text{B.11})$$

明らかに、 L' と L^2 は可換であるから ($L'L^2 = L^2L'$) 共に同じ直交行列により対角形式に変換できるはずである [8]。そこで (B.11)式から

$$L' = VDSV^T, \quad (\text{B.12})$$

$$\text{where } S = \text{diag}(s_i), \quad s_i = 1 \text{ or } -1$$

とできる。(B.12)式から

$$\begin{aligned} \det(L') &= \det(VDSV^T) \\ &= \det(V) \det(D) \det(S) \det(V^T) \\ &= \det(D) \det(S). \end{aligned} \quad (\text{B.13})$$

一方、(B.9) から

$$\begin{aligned}\det(L') &= \det(R^T A B^T) \\ &= \det(R^T) \det(A B^T) \\ &= \det(A B^T)\end{aligned}\tag{B.14}$$

そこで

$$\det(D) \det(S) = \det(A B^T)\tag{B.15}$$

特異値は非負であるから、 $\det(D) = d_1 d_2 \dots d_m \geq 0$ 。ゆえに $\det(A B^T) > 0$ の時、 $\det(S) = 1$ 、 $\det(A B^T) < 0$ の時、 $\det(S) = -1$ でなければならない。

次に $\|A - RB\|^2$ の極値は次のように求められる。まず (B.9) 式から次式が成立する。

$$\begin{aligned}\|A - RB\|^2 &= \|A\|^2 + \|B\|^2 - 2\text{tr}(A B^T R^T) \\ &= \|A\|^2 + \|B\|^2 - 2\text{tr}(R^T A B^T) \\ &= \|A\|^2 + \|B\|^2 - 2\text{tr}(L')\end{aligned}\tag{B.16}$$

(B.12) 式を (B.16) 式へ代入して、

$$\begin{aligned}\|A - RB\|^2 &= \|A\|^2 + \|B\|^2 - 2\text{tr}(V D S V^T) \\ &= \|A\|^2 + \|B\|^2 - 2\text{tr}(D S) \\ &= \|A\|^2 + \|B\|^2 - 2(d_1 s_1 + d_2 s_2 + \dots + d_m s_m)\end{aligned}\tag{B.17}$$

この式から、 $\|A - RB\|^2$ の最小値は、 $\det(A B^T) \geq 0$ ならば $s_1 = s_2 = \dots = s_m = 1$ の時、 $\det(A B^T) < 0$ ならば $s_1 = s_2 = \dots = s_{m-1} = 1, s_m = -1$ のとき達成されることがわかる。

最後に、もし $\text{rank}(A B^T) = m$ ならば、 L' は正則であるからその逆行列 $L'^{-1} = (V D S V^T)^{-1} = V S^{-1} D^{-1} V^T = V D^{-1} S V^T$ ($S^{-1} = S, S D^{-1} = D^{-1} S$ であることに注意) が存在する。そこで (B.9) 式から

$$R = A B^T L'^{-1} = U D V^T V D^{-1} S V^T = U S V^T.\tag{B.18}$$

これで補題は証明された。

次に、この補題を用いて定理を証明する。

集合 X_{Φ} 、 Y_{Φ} を $m \times n$ 行列 $X = [x_1, x_2, \dots, x_n]$ 、 $Y = [y_1, y_2, \dots, y_n]$ で表す。これで (3.24) 式の $e^2(R, t, c)$ は、

$$e^2(R, t, c) = \frac{1}{n} \| Y - cRX - th^T \|^2 \quad (\text{B.19})$$

と表現される。ここで

$$h = (1, 1, \dots, 1)^T \quad (\text{B.20})$$

である。ここで行列 $K = I - (1/n)hh^T$ ($K^2 = K^T = K$) を用いると、(3.28) (3.29) (3.30) 式の統計量は次のようになる。

$$\sigma_x^2 = \frac{1}{n} \| XK \|^2 \quad (\text{B.21})$$

$$\sigma_y^2 = \frac{1}{n} \| YK \|^2 \quad (\text{B.22})$$

$$\Sigma_{xy} = \frac{1}{n} YKX^T \quad (\text{B.23})$$

さらに、次式をもちいると、

$$X = XK + \frac{1}{n} Xhh^T \quad (\text{B.24})$$

$$Y = YK + \frac{1}{n} Yhh^T \quad (\text{B.25})$$

誤差 $e^2(R, t, c)$ は次式のように変形される。

$$\begin{aligned} e^2(r, t, c) &= \frac{1}{n} \| YK + \frac{1}{n} Yhh^T - cRXK - \frac{c}{n} RXhh^T - th^T \|^2 \\ &= \frac{1}{n} \| YK - cRXK + (\frac{1}{n} Yh - \frac{c}{n} RXh - t)h^T \|^2 \\ &= \frac{1}{n} \| YK - cRXK - t'h^T \|^2 \\ &= \frac{1}{n} (\| YK - cRXK \|^2 + \| t'h^T \|^2 \\ &\quad - 2\text{tr}(K(Y^T - cX^T R^T)t'h^T)) \end{aligned} \quad (\text{B.26})$$

ここで

$$t' = -\frac{1}{n}Yh + \frac{c}{n}RXh + t \quad (\text{B.27})$$

である。ここで、次の2式から

$$\begin{aligned} \text{tr}(K(Y^T - cX^T R^T)t'h^T) &= \text{tr}(h^T(I - \frac{1}{n}hh^T)(Y^T - cX^T R^T)t') \\ &= \text{tr}((h^T - h^T)(Y^T - cX^T R^T)t') \\ &= 0 \end{aligned} \quad (\text{B.28})$$

$$\|t'h^T\|^2 = n\|t'\|^2 \quad (\text{B.29})$$

次式が成立する。

$$e^2(R, t, c) = \frac{1}{n} \|YK - cRXK\|^2 + \|t'\|^2 \quad (\text{B.30})$$

この式から、 $e^2(R, t, c)$ を最小化するためには $t' = 0$ でなければならないことがわかる。すなわち、

$$t = \frac{1}{n}Yh - \frac{c}{n}RXh = \mu_y - cR\mu_x \quad (\text{B.31})$$

である。

次に、 UDV^T が $\Sigma_{xy} = (1/n)YKX^T$ の特異値分解の時、

$$YK(cXK)^T = cYK K^T X^T = cYKX^T \quad (\text{B.32})$$

の特異値分解は $cnUDV^T$ となる。そこで補題より $(1/n)\|YK - cRXK\|^2$ の R に関する最小値 $e^2(c)$ は次のように定まる。

$$\begin{aligned} e^2(c) &= \frac{1}{n} \{ \|YK\|^2 + \|cXK\|^2 - 2\text{tr}(cnDS) \} \\ &= \sigma_y^2 + c^2\sigma_x^2 - 2\text{ctr}(DS) \end{aligned} \quad (\text{B.33})$$

ここで

$$S = \begin{cases} I & \text{if } \det(\Sigma_{xy}) \geq 0 \\ \text{diag}(1, 1, \dots, 1, -1) & \text{if } \det(\Sigma_{xy}) < 0 \end{cases} \quad (\text{B.34})$$

である。同様に補題より、 $\text{rank}(YKX^T) = \text{rank}(\Sigma_{xy}) = m$ の時、最適な回転行列は

$$R = USV^T \quad (\text{B.35})$$

である。以上で定理が証明された。