

中心軸変換を用いたシルエット図形の構造化

影山 勝彦[†] 大町真一郎[†] 阿曾 弘具[†]

[†] 東北大学大学院工学研究科 〒980-8579 宮城県仙台市青葉区荒巻字青葉 6-6-05

E-mail: †{kage,machi}@aso.ecei.tohoku.ac.jp, aso@ecei.tohoku.ac.jp

あらまし インターネットの発展による画像データ検索の要望が高まっていることに代表されるように、多くの画像を適切に識別することが望まれている。しかし、動物等の非剛体の画像を対象とする場合、形状の変形等があり、通常のテンプレートマッチングではそれに対応することが困難である。本論文では、構造の変化に柔軟に対応するためシルエット図形を対象とし、図形を階層的なパーツに分解し、グラフマッチングを利用して識別する手法を提案する。まず、中心軸を得る過程で得られる内接円の半径の変化に着目し、シルエット図形を階層的な構造に分解する。そして、これらの階層的な構造を考慮し、構造変化に頑健なグラフマッチングのコストを定義する。提案手法を画像に適用し、画像のノイズ、構造的変化、スケール変化等について頑健であり、実際の画像識別に有効であることを示す。
キーワード 中心軸変換, グラフマッチング, シルエット図形, 画像識別

Structurization of Silhouette Images Using Medial Axis Transforms

Katsuhiko KAGEYAMA[†], Shinichiro OMACHI[†], and Hirotomo ASO[†]

[†] Graduate school of Engineering, Tohoku University

Aramaki-aza-Aoba 6-6-05, Aoba-ku, Sendai-shi, Miyagi, 980-8579 Japan

E-mail: †{kage,machi}@aso.ecei.tohoku.ac.jp, aso@ecei.tohoku.ac.jp

Abstract It is desired that various images can be recognized correctly. However, it is difficult to deal with non-rigid objects for recognition by a simple template matching method because of the change in shape. In this paper, we focus on silhouette figure images and propose a method that can flexibly deal with the change in shape by decomposing the figure into the parts hierarchically by using medial axis transforms. Experimental results demonstrate its robustness in the presence of noise, structure change and scale transformation and show the effectivity of the shape recognition.

Key words Medial Axis Transformation, Graph Matching, Silhouette Figure Image, Image Recognition

1. はじめに

画像中の物体を認識することはコンピュータビジョンの分野において主要な問題である。また近年ではネットワークの発展により Web 上などに散在する画像の検索への要望も強まっている。

人間は三次元物体を認識する場合に二次元平面への投影を行う。その平面画像をどのように認識するのかという問題について、認知心理学の分野において、Rosch らが「概念のカテゴリはプロトタイプからの類似性によって構成される」というモデルを提案している [1]。これは具体的には、人間が動物を見たときに、「これは三毛猫である」または「これはシャム猫である」ということを判断する前に、まずその動物を「猫」という大まかなカテゴリに分類した上で、更に細かい特徴を見て、種類やその個体を認識していくという過程にあてはまる。

この考えに基づいた物体の認識の第 1 段階として、物体を平面に投影した二値シルエット図形をカテゴリに分類することが本研究の目的である。認識の過程のイメージ図を図 1 に示す。また Rosch らが提案したカテゴリ分類のイメージを図 2 に示す。図 2 の分類は人間がシルエットから図形を判断できる限界を示している（「シルエットから猫の種類は分からない」）

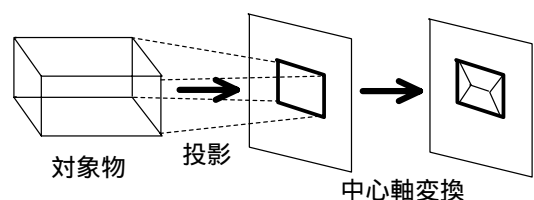


図 1 シルエットの投影

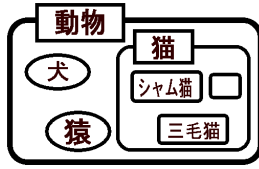


図2 カテゴリの分類について

カテゴリに分類するためには、そのカテゴリのプロトタイプからの類似度を計算する必要がある。類似度を求めるときに用いる特徴には主に以下の特性が求められる。

- パーツの曲げや伸びなどの本質的ではない構造変化に対する頑健性
- 回転に対する不変性
- スケールに対する不変性
- オクルージョンに対する頑健性

例えば単純なテンプレートマッチングは回転やスケールの変化には幾何学的な変形に対応できるが、構造変化やオクルージョンへの対応は困難である。

構造変化を考慮して二つの図形の類似度を計算するために用いる特徴は、これまでも提案されており、その一つに輪郭情報を用いる方法がある。医療画像の分野では二つの画像の輪郭線の弾性マッチングコストから類似度を求める方法 [2] などがあるが、複雑な構造を持つ図形にそのまま適用はできない。

もう一つの方法として中心軸による表現を用いるものがある [3]。中心軸とは図形の輪郭線からの距離が釣り合う点の集合であり、骨格 (Skeleton) と呼ばれることもある。中心軸を用いることで図形の構造を的確に表現することが可能で、軸線によって作られるグラフのトポロジは、基本的にパーツの折れ曲がり、回転、スケールの変化に対して不変である。問題点としては一般的に知られる中心軸変換の手法は中心軸の連結性、心線化を保証していない点や、輪郭のノイズによって中心軸のグラフのトポロジが大きく変化しやすい点などがある。しかしそのような難点を克服できれば、グラフマッチングの問題として類似度計算を扱うことが可能になり、中心軸は図形の構造を表現する良い手法と言えよう。

本論文では、シルエット図形のカテゴリ分けを目標とし、図形の構造変化に柔軟に対応できるグラフマッチングを用いた整合法を提案する。まず、シルエット図形の中心軸を得る過程で得られる内接円の半径の変化に着目し、シルエット図形を階層的な構造に分解する手法を提案する。階層的な構造とは、例えば動物であれば胴体 - 腕 - 手 - 指といった構造を指す。そして、これらの階層的な構造を考慮した、構造変化に頑健なグラフマッチングの手法を提案する。

2. グラフを用いた類似度の計算

本節では二つのシルエット図形の類似度の計算全体の手順の概要を述べる。類似度を計算する全体の流れは次のようになっている。

- (1) シルエット画像の中心軸変換
- (2) 中心軸の木グラフ化
- (3) グラフへの重み付け
- (4) グラフマッチング

各ステップについて、以下に述べる。(3)については提案手法として3節で詳細を解説する。

2.1 中心軸変換

まずシルエット図形をグラフとして表現するために、中心軸変換を行う。古典的な方法では連結が保たれないため、Siddiqiらの手法 [4] を用いた。この手法は中心軸を求める古典的な草焼きのイメージ [3] を閉曲面内における流体の保存則方程式の特異点を求めるという一般的な形で定式化したものであり、以下に述べる内接円の半径関数による表現と概念的に相性が良いアルゴリズムである。

平面上に流れがあるとき、任意の点で以下の微分方程式が成り立つ。

$$\frac{\partial u}{\partial t} + \nabla \cdot f(u) = 0 \quad (1)$$

ここで t は時間、 u は流量、 $f(u)$ は流束のベクトルである。これは微小領域での流量の変化と溢れ出す流束の変化の和は0で一定であることを示す。輪郭線上のすべての点を流束の湧き出し点として、そこから流体が流れ出して等速で広がっていくとすると、中心軸に対応する点で流れが衝突する。この地点は流

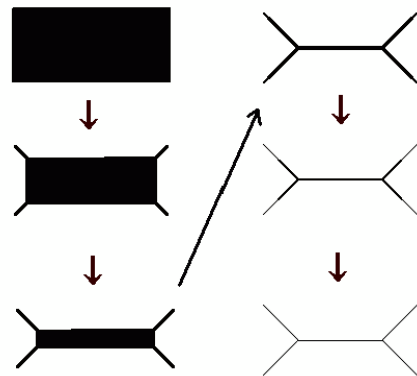


図3 中心軸変換の過程

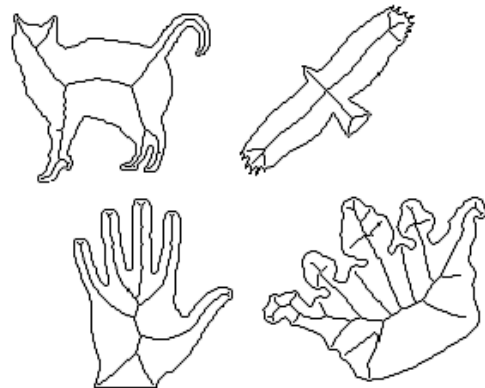


図4 中心軸変換の例

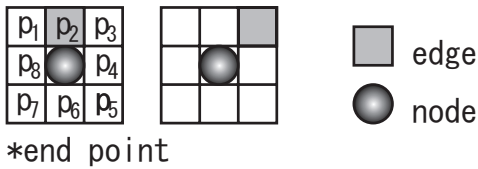


図5 ノードの例

束が流れていけず流れ込む点であるので $\nabla \cdot f(u) < 0$ となる。これは輪郭線から等距離の点であるので、中心軸の点である。しかしこれだけの条件では連結性が保証されないため、 $\nabla \cdot f(u)$ が大きい順に処理を行い、連結を切断しないように細線化の判定を行う。その過程と結果の例を図3に示す。これは判定過程のステップを順に追ったものであり、黒い部分が考慮中の領域で、白い部分が中心軸ではないと判定された領域である。外周から徐々に内部へ考慮点が進行しているのが分かり、輪郭線から流れが進行していく様子に対応している。最終的にこれ以上考慮すべき点が無くなった時点でアルゴリズムは終了する。

実際に中心軸を抽出した例を図4に示す。安定して中心軸を抽出できていることが分かる。一部不要と思われる小さな枝も見られたが、輪郭が滑らかであれば、過度に生じることは無かった。

一般的に、中心軸変換において輪郭のノイズによって本来存在するべきではない軸が生じてしまう問題が知られている。対策として輪郭線にスムージングを施したり、輪郭との対応を考慮することで枝刈りを行う方法[5]や本節の手法であれば $\nabla \cdot f(u)$ の閾値を厳密にすることで回避できるが、完全には回避できない。本手法では重み付けとマッチング手法の双方によってそれを回避している。

2.2 中心軸の木グラフ化

次に中心軸変換によって得られた中心軸画像を木グラフ表現に変換する。グラフのノードの集合を以下のように定義する。

$$node = \left\{ P \mid \sum_{a=1}^8 p_a = 1, 3, 4 \right\} \quad (2)$$

P : 注目する画素

p_a : P の8近傍に画素が存在すれば1

すなわち、線の端点、三叉路、十字路の点をノードとする。図5にその例を示した。またエッジはノード間を結ぶ残りの $\sum_{a=1}^8 p_a = 2$ である画素の集合である。このようにして見つめられたノードは隣接するノードとエッジの情報を保持しており、エッジは隣接する各ノードの情報を保持している。そしてノード、エッジの各点はその点の座標、内接円半径 (= 輪郭への最小ユークリッド距離) を記憶している。

この木グラフでは特定のノードを根とせず、非巡回のグラフとして扱う。

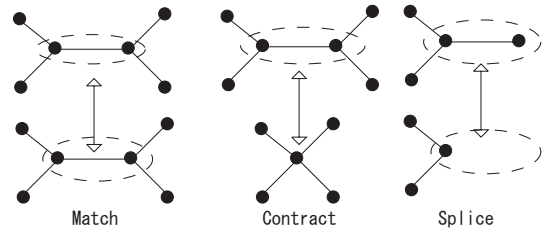


図6 編集操作

2.3 グラフマッチング

以上の過程で得られた中心軸からなる木グラフに対してグラフマッチングを行う。グラフマッチングとは異なるトポロジを有する二つのグラフの距離 (類似度) を求めるものである。一般にグラフマッチングの計算は複雑であり、最適なマッチングを求める問題はNP完全であることが知られている。そのため実際に計算する上では何らかの手法を適用して計算量を抑制しなくてはならない。

本論文では Klein の文字列操作を応用したアルゴリズム[6]を用いる。これはすなわち二つのグラフに対して以下に述べる変形操作を施して一致させるときの最小の編集操作量を類似度とするものであり、このアルゴリズムの計算量は動的計画法を用いることで $O(n^3 \log(n))$ となる。この手法は図形を滑らかに変形させる人間の自然な感覚に近く、各編集操作のコストは個別に独立して設定することができるため、計算の自由度が高いという利点がある。

定義する編集操作は以下の通りである。図6に各操作を図示する。

- 二つのエッジの一致 (match)
- 一方のノード間のエッジの縮退 (contract)
- 一方の葉のエッジの削除 (splice)

図形の類似度計算に適用する場合には、グラフ上の各ノードとエッジにその部分の図形のパーツを表現する重みを設定し、各々の編集操作にコストを定義する。

従来法としては編集操作としての各パーツの輪郭線と中心軸各々の弾性マッチングコストを割り当てる方法[7]や、輪郭線の長さをそのまま当てはめる方法[8]がある。またこのような編集操作の系列を用いず、隣接行列の固有値問題を解くことによって二つの図形の類似度を求める方法[9]があるが、自由度の面で劣る。

以降この重みと編集操作について新たに提案を行う。

3. 提案手法

シルエット図形の中心軸を元に、シルエット図形構造を解析する手法を提案する。本手法は中心軸の幾何的特徴を利用することで、中心軸から成るグラフのエッジに構造的かつ、人間の感覚として妥当な重みを与えることを可能にするものである。

3.1 半径関数による構造の解析

まず中心軸の幾何的特性について述べる。中心軸上の各点は閉曲線で囲まれた図形の内接円の中心であり、連続した曲線上

にある．よって中心軸の各点に内接円の半径を割り当てること
ができる．中心軸と軸上の各点の内接円の半径が分かれば元
の図形を完全に再構成することが可能である．そのため中心
軸は元の図形の情報を完全に保持しているコンパクトな表現
とすることができる．

ここでは内接円の半径 r を中心軸上の各点 s の関数 $r(s)$ と
見なし，その特徴に注目する． s は中心軸上の位置（あるノ
ードからの長さ）である．

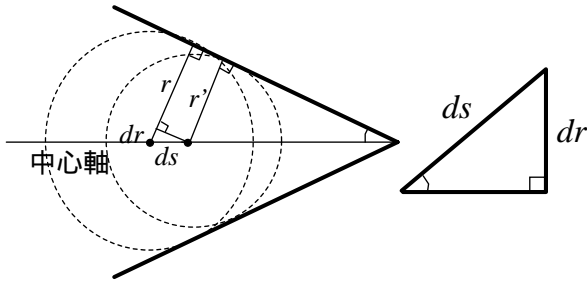


図 7 内接円半径の性質

ここで図 7 より内接円半径の性質を考えると，次の式

$$\left| \frac{dr(s)}{ds} \right| = \sin \theta \quad (3)$$

が成立し， $r(s)$ が中心軸上の内接円半径の関数であるとい
うことから $0 < \theta < 90^\circ$ という条件が成り立つ．よって (3) 式
の値の範囲は下記ようになる．

$$\left| \frac{dr(s)}{ds} \right| < 1 \quad (4)$$

(4) 式が 1 に近づくほど注目するエッジに対応する輪郭線が成
す角度が垂直に近づいていることを表す．もし $\left| \frac{dr(s)}{ds} \right| > 1$
ならばその点で対応する輪郭線の連続性が失われ，その点は内
接円の中心ではなくなってしまう．

図 8 に入力図形および得られた中心軸の例を示す．図 9 はそ
の中心軸上の $r(s)$ を示している．入力された図形の内接円半径
は中央の棒状の部分へ至るにつれて急激に減少している．つま
り半径の減少に比べて内接円の中心点の位置はあまり進まない．

このような急な半径の減少は図 10 のように大きな構造から
小さな構造への連結部分で生じる．この半径関数の傾きが 1 に
近い部分を抽出すれば，図形のパーツの繋がりだけでなく従
属関係も調べることが可能になる．よって図形がどのようなヒ
エラルキーを持っているかを判別することが可能であることが
予想される．

また (4) 式より，内接円半径の変化率は二つのパーツの大き
さの比で定まるスケール不変の特性をもつため，どの図形でも
同じ基準で扱うことができる．よって半径関数の変化率は比較
的安定な特徴とすることができる．デジタル画像に適用した
場合，完全とまではいかないまでも比較的良好な結果を得て
いる．

この特徴を用いて図形の構造を分析し，その特徴量とパーツ
の繋がりとのトポジを示すグラフを用いてグラフマッチングの
計算を行う．

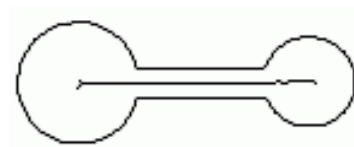


図 8 入力された図形

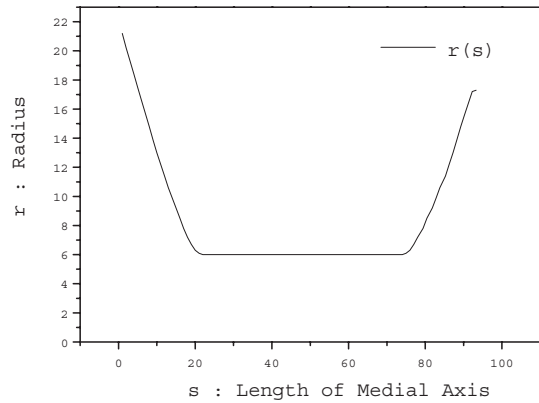


図 9 半径の変化

3.2 図形の構造化

本節では図形の構造化を行う方法について述べる．まず図形
の木グラフのエッジ，つまり中心軸上の各点は輪郭画素への最
短ユークリッド距離を保持しており，これを内接円の半径とし
ている．各エッジについて内接円半径の変化率の大きい点を抽
出する．変化率 $a (= \frac{dr}{ds})$ は隣接する 3 点についての最小二乗
法を各点に適用して計算する．

$$a(s) = \frac{\sum_{i=1}^3 s_i r_i - 3\bar{s}\bar{r}}{\sum_{i=1}^3 s_i^2 - 3\bar{s}^2} \quad (5)$$

s_i が中心軸上の各点， r_i がその中心軸の点に対応する内接円の
半径であり， \bar{s}, \bar{r} は 3 点の平均である．

そして次に $|a| > 0.5$ である中心軸の点にマークを付ける．
0.5 という数値は実際に実験をいくつかの図形で行った結果，良
好だった閾値である．

3.1 節での考察に基づき，各ノードから見たエッジの接続の
状態にラベル付けを行う．今回は以下の 3 つの状態を定義し，
ノードに Dependence として記憶することにする．

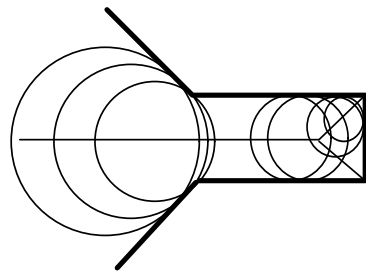


図 10 中心軸上での半径の変化

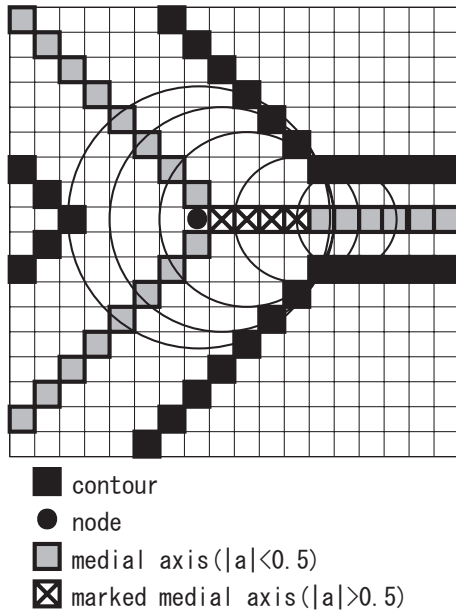


図 11 マーキングの例

- Even: エッジはノードに対して対等に接続
- Under: エッジはノードに対して従属している
- Over: ノードはエッジに対して従属している (被従属)

全ての中心軸の点についてマーク付けを行ったあと各ノードに注目し、それにつながるエッジ点の傾き a の正負を用いて従属か被従属を判断する。ノードから見て5つまでの範囲で $a < -0.5$ である点が3以上あるのであれば、従属 (Under) しているとし、 $a > 0.5$ の点が3つ以上存在するならば被従属 (Over) であるとする。それ以外の場合が対等 (Even) である。

例として図 11 を示す。実際のデジタル画像でのマーキングの様子を模式的に表している。マークされた点を x で示している。この場合、ノードから見て右側のエッジに注目すると、ノードから5つまでのエッジ上で $a < -0.5$ である点が4つあり、これは従属の条件を満たしている。よってノードから見て右のエッジはノードに従属している。

図 12 には実際の画像に適用した場合の結果を示す。図 12 の点線部分が図 11 の x 点に対応する $|a| > 0.5$ のマーキング点である。主要なパーツに対応する中心軸に従属する構造のエッジは正しくマーキングされており、それによって分離される様子が分かる。一部量子化誤差による孤立したマーキング点が存在するが、孤立点は無視し、集中しているマーキング点を選択するようにしているため正しい構造を取得できる。

このように中心軸上の内接円半径の変化が大きい部分を抽出して全てのノードにラベルを付けることで図形を階層的に表現したグラフが得られる。

3.3 エッジ重みの定義

前節の各ノードの *Dependence* フラグを用いて、実際にどのエッジにどの領域が属しているのかを考える。本論文では各エッジに図形の領域を対応させ、その面積を重みとする。

実際計算するときには、内接円がどの点で輪郭に接触してい

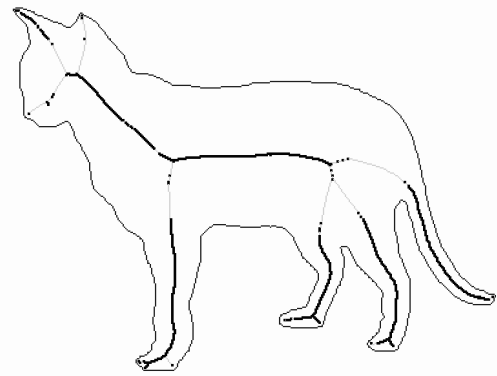


図 12 マーキングを行った中心軸グラフ

るかを厳密に求め、領域を分割する必要がある。接触しているか否かは幾何学的には簡明な性質であるが、デジタル画像上では量子化誤差のために厳密に決定することが難しい。よって各エッジ上の各点で内接円を再描画することで各エッジに対応する領域を再構成し、注目するエッジに対応する領域をパーツとして決定した。

その再構成の例を図 13 に示した。図 13(a) は輪郭と中心軸の様子を表している。中央の2番のエッジを中心軸上の内接円を再描画することで再構成したのが、図 13(b) の色分けしてある部分である。しかしこの面積をそのまま重みとすると、他のエッジの領域と重複する部分が生じてしまう。よってエッジの端点のノードにおいては交差する各々のエッジに領域を配分する必要がある。ここで3.2節で述べた従属関係を用いる。

基本的なポリシーとして従属関係により、親構造の領域は親構造に与え、子の構造には与えない。また対等な関係のエッジ同士には領域を対等に配分することとする。

具体的にはエッジの重みを *edge.weight* とすると、

$$\text{edge.weight} = (\text{再構成した領域}) - (\text{対象エッジに属さない領域})$$

となる。エッジに属さない領域は端点のノードの内接円の領域を分割することで求める。対象エッジについて *Dependence* = Even, Over だった場合は、その判断基準は以下の通りである。

$$(\text{エッジに属さない領域}) = (\text{端点ノードの内接円の領域}) \times \left(1 - \frac{1}{(\text{Dependence} = \text{Even or Over の数})} \right)$$

対象エッジについて *Dependence* = Under だった場合は (エッジに属さない領域) = (端点ノードの内接円の領域) となる。

図 13(c) は図 13(a) の情報を元に、上で定義した方法を用いて領域を割り当てた結果である。エッジの番号と領域の番号 (rNo) が対応関係を示している。

2番のエッジに注目し、割り当ての方法の詳細を示す。両端のノードから見て、接続しているエッジはそれぞれ3本であり、

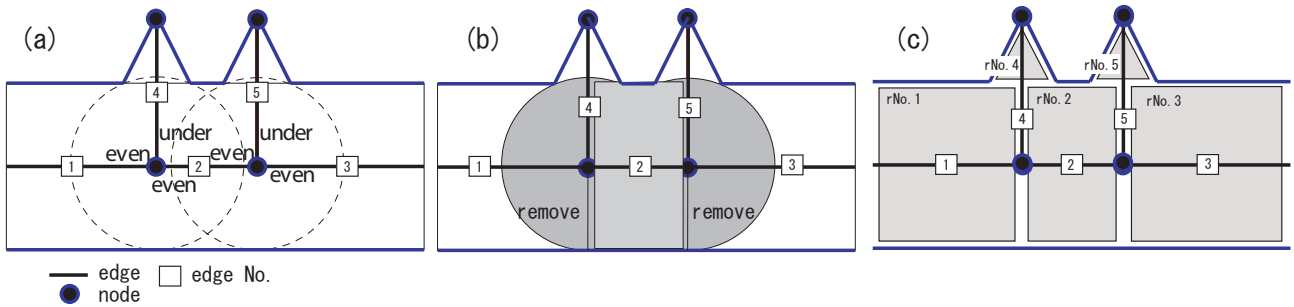


図 13 重み割り当ての例

かつ $Dependence = Even, Over$ である数はそれぞれ Even が 2 つなので、最終的に再構成した領域から (内接円の領域)/2 の領域がそれぞれ引かれることとなる。取り除かれた分の領域はそれぞれエッジ 1, 3 に与えられることになる。その結果エッジには図 13(c) の領域 2 (rNo.2) のような長方形の領域が割り与えられる。またその他のエッジについても見てみると、中央の軸には大きな重みが与えられ、上部の突起には小さな重みしか与えられていないことがわかる。

また実際に、図 12 について各エッジに領域の割り当てを行ったのが図 14 である。頭と胴体から耳、足、尾と自然な形で分離できていることが分かる。また胴体と後ろ足を連結する中間の重要度の小さいエッジには小さい領域が与えられていることが分かる。

これらの領域の面積を $edge.weight$ とするが、最終的に $\sum edge.weight = 1$ となるように正規化を行った。

この重みの特徴を考えると、まずこのように領域を階層的に分割することは人間の感覚的にも自然なものであると思われる。更に言うならば認知科学の一般化円錐の発想にも通ずるものがあると言える。もし Even で結ばれた主たる構造の軸を取り出せば、それを主軸と見なすことも可能だろう。その他の望ましい特徴として、正規化した面積はスケール不変であることが挙げられる。

3.4 グラフ編集操作の定義

以上のようにして求めた重みを用いてグラフマッチングの

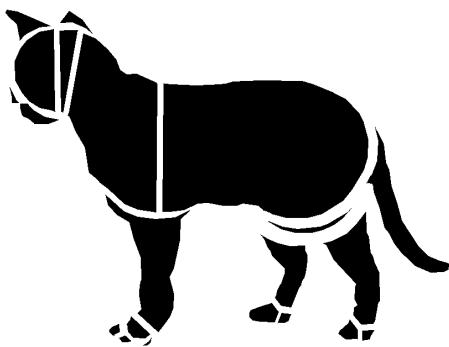


図 14 実画像での領域割り当ての例

計算を行う。グラフマッチングのコストは下記のような特性を持っていることが望まれる。

- (1) 図形の滑らかな変形に対して連続であること
- (2) 見た目上は大きな変化がないようなグラフ上の局所的なトポロジの変化に対して安定であること
- (3) 人間の感覚として自然な範囲のアフィン変換に対して頑健であること

上記の条件を満たすグラフマッチングの重みとして構造化された情報を用いた重みを利用する。すなわち、本論文では図 14 のように構造的に分解された要素の面積 (画素数) を正規化してエッジの重みとして与え、類似度計算に用い、その結果 $MatchingCost(MC)$ が小さいほど似ているということにした。

エッジ領域の面積という重みをグラフ編集操作の重みとしての面で考えてみると、図形が変形し、面積が滑らかに増減したとき、この重みも滑らかに変化するという特性を持っている。また輪郭のノイズはグラフのトポロジに大きな影響を及ぼすが、生じてしまった本来存在しないエッジについて、他のエッジに比べて重みを非常に小さく与えることができれば、グラフマッチングの編集過程で削除され、大勢に影響しない。回転などを伴う変換にもグラフ表現によってトポロジが保持されているため一定の耐性があるものと考えられる。

以上を考慮して各編集操作と $MatchingCost$ を以下のように定める。二つのグラフの対応するエッジを $edge(1), edge(2)$ とし、それに対応する重みを $edge(1).weight, edge(2).weight$ と表す。

- $cost(match\ edge(1)\ and\ edge(2)) = |edge(1).weight - edge(2).weight|$
- $cost(contract\ edge(n)) = edge(n).weight$
- $cost(splice\ edge(n)) = edge(n).weight$

$$MatchingCost(MC) = \min\left(\sum(cost(match) + cost(splice) + cost(contract))\right) \quad (6)$$

これは重みのみを考慮した最もシンプルと考えられる編集操作であり、この定義を用いて実験を行えば、純粋に本手法の性能を確認することができると思われる。

ここで特に問題となるのは $contract$ である。異なる内接円半径に対応している隣接する二つのノードを一致させる操作であるために、単にノードを繋ぐエッジを削除しただけでは不自然

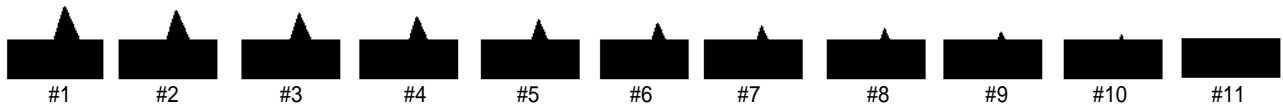


図 15 実験で用いた画像の系列 (図 16 に対応している)

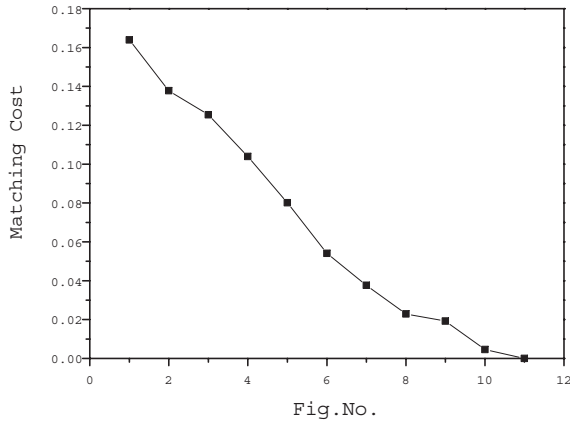


図 16 図形の滑らかな変化に対する類似度の変化

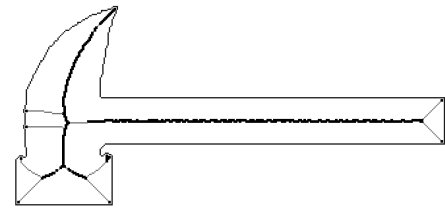


図 17 hammer1 (元画像)

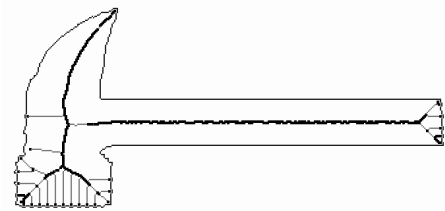


図 18 hammer2 (ノイズあり)

な変形を招き、2.3 節で述べた操作の滑らかな連続性に反することが起こりえる。しかしここでは提案手法の基本的な有効性を確認することが目的であるため、以上に述べたコストだけを用いた。contract の最適性の考察は今後の課題である。

4. 実験

本手法の有効性を確認するための実験を行った。まず評価法としての有効性を調べるため、その連続性、ノイズ、スケール不変性について実験を行い、最後に 16 枚の画像を用いて類似度を計算する実験を行った。画像は図 15 以外は全て Sebastianらのグループの用いたデータベース [7] の画像を一部加工して用いた。

4.1 評価量としての妥当性

まず、図 15 の画像を用い、構造の滑らかな変化に対してマッチングコストが連続的に変化するかどうかを調べるために 11 番を基準として 1 番から 11 番までの各図形に対してその類似度を計算した。結果を図 16 に示す。

結果より入力画像の単調な変化に対して、コストもまた単調に変化していることが分かる。従って提案手法によって構造化された領域の面積は、滑らかな変形に対応した特徴を有していることが分かる。

次にノイズの影響について調べた。図 17 と図 18 の二枚を比較した。ほぼ同型のハンマーであるが、図 18 には多くのノイズによる枝が存在している。この二つの図形のマッチングコストは $MC = 0.097$ である。これは後述する図 21 のハンマーの行と比較しても、同じハンマーのクラスに属するが形状のパターンが違う hammer3 と hammer4 とのマッチングコストよ

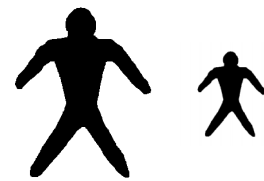


図 19 大きさの違う画像

りも小さい値となっており、ノイズの影響が最低限で押さえられていることが分かる。またスケールに対する頑健性についても実験を行った。図 19 の二つの図形のマッチングコストを計算したとき、 $MC = 0.087$ となった。これは十分小さい値であり、スケール変化に対しても頑健であると言える。またこの 2

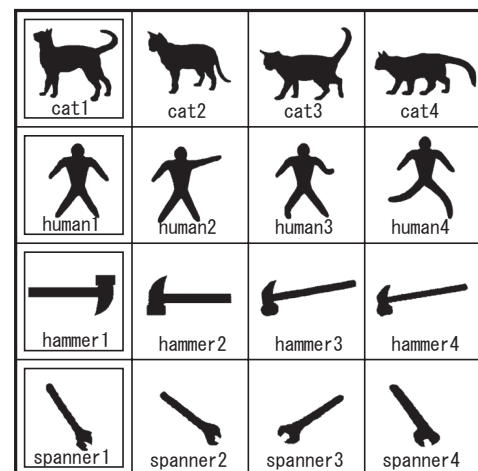


図 20 識別実験に用いた画像

































































															
cat	0.171	0.273	0.291	0.338	0.340	0.351	0.361	0.362	0.387	0.389	0.402	0.408	0.414	0.428	0.481
															
human	0.042	0.141	0.214	0.342	0.355	0.372	0.379	0.384	0.388	0.399	0.412	0.425	0.426	0.449	0.486
															
hammer	0.097	0.177	0.182	0.296	0.308	0.337	0.357	0.372	0.378	0.394	0.402	0.415	0.468	0.473	0.574
															
spanner	0.149	0.159	0.186	0.284	0.297	0.332	0.408	0.442	0.442	0.462	0.467	0.481	0.486	0.512	0.534

図 21 識別実験結果 (左端が比較元画像)

つは理論的には等価な図形であるはずだが、コストが 0 にならないのは、縮小操作で輪郭の形状が変化し、中心軸の構成が僅かに変わったためと考えられる。

4.2 識別能力について

最後に図 20 に示す 16 枚の画像について類似度、すなわちマッチングコストを計算した。この実験では 4 枚ごとに「猫」、「人」、「ハンマー」、「スパナ」という 4 つのクラスを想定しており、各カテゴリについて代表画像を一枚設定し、代表画像と代表画像を含む 16 枚の画像全体に対してマッチングコストを計算し、コストが小さい順に全ての画像を並べた (図 21)。代表画像自身とのマッチングコストは 0 である。図 20 において枠で囲まれているのが代表図形である。図 21 の左端が比較元の代表図形であり、右側の各図形の下の数値が左端の代表図形とのマッチングコストを表している。

結果を見ると各代表図形について各々が属するクラスの画像を上位に抽出することに成功している。特に比較的構造がシンプルなハンマーとスパナでは自クラスとそれ以外で明確なコストの差が確認でき、本手法は図形識別に有効であると言える。またハンマー、スパナの項に注目すると、代表図形が回転したような図形が存在しているが、それらについても正しくコストが求められている。特にハンマーの項では hammer1 と hammer2 が類似しているが、その他のハンマー、そして比較的似ていると思われるスパナと混在することなく識別できている。

しかし構造が複雑な猫では 3 位と 4 位の間の差があまり小さくなく、4 位ではハンマーが猫に比較的近いと言えるような結果になった。これは形状を全く考慮せず、正規化された面積だけで計算を行っているため、想定外の適合が起こっているためである。これは軸の長さを考慮するなどすれば、大きく改善するものと考えられる。また構造が複雑になると領域の従属関係の判定が不安定になりやすいという問題も関連していると思われる。

5. おわりに

本論文ではシルエット画像の類似性判定のために、中心軸を用いたグラフマッチング手法を提案した。中心軸変換によって

得られた中心軸と内接円半径の幾何的特性に注目し、シルエット画像を階層的に表現することを提案した。またその表現を用いて階層的な図形の領域分割を行って、グラフマッチングを適用するためのコストを割り当てる手法を提案し、その有効性を確認した。

今後の課題として、さらに多くの画像群についてその有効性を確かめることが挙げられる。

また理論的には正しくても、量子化誤差に起因する半径関数の不安定さのためにうまく計算できない場合がある。確実に図形の構造を表現するために、より安定した特徴を抽出することが欠かせない。そのために半径関数の部分ごとの適応的な直線近似なども考慮する必要がある。

またシンプルな領域だけの定義でも一定の成果を得ることができたが、更に精度を向上させるために、図形の曲がりや長さも考慮に入れる手法を検討することが今後の課題である。

文 献

- [1] E. Rosch, C. B. Mervis, W. D. Gray, D. M. Johnson and P. Boyes-Braem, "Basic objects in natural categories," *Cognitive Psychology*, vol. 8, no 3, pp. 382-439, July, 1976.
- [2] R. Basri, L. Costa, D. Geiger, and D. Jacobs, "Determining the Similarity of Deformable Shapes," *Vision Research*, vol. 38, no. 1, pp. 2365-2385, Jan. 1998.
- [3] H. Blum, "Biological shape and visual science(part I)," *Journal of Theoretical Biology*, vol. 38 pp. 205-287, 1973.
- [4] K. Siddiqi, S. Bouix, A. Tannenbaum, and S. W. Zucker, "Hamilton-Jacobi Skeleton," *International Journal of Computer Vision* vol. 48 no. 3 pp. 215-231, 2002.
- [5] 坂井秀行, 杉原厚吉, "図形の中心軸の安定した生成法," *信学論 (D-II)*, vol. J85-D-II, no. 11, pp. 1637-1644, Nov. 2002.
- [6] P. N. Klein, "Computing the Edit-Distance Between Unrooted Ordered Trees," *Proc. 6th European Symposium on Algorithms*, pp. 91-102, 1998.
- [7] T. B. Sebastian, P. N. Klein, and B. B. Kimia, "Recognition of Shapes by Editing Their Shock Graphs," *IEEE Trans. PAMI*, vol. 26, no. 5, pp. 550-571, May 2004.
- [8] A. Torsello, and E. R. Hancock, "A skeletal measure of 2D shape similarity," *Computer Vision and Image Understanding* vol. 95, pp. 1-29, 2004.
- [9] K. Siddiqi, A. Shokoufandeh, S. J. Dickenson, S. W. Zucker, "Shock Graphs and Shape Matching," *International Journal of Computer Vision* vol. 35 no. 1 pp. 13-32, 1999.