コンピュータビジョンと
イメージメディア 104－1
（1997．3．27）

スケルトンの抽出と階層化による輪郭線図形の多重解像度解析

中尾 学　本谷 秀堅　出口 光一郎
東京大学工学部

図形の形状を解析するためにスケルトンに着目した。スケルトンは図形をその骨格を表す線によって記述するものである。数学的な定義に基づいてスケルトンを抽出する手法が従来提案されてきたが、これらは細部の構造に敏感すぎるという問題点があった。そこで、スケールを変化させて輪郭線をぼかして、スケルトンを抽出する。このスケルトンの組を多重解像度スケルトンと呼ぶ。この多重解像度スケルトンをもとに、スケルトンの各部分の重要度を求め、図形を特徴をうまく捉えるスケルトンを抽出する。さらにその構造を階層化させることによって図形の持つ階層的な構造を捉えることができる。本稿では多重解像度スケルトンからスケルトンの階層化を行う手法を提案する。

A Multi-resolution Skelton Analysis for Hierarchical Description of Planar Contour Shape

Manabu Nakao, Hidekata Hontani
and Koichiro Deguchi

Faculty of Engineering, University of Tokyo,
Bunkyo-ku, Tokyo, Japan

We propose a new method of constructing a hierarchical structure of multi-resolution skeltons. Skelton is well defined mathematically. But such a mathematically defined skelton is usually too sensitive to details of original contour. To overcome this problem, we blur the planar shape contour and extract skeltons at each resolution. This set of skeltons are called multi-resolution skeltons. From these multi-resolution skeltons, we evaluate the importance of original shape's skelton and select their components which characterize the original shape. By hierarchical description for these skeltons, we can construct the hierarchical structure of shape.

1 はじめに

本研究では図形形状を解析するために、スケルトンに着目した。スケルトンは2次元の図形を1次元の曲線で記述することによって図形を把握しやすくしている。図形形状の解析を行うためには骨格をうまく表すようなスケルトンを図形から直接求めたい。そこで、数学的な定義に基づいて図形からスケルトンを求めるという手法は従来から提案されていて、幾つかの等価なスケルトンの定義がある。例えば最大円の中心の集合としてスケルトンを定義する方法がある。これは図形の輪郭線に内接する最大の円を順次移動していったときの円の中心の軌跡としてスケルトンを定義している。しかし、このような数学的な定義に基づくスケルトンは細部の構造に敏感すぎるので、抽出されたスケルトンから図形の特徴をうまく捉えた骨格を得るのは困難である。

一般に図形は局所的な構造から大局的な構造へと階層をなしている。この階層的な構造はdeep structureと呼ばれる[1]。そこで、スケルトンのどの部分が重要な構造を表すかを区別することにより、図形の特徴をうまく捉えたスケルトンを見分けることができれば、スケルトンの構造を階層化することによって、それらの図形のdeep structureを捉えることができる。

このため、多重解像度でスケルトンを生成し、スケルトンの構造を階層化させるという考えがPizerら[2]によって提案された。各解像度のでスケルトンの集合が多重解像度スケルトンである(図1)。

解像度を次第に粗くしていくとき、重要な構造を表すスケルトンは粗い解像度まで残る。逆にノイズなどの細部の構造を表すスケルトンは細かな解像度でしか存在しない。つまり、元図形のスケルトンの各部分がどれだけ解像度を粗くしても生き残るかを求めることによって、そのスケルトンの各部分の重要度を求めることができる。そしてこの重要度からスケルトンの構造を階層化することができる。

図形の解像度を変える方法としてPizerらは、2つの手法について考察をおこなっている。1つ目の手法は図形内部を値1、図形外部を値0にした特性関数をつくり、その解像度に対応するガウシアン関数で畳み込み積分する。そしてその特性関数のある等高線の内部をぼかされた図形とする。もう1つの手法は輪郭線の座標系列をガウシアン関数でぼかすことによって図形をぼかすものである。彼らは前者の手法の方が形状の大局的な関係を捉えられると主張してスケルトンの構造を階層化させたが、各解像度間のスケルトンの構造の対応付けが困難であるため自動的な階層化まではおこなわなかった。自動的な対応付けをおこなうには後者の方が容易であるとも指摘し、前者または後者の手法のもとで、スケルトンの変化を自動的に捉えていく研究をおこなう必要があると述べている。

そこで本研究では後者の手法で図形をぼかし、多重解像度スケルトンを生成し、スケルトンの構造の自動的な階層化をおこなった。そして、この多重解像度スケルトンの応用として図形の分割についても試みたので報告する。

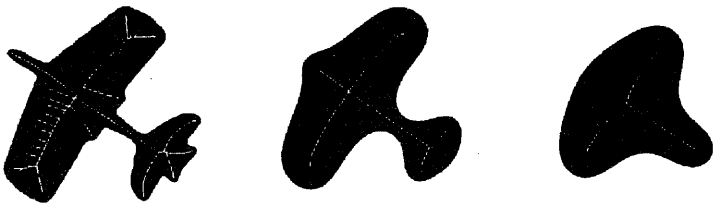
本稿の構成は第2節でスケルトンの生成法、第3節で多重解像度スケルトンの生成法について示す。得られた多重解像度スケルトンの各解像度間のスケルトンの対応をとる手法を第4節で述べ、それをもとにスケルトンの構造を階層化させる手法を第5節で示す。図形の階層的な分割ついては第6節で述べ、最後にまとめを述べる。

2 スケルトンの生成法

スケルトンの生成には以下に示すKimmelらの手法[3]を用いた。ここでは、スケルトンを構成する各点をスケルトン点と定義する。また、スケルトン点を中心とする最大円が輪郭線に接する点をそのスケルトン点に対応するスケルトン生成点と定義する。

1. 輪郭線を曲率が正の極大となる点で部分輪郭線$\boldsymbol{C}_0,\cdots,\boldsymbol{C}_{N-1}$へと分割する。$N$は部分輪郭線の数である。
2. 図形内部の各点pにおいて、各部分輪郭線$\boldsymbol{C}_i$への距離関数$D_i(p)$をfast marching methodという高速計算アルゴリズム[4]を用いて計算する。ただし、距離関数$D_i(p)$は点pから部分輪郭線$\boldsymbol{C}_i$への最短距離、すなわち$d(p,q)$を点pと点qの間の距離として、

$$D_i(p) = \min_{v \in C_i} d(p,v) \qquad (1)$$

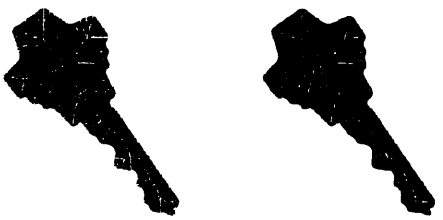


細かい解像度 ⟷ 粗い解像度

図 1: 多重解像度スケルトン

3. 全ての 2 つの部分輪郭線の組み合わせにおいて、2 つの距離関数の差のゼロ等高線上にあり、なおかつその点でのそれら 2 つの距離関数の値が他の距離関数よりも小さい図形内部の点の集合を求める。この集合はスケルトン点をすべて含むので、これを仮のスケルトンと呼ぶことにする。
4. 刈り取り処理
   輪郭線上で曲率が極大となる点から始まる仮のスケルトンの枝で、曲率が極大となる点からの距離が曲率半径以下に位置する全ての点を削除する。これらの点は輪郭線で最も近い点が 1 点しかなく、最大円の中心とはならない。ただしスケルトンの枝とは、枝別れを持たないようにスケルトンを分割したときの、各分割されたスケルトンの部分である。

この手法によるスケルトンの生成結果を図 2 に示す。



(a) 刈り取り処理前　　(b) スケルトン

図 2: スケルトンの生成結果

3 多重解像度スケルトンの生成

図形をぼかすことによって各解像度に対応する図形を生成する。この各解像度に対応する図形から求めたスケルトンの集合が多重解像度スケルトンである。本稿では図形をぼかす手法として、Mokhtarian と Mackworth[5][6] の輪郭線のぼけ変換を用いた。

まず、元図形の輪郭線 C_0 を、弧長を全周の長さで割ったパラメータ w で表す。

$$C_0(w) = \{(x(w), y(w)) | w \in [0,1]\} \quad (2)$$

この元図形をパラメータ t でぼけ変換した輪郭線を以下のように定義する。

$$C(u,t) = \{(X(u,t), Y(u,t)) | u \in [0,1]\} \quad (3)$$

ただし

$$X(u,t) = x(u) * g(u,t) \quad (4)$$

$$Y(u,t) = y(u) * g(u,t) \quad (5)$$

$$g(u,t) = \frac{1}{\sqrt{4\pi t}} e^{\frac{-u^2}{4t}} \quad (6)$$

である。これは元図形の輪郭線の各 x, y 座標を幅 $\sqrt{2t}$ のガウシアン関数で畳み込み積分している 。この t をスケールパラメータと呼ぶことにする。スケールパラメータが大きいと大きくぼかすことになり粗い解像度に対応する。

このぼけ変換によって各解像度に対応する図形を求めた結果を図 3 上に示す。

このぼけ変換では、解像度が粗くなる過程において、曲率の正の極大点は新たに生成されない。その様子を示したのが図 3 下である。これはスケールス

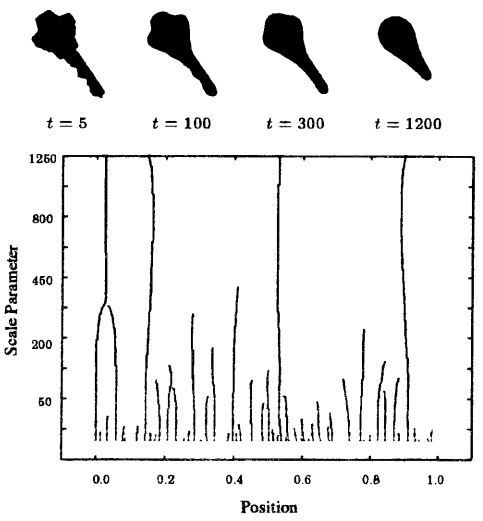


図 3: ぼけ変換の結果 (上はぼけ変換された図形、下は曲率が正の極大となる点をスケールスペースにプロットしたもの)

ペースと呼ばれる (位置)-(スケールパラメータ) 空間に、各解像度での曲率が極大となる点をプロットしたものである。

このように各解像度に対応する図形は、少しぼかしてはパラメータを弧長で表現し直すことを繰り返しすことによって生成される。そこで、ぼけ変換を用いて図形を少しぼかしてはスケルトンを求めるという処理を繰り返すことによって、多重解像度スケルトンを生成する。

本稿で用いたスケルトン生成法は、曲率が正の極大となる点で輪郭線を分割することによってスケルトンを生成している。スケルトンの枝のうち端にあるものはこの分割点を境とした 2 つの部分輪郭線から生成される。この分割点は解像度を粗くしていくと消滅するのみであるため、これらのスケルトンの枝は徐々に消滅していく。そして、スケルトンの枝が消滅した後では曲率が正の極大であってもその点で輪郭線を分割する必要はない。

そこで、各解像度で単に曲率が正の極大となる点で分割するだけでなく、分割する点を追跡をおこなう。そして、分割しなくなったらそれより粗い解像度ではその点では輪郭線の分割を行わないことにする。この分割点の追跡は、後節で述べるスケルトンの枝の対応付けに用いる。

多重解像度スケルトンの生成アルゴリズムの概略を以下に示す。

1. 初期処理
   (a) 離散化の影響のなくなる $t = t_0$ まで輪郭線をぼかす。
   (b) 解像度 $t = t_0$ での輪郭線からスケルトンを生成する。そして、曲率が正の極大となる点のうち、スケルトンの枝を生成するものだけを求め、分割点の集合 P をつくる。
2. 多重解像度スケルトンの生成
   (a) Δt だけ更に輪郭線をぼかす。新しくぼかされた輪郭線で曲率が正の極大となる点を抽出する。
   (b) 分割点の集合 P に属する各点が、(a) で得られる曲率が正の極大となる点のどれに対応するかを求める。これらの点で新しくぼかされた輪郭線を分割し、スケルトンを生成する。
   (c) スケルトンの枝を生成しなくなった分割点を P から削除する。
   (d) (a) に戻る

生成された多重解像度スケルトンを図 4 に、そして各解像度での分割点をプロットしたスケールスペースを図 5 に示す。

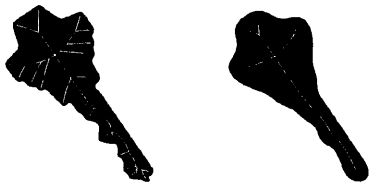
4 各解像度間のスケルトンの枝の対応付け

各解像度間のスケルトンの枝の対応付けをおこなう。

解像度を粗くすると、分割点は消滅する、あるいは隣の分割点と融合していく。

1. 消滅する場合 (図 6)
   この分割点を境とする部分輪郭線は融合
2. 隣の分割点と融合する場合 (図 7)
   この 2 つの分割点によって挟まれた部分輪郭線は消滅

この判別は次のようにしておこなう。ある分割点を P_1、その隣の分割点を P_2 とする。分割点 P_1 から出



(a) $t = 5$ (b) $t = 100$

(c) $t = 300$ (d) $t = 1200$

図 4: 多重解像度スケルトン

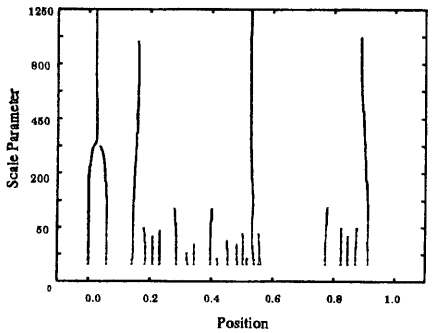


図 5: 各解像度での分割点をプロットしたスケールスペース

ている枝が消滅したときに、

- 分割点 P_1 から出ている枝と分割点 P_2 から出ている枝が接続
- 分割点 P_2 から出ている枝の長さがある一定値以下

ならば、分割点 P_1, P_2 が融合するとする。そうでなければ分割点 P_1 は消滅するとする。

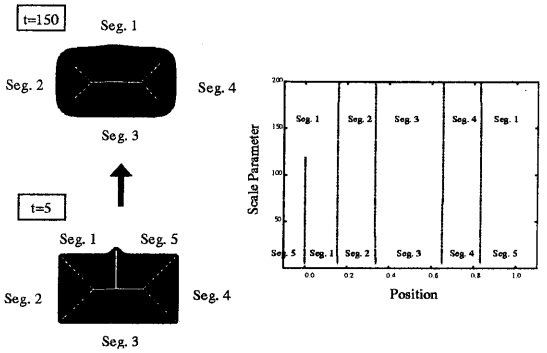


図 6: 分割点が消滅する場合

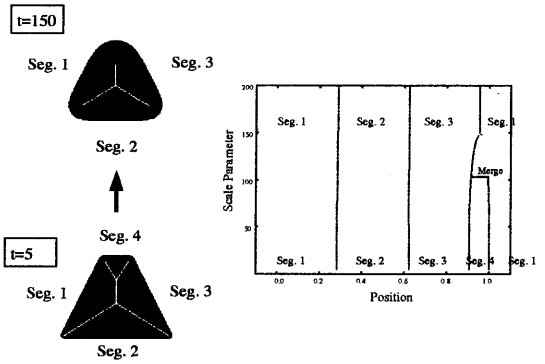


図 7: 分割点が隣の分割点と融合する場合

スケルトンの各枝は 2 つの部分輪郭線から生成される。そこで、各解像度間のスケルトンの枝の対応付けを、部分輪郭線の対応付けを用いておこなう。粗い解像度でスケルトンの枝を生成する 2 つの部分輪郭線に対応する、細かな解像度での部分輪郭線の集合をそれぞれもとめる。そして、その 2 つの集合の要素同士で生成される、細かな解像度でのスケルトンの枝の集合が粗い解像度でのスケルトンの枝に対応するスケルトンの部分である。

図 6, 図 7 の図形に対し、スケールスペースから得られる部分輪郭線の対応、およびその対応から得られるスケルトンの枝の対応を表 1 と表 2 に示す。ただし、ここではスケルトンの枝を部分輪郭線の組で表すことにする。

このように、解像度が粗くなると部分輪郭線は融

表 1: 図 6 の $t=5$ と $t=150$ 間の部分輪郭線の対応 (左) とスケルトンの枝の対応 (右)

$t=150$	$t=5$
Seg.1	Seg.1
Seg.2	Seg.2
Seg.3	Seg.3
Seg.4	Seg.4
Seg.1	Seg.5

$t=150$	$t=5$
Seg.1–Seg.2	Seg.1–Seg.2
Seg.1–Seg.3	Seg.1–Seg.3
	Seg.5–Seg.3
Seg.2–Seg.3	Seg.2–Seg.3
Seg.3–Seg.4	Seg.3–Seg.4
Seg.4–Seg.1	Seg.4–Seg.5
なし	Seg.1–Seg.5

表 2: 図 7 の $t=5$ と $t=150$ 間の部分輪郭線の対応 (左) とスケルトンの枝の対応 (右)

$t=150$	$t=5$
Seg.1	Seg.1
Seg.2	Seg.2
Seg.3	Seg.3
なし	Seg.4

$t=150$	$t=5$
Seg.1–Seg.2	Seg.1–Seg.2
Seg.2–Seg.3	Seg.2–Seg.3
Seg.3–Seg.1	Seg.3–Seg.1
なし	Seg.1–Seg.4
	Seg.3–Seg.4

合あるいは消滅する。スケールスペースを用いて、2 つの解像度の間での部分輪郭線の対応をとれば各スケルトンの枝の対応をとることができる。

5 スケルトンの構造の階層化

元図形をぼかしていき解像度を粗くしていくと、スケルトンの枝は滑らかに変化していき、ある程度までぼかすと枝は消滅する。

そこで、この消滅する枝に対応する元図形のスケルトンの枝の集合を一つの枝集合として扱う。そして消滅するときのスケールパラメータの値をその枝集合の構造の大きさを表す尺度として用いることにする。そして、最後に一つの枝となるまで解像度を粗くしていく。

この処理により、最終的には元図形のスケルトンをスケールパラメータ付きの枝集合に分解できる。この各集合は枝分かれがない。

ここで、各枝集合間で親子関係を構築していく。接続している枝集合の間でスケールパラメータが大きい方を親、小さい方を子とする。一般には子の枝集合は親の枝集合へと消滅していく。

このように各枝集合間で親子関係を構築していくことにより、木構造の関係を作り上げることができる。木構造の頂点にはもっとも粗い解像度まで生き残る枝に対応する元図形の枝集合が置かれる。そしてその子にはその枝集合に接続しているすべての枝集合がおかれる。さらにその子には、というように階層的な構造を作り上げることができる。

解像度を粗くすると枝は徐々に消滅していく。この消滅した枝に対応する元図形のスケルトンの集合で 1 つの枝集合を作る (図 8)。この消えるときのスケールパラメータはその枝集合の構造の大きさを表す。

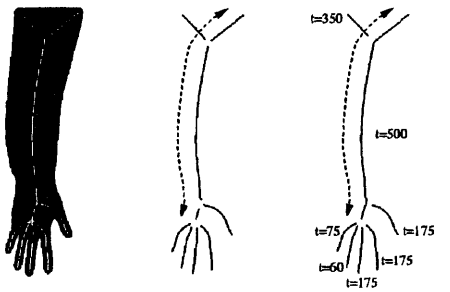


図 8: スケルトン (左), 各枝 (中), 各枝集合 (右)。矢印で示す枝の集合が一つの枝集合となる。

そして、接続している枝集合の間で親子関係を作っていく。スケールパラメータが大きな方の枝集合を親、小さな方の枝集合を子とする。この親子関係から木構造を作り上げることができる。

階層化を行った結果を図 9 に示す。

6 図形の階層的な分割

多重解像度スケルトンを用いた図形の階層的な分割について述べる。図形の分割については、突出部とくびれに基づいて図形をパーツへ分割したする手法 [7] がある。これをスケルトンに着目しておこなった。

- 突出部に基づく分割

  突出部では連絡橋・半連絡橋というものが存在

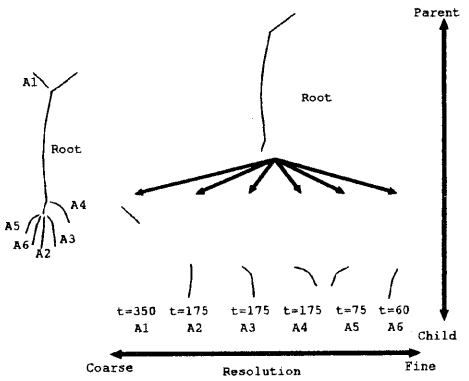


図 9: スケルトンの階層化の結果

する。連絡橋とは対応するスケルトン生成点が2点とも同じであるスケルトン点の集合のことである (図 10 左)。半連絡橋とは対応するスケルトン生成点のうち 1 点が同じであるスケルトン点の集合のことである (図 10 右)。そこで、図 10 のように連絡橋・半連絡橋の端点の輪郭線への距離関数が小さな方をもとに図形を分割する。図形の分割はこの点のスケルトン生成点を結ぶ直線でおこなう。

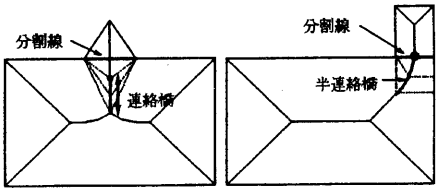


図 10: 突出部に基づく分割

- くびれに基づく分割
  くびれでは、スケルトン上で輪郭線への距離関数が極小となる点が存在する。そこで、図 11 のようこの点をもとに図形を分割する。図形の分割はこの点のスケルトン生成点を結ぶ直線でおこなう。

図形を分割した結果を図 12 に示す。

この分割では各パーツの構造は階層化されていない。そこで多重解像度スケルトンと組み合わせることにより、図形の階層的な分割を行う。

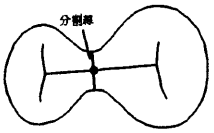


図 11: くびれに基づく分割

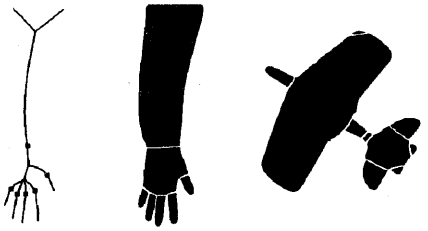


図 12: 図形の分割結果。左の図は分割の基となるスケルトン点をスケルトン上にプロットしたもの

まず、各解像度で図形を分割する (図 13)。元図形の分割線は解像度が粗くなるに従って徐々に消失して行く。粗い解像度まで残っている分割線は大きなパーツへの分解を示し、分割線が消える時の解像度によってその分割の強さを表すことができる。そこ

(a) $t = 5$ (b) $t = 60$ (c) $t = 200$

図 13: 各解像度での分割

で、分割線が出現する過程を最も粗い解像度から細かな解像度へと辿ることにより図形を強い分割線から分割していく。そして、多重解像度スケルトンによって階層化された枝構造を利用して、同じ階層に置くパーツの決定を行う。その概略を以下に示す。

1. 解像度を徐々に粗くしていき、各解像度で図形の分割をおこなう。そして、元図形の各分割線の分割の強さを求める。
2. 強い分割線から弱い分割線へと徐々にみていく。階層化されたスケルトンでは、その分割線に対応するスケルトン点を持つ枝が、
   (a) そのパーツで最も上位の階層の枝の場合
   その分割線でそのパーツを分解する。親はそのパーツ、子は分割した後の子パーツ達という親子関係をつくる。
   (b) そのパーツで最も上位の階層にない枝の場合
   そのパーツで次に強い分割線に対応するスケルトン点がそのパーツで最も上位の階層の枝にある、またはそのような分割線が存在しないときに、まとめて分割する。親はそのパーツ、子は分割した後の子パーツ達という親子関係をつくる。そうでなければ、取り合えずそのままおいておく。

このようにすれば図 14 のような階層的な分割を行える。

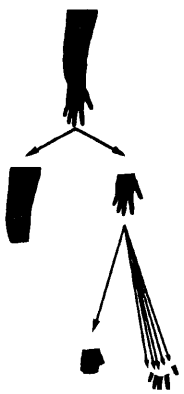


図 14: 図形の階層的な分割

7 まとめ

本研究では多重解像度スケルトンを用いたスケルトンの構造の自動的な階層化をおこなった。この階層化では自然な階層構造を得ることができた。そして、それを用いた図形の階層的な分割の手法について考案した。これらのような階層化により、図形の持つ階層的な構造を捉えることができると考える。

参考文献

[1] J. J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.

[2] S. Pizer, W. R. Oliver, and S. H. Bloomberg. Hierarchical shape description via the multiresolution symmetric axis transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(4):505–511, 1987.

[3] R. Kimmel, D. Shaked, and N. Kiryati. Skeltonization via distance maps and level sets. *Compuiter Vision and Image Understanding*, 62(3):382–391, 1995.

[4] R. Kimmel and J. A. Sethian. Fast marching method for computation of distance maps. *LBNL report, LBL-38451, UC Berkeley*, 1996.

[5] F. Mokhtarian and A. Mackworth. Scale-based description and recognition of planner curves and two-dimensional shapes. *IEEE Transactions on Pattern Analysis nd Machine Intelligence*, 8(1):34–43, 1986.

[6] F. Mokhtarian and A. Mackworth. A theory of multiscale, curvature-based shape representation for planner curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(8):789–805, 1992.

[7] K. Siddiqi and B. B. Kimia. Parts of visual form: Computational aspects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(3):239–251, 1995.