

블록 혈을 이용한 필기 한글 패턴의 모양 분해

박정선* 오일석
전북대학교 컴퓨터과학과

A Shape Decomposition of Handwritten Hangeul Patterns Using Convex Hull

Jeong-Sun Park and Il-Seok Oh
Department of Computer Science, Chonbuk National University, Korea

요약

필기 한글 문자 인식을 위해서는 패턴을 구성하는 획 성분을 분석하는 작업이 필수적이다. 획 성분 추출을 위해 사용한 세션화 방법은 입력 영상을 왜곡하는 단점을 가지고 있다. 이를 극복하기 위하여 본 논문은 입력 영상을 왜곡하지 않고 의미 있는 부품 단위로 분할하는 방법을 제안한다. 의미 있는 부품이란 유사 블록하게 분할된 영역을 의미한다. 분할 방법은 먼저 입력 영상에 블록 혈 연산을 적용하여 오목 영역을 생성한다. 이 오목 영역에서 분할 기준(anchor point)점을 탐지하고 획의 반대편 외곽선 상에서 분할 끝점(terminal point)점을 찾아 분할 경로를 구성하여 획을 분할한다. 모든 부품이 유사 블록 조건을 만족할 때까지 위 과정을 반복 수행한다. 제안한 방법은 두 개의 파라미터만을 가지며 간단한 프로시저로 구성되어 있다. 또한 필기 한글 패턴뿐 아니라 여러 언어에 적용 가능하다는 장점을 갖는다.

1. 서론

필기 한글 문자 인식을 위해서는 패턴을 구성하는 획 성분을 분석하여 특징을 추출하는 작업이 필수적이다. 이런 특징 추출을 위하여 기존 인식 방법들은 세션화와 직선 근사에 기반한 방법을 사용하였다 [김수형98, 이성환93]. 세션화 방법은 입력 영상의 화소를 제거하여 한 화소 두께로 이루어진 골격선을 추출하는 것이다. 하지만, 이 방법은 화소 제거 과정에서 꼭 필요한 정보를 제거하거나 불필요한 정보를 추가시키는 경우가 종종 발생한다. 또한 입력 영상을 새로운 형태인 골격선으로 변환시키기 때문에 입력 영상이 가지고 있는 고유 정보를 모두 사용할 수 없는 단점이 있다.

특징 추출 방법 중 다른 하나는 모양 분해 알고리즘이다. 모양 분해 알고리즘은 문자가 아닌 이미지 기반에서 블록하게 튀어나는 부분을 인간의 시각에 가깝도록 분할하는 것이다 [Latecki99, Rosin2000]. 모양 분해 알고리즘을 문자 패턴에 사용하면 왜곡되지 않는 획을 입력 영상에서 추출할 수 있다. 분해된 결과에서 세션화 방법처럼 골격선을 추출할 수 있으며 그 자체를 인식기의 특징으로 사용할 수도 있다. 또한 입력 영상이 그대로 존재하기 때문에 인식이 가 입력 특징의 재요청을 했을 때 간단한 파라미터 조정으로 입력 영상의 재분할이 용이하다. 모양 분해 알고리즘을 사용하면 입력 영상에 대한 모든 정보를 사용할 수 있으므로 근본적인 해결 방법이라 볼 수 있다.

모양 분해 알고리즘을 문자에 적용한 예를 한문 [Lin95, Jiang93, Tseng92]과 한글 [박정선99]에서 찾아 볼 수 있다. [Lin95]는 한문을 구성하는 4가지 특성(' ', '!', '!', ' ')을 분석하고 각도를 이용한 기하학 조건을 검사하여 획을 추출한다. [Tseng92]는 [Lin95]와 마찬가지로 4가지 특성의 획을 찾고, 찾은 획의 상호 연결 조건을 검사하여 골격선까지 추출한다. [Jiang93]은 획을 추출하기 위해 블록 혈 알고리즘을 적용하여 얻어진 블록 혈 영상에 입력 영상을 제거하고 생성된 영역들 중 하나의 영역으로 묶을 수 있는 두 영역을 선택하고 두 영역을 획으로 추출한다. [박정선99]는 한글에 모양 분해 알고리즘을 도입한 최초의 논문이다. [박정선99]는 한글을 이루고 있는 패턴을 세 가지 특징(B-점점, T-점점, 결합 부품)으로 분류하고, 이 지점을 분할하기 위해 외곽선 상의 정보를 이용한다. 자세한 내용은 2.1절과 [박정선99]를 참조하기 바란다.

[Lin95, Tseng92]는 획이 이루는 각도로 네 가지 특성을 분할하기 때문에 글꼴이 심한 필기 문자에 적용하기에는 한계점을 가지고 있다. [박정선99]는 필기 문자에 적용 가능한 방법이지만 알고리즘에 몇 가지 문제점을 가지고 있다. 첫째 지역적인 정보를 사용하는 우세점 탐지 알고리즘[Teh89]을 사용한다는 것과 둘째 두 단계로 나누어 수행한다는 것이다. 우세점 탐지 알고리즘은 인간의 시각에 맞는 특징점을 탐지하지 못한다는 단점이 있으며 두 단계로 나누어 수행하는 것은 두 번째 단계 입력이 첫 번째 단계의 출력에 종속적이기 때문에 첫 번째 단계에서 오류를 범한 데이터에 대해 재 오류를 범할 여지가 남아 있는 단점이 있다.

본 논문은 [박정선99]의 문제점을 파악하고 이를 해결하기 위한 방법을 제안한다. 제안한 분할 방법은 블록 혈 알고리즘을 적용하여 생성된 오목 영역에서 분할 시작점을 탐지한다. 이 분할 시작점 기준(anchor point)점이라 정의한다. 그리고 이 기준점의 반대편 외곽선에 기하학 조건을 검사하여 분할 끝점(terminal point)을 탐

지한다. 분할은 기준점-끝점의 쌍으로 이루어진다. 분할된 영상은 다시 알고리즘의 입력이 되어 위 과정을 모든 부품이 유사 블록할 때까지 반복 수행한다. PE92데이터베이스에 대해 실험한 결과 대부분의 데이터에 만족할만한 성과를 얻었다. 2장은 [박정선99]와 제안한 방법과의 비교 설명, 3장은 실험 결과, 4장은 결론 및 향후 연구 과제를 기술한다.

2. 한글 패턴에 적용한 모양 분해 알고리즘

제안한 방법의 입력은 이진 영상이며 최종 결과물은 유사 블록 부품들의 집합으로 분해된 영상이다. 모양 분해 알고리즘을 처음으로 적용한 연구는 [박정선99]이다. 그 연구를 간단히 요약하면 2.1절과 같고, 2.2절은 이 논문이 제안한 알고리즘을 기술한다.

2.1 기존 알고리즘 [박정선99]

[박정선99]는 한글을 이루고 있는 패턴을 세 가지(B-점점(그림 1(a)), T-점점(그림 1(b)), 결합 부품(그림 1(c)))로 분류한다. 오목한 부분에 있는 특징점을 오목 특징점(그림 1의 '○'로 표시)이라 하고, 블록한 부분에 있는 특징점을 블록 특징점(그림 1의 '□'로 표시)이라 정의한다. 세 가지 패턴인, B-점점은 오목-블록 특징점의 쌍으로, T-점점은 오목-오목 특징점의 쌍으로, 결합부품은 세 오목 특징점의 쌍으로 분할한다.



(a) B-점점 (b) T-점점 (c) 결합 부품

(○: 오목 특징점 □: 블록 특징점)

그림 1. 한글을 이루고 있는 세 가지 패턴

[박정선99]는 오목/블록 특징점을 찾아 유사 블록으로 분할하는 방법을 두 단계에 걸쳐 수행한다. 첫 번째 단계는 우세점 탐지 알고리즘[Teh89]을 적용하여 오목/블록 특징점을 탐지하고 이를 기하학 조건을 검사하여 T-점점과 B-점점을 탐지하고 분할한다. 우세점 탐지 알고리즘[Teh89]은 인간의 시각 관점에서 볼 때 완전한 오목/블록 특징점 탐지가 불가능하다. 이를 보완하기 위하여 두 번째 단계에서는 블록 혈 알고리즘[Zimmer97]을 적용하여 오목 영역을 생성하고, 이 영역 안에서 오목 특징점을 탐지하고 특징점 개수에 따라 T-점점, B-점점, 결합 부품으로 분할한다. 자세한 내용은 [박정선99]를 참조하기 바란다.

[박정선99]의 방법은 몇 가지 문제점을 가지고 있다. 첫 번째, 첫 번째 단계에서 오목/블록 특징점을 탐지하기 위하여 우세점 탐지 알고리즘[Teh89]을 사용한다. 이 알고리즘은 좌우 몇 픽셀 떨어진 점을 이용한 지역적인 정보에 의존하기 때문에 인간의 시각 관점에 벗어나 탐지된 경우가 종종 발생하여 오 분할될 여지가 존재

한다. 두 번째, 우세점 탐지 알고리즘(Teh89)은 비스듬히 쓰여진 영상에 대해 오목/볼록 특징점을 못 찾는 경우가 종종 발생하여 미분할필 여지가 존재한다. 세 번째, 두 단계에 걸쳐 수행하는 것이다. 이것은 두 번째 단계 입력이 첫 번째 단계의 출력에 종속되기 때문에 첫 번째 단계에서 오류를 범한 데이터에 대해 재 오류를 범할 여지가 있어 근본적으로 보완하는 방법이라 볼 수 없다.

2.2 새로 제안한 알고리즘

본 논문에서 제안한 방법은 위의 문제점들을 해결하고자 한다. 제안한 알고리즘은 다음과 같다.

입력 : 한 문자 패턴
출력 : 유사 블록 부품으로 분할된 패턴

1. 연결 요소 알고리즘을 적용하여 연결 요소를 찾고 이들 각각을 부품으로 하는 부품 집합 S를 생성한다.
2. S내의 각각의 부품 s_i 에 대해,
 - 2.1 홀이 존재하는지 검사한다.
 - 2.2 홀이 존재하면 홀 분할 과정을 적용하여 s_i 를 두 개의 부품 s_i^1 과 s_i^2 로 분할한다.
 - 2.3 $S \leftarrow S - s_i$; $S \leftarrow S + s_i^1$; $S \leftarrow S + s_i^2$;
3. S내의 각각의 부품 s_i 에 대해,
 - 3.1 s_i 의 볼록 hull을 계산한다.
 - 3.2 볼록 hull에서 최대 오목성을 갖는 오목 영역을 구한다.
 - 3.3 3.2에서 구한 오목 영역에서 분할 기준점(anchor point)를 탐지한다.
 - 3.4 3.3의 기준점에 대응하는 분할 끝점(terminal point)를 탐지한다.
 - 3.5 기준점-끝점으로 s_i 를 s_i^1 과 s_i^2 로 분할한다.
 - 3.6 $S \leftarrow S - s_i$; $S \leftarrow S + s_i^1$; $S \leftarrow S + s_i^2$;
4. S내에 유사 블록한 부품만 존재할 때까지 3을 반복한다.

단계 2.1에서 홀이란 내부 외곽선으로 둘러싸인 안쪽 영역을 말한다. 홀을 가진 부품은 볼록 hull 알고리즘을 적용하여 내부 외곽선 상에 존재하는 특징점 탐지가 불가능하다. 본 논문에서는 이런 부분을 우선적으로 홀을 가지지 않는 두 영역으로 분할하는 작업을 수행한다.

홀 분할(그림2)은 내부 외곽선 상의 점들 중 거리가 가장 먼 두 점을 찾아 이들을 중심으로 수행한다. 가장 먼 점을 중심으로 한 이유는 그 점의 곡률값이 가장 커 굴곡 지점에 위치할 확률이 크기 때문이다. 홀을 갖는 부품의 경우 두 개의 외곽선(contour)을 갖는다. 내부 외곽선의 체인코드 표현을 $P = \langle p_1, p_2, \dots, p_m \rangle$, 외부 외곽선은 $Q = \langle q_1, q_2, \dots, q_n \rangle$ 로 표기하자. P에서 가장 먼 두 점을 찾는 과정은 다음과 같이 할 수 있다.

find (p_i, p_j) such that

$$pointdist(p_i, p_j) = \underset{i, j}{Max} \{ pointdist(p_i, p_j) \}$$
 where $pointdist(p_i, p_j)$ is the Euclidean distance from p_i to p_j .

이제 p_i 과 대응하는 외부 외곽선 상의 q_i 를 구한다. p_i 과 q_i 는 최을 분할하는 경로의 양 끝점 역할을 해야한다. 따라서 q_i 는 p_i 에 가까운 점이어야 한다. 그런데 거리만으로 결정하는 경우 굴곡이 심한 패턴에서 인간 시각에 적절하지 않은 곳을 탐지될 경우가 발생하므로 외부 외곽선 상의 점의 곡률(curvature)도 같이 고려하였다. 외곽선 상의 점 q_i 의 k-곡률(k-curvature)은 q_i 를 중심으로 좌측 k만큼 떨어진 점 q_{i-k} 로부터의 벡터 $\vec{q}_{i-k, i}$ 와 우측 k만큼 떨어진 점 q_{i+k} 까지의 벡터 $\vec{q}_{i, i+k}$ 가 이루는 각도로 정의한다. 그림3은 이를 설명한다.

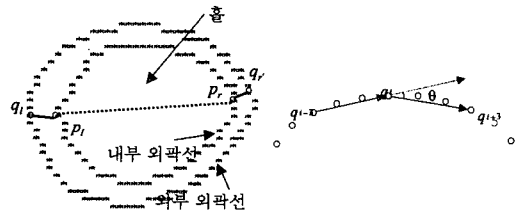


그림 2. 홀 영역 분할

3-곡률(q_i) = θ
 그림 3. 3-곡률

p_i 에 대응하는 q_i 는 거리 상으로 가까울수록 좋고 곡률은 클수록 좋다. 이 두 가지 측정값을 가지고 q_i 를 결정하기 위해 순위값을 사용하였다. 아래 수식은 결정 규칙을 보여준다. p_i 에 대응하는 q_i 도 같은 방법으로 찾는다.

$rankup_distance(q_i)$: $q_i, 1 \leq i \leq n$,를 $pointdist(q_i, p_i)$ 값으로 오름차순 정렬한 순서값
 $rankdown_curvature(q_i)$: $q_i, 1 \leq i \leq n$,를 k-곡률(q_i)값으로 내림차순 정렬한 순서값
 $rank(q_i) = rankup_distance(q_i) + rankdown_curvature(q_i)$

find q_i such that

$$rank(q_i) = \underset{i}{Min} \{ rank(q_i) \}$$

단계2를 수행하면 S안의 각 부품들은 하나의 외곽선만을 가진다. 이들 각각을 유사 블록한 부품으로 분할하기 위하여 단계3을 수행한다.

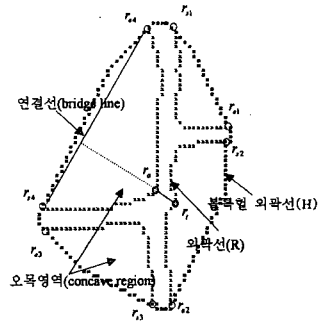


그림 4. 오목 영역 탐지 및 분할

단계3.1에서 오목 영역(그림4)이란, 볼록 hull 알고리즘[Zimmer97]을 적용하여 나온 볼록 hull 영역에서 원 부품 영역을 뺀 나머지 영역을 말한다. 부품 외곽선의 체인 코드 표현을 $R = \langle r_1, r_2, \dots, r_n \rangle$ 로 표기하고, 볼록 hull 외곽선을 H라 하자. H와 R은 겹치는 세그먼트가 있다. 그림4의 경우 네 개의 세그먼트가 있다. 이들 세그먼트의 끝점을 오목 영역 중심으로 r_{si} 와 $r_{ei}(1 \leq i \leq \text{연결선의 개수})$ 로 표기하고

직선 $r_{si}r_{ei}$ 을 연결선이라 하자. 네 개의 오목 영역 중 어느 영역을 먼저 분할하느냐에 따라 다음 번의 분할 모양이 달라질 수 있으므로 최대 오목성을 갖는 오목 영역부터 분할한다. 그 이유는 실험 결과 최대 오목성을 가진 오목 영역부터의 분할이 인간 시각에 가장 가깝게 분할되었기 때문이다. 최대 오목성을 갖는 오목 영역 연결선의 시작점 r_{sh} 과 끝점 r_{eh} 을 찾는 수식은 다음과 같다.

$depth(r_{si}, r_{ei}, r_j)$: $1 \leq i \leq \text{연결선의 개수}, si \leq j \leq ei, r_j$ 에서 직선 $r_{si}r_{ei}$ 까지의 수직 거리

find (r_{sh}, r_{eh}, r_b) such that

$$depth(r_{sh}, r_{eh}, r_b) = \underset{i}{Max} \{ \underset{j}{Max} \{ depth(r_{si}, r_{ei}, r_j) \} \}$$

파라미터 $p1 = r_b$ 에서 $r_{sh}r_{eh}$ 까지의 직선거리 ≥ 3
 파라미터 $p2 = pointdist(r_{sh}, r_{eh}) \geq 5$

부품의 최대 오목 영역의 최대 높이값($p1$) < 3 그리고 연결선의 길이($p2$) < 5라면 이 부품은 유사 블록하게 이루어진 부품이다. 제안한 방법은 두 파라미터만을 가지고 분할을 결정한다. 유사 블록하지 않다고 판명된 부품에 대해서 단계3.3을 수행한다. 단계3.3에서 분할 기준점(anchor point) r_a 을 위 수식에서 찾은 r_b 로 사용할 수 있지만 곡률값을 같이 고려하여 인간 시각에 가까운 점을 탐지하기 위하여 아래 과정을 수행한다.

$rankup_verticaldistance(r_i)$: $r_i, sh \leq i \leq eh$,를 $depth(r_{sh}, r_{eh}, r_i)$ 값으로 오름차순 정렬한 순서값
 $rankdown_curvature(r_i)$: $r_i, sh \leq i \leq eh$,를 k-곡률(r_i)값으로 내림차순 정렬한 순서값
 $rank(r_i) = rankup_verticaldistance(r_i) + rankdown_curvature(r_i)$
 find r_a such that

$$rank(r_a) = \underset{i}{Min} \{ rank(r_i) \}$$

r_a 에 대응하는 분할 끝점(terminal point) r_t 는 거리 상으로 가까울수록 좋고 곡률은 클수록 좋다. r_t 탐지는 홀 분할 과정에서 사용한 방법을 사용하지만, 연결선(bridge line)을 중심으로 바깥쪽 외곽선만을 대상으로 검사한다.

$rankup_distance(r_t) : r_t, eh \leq i \leq sh$,를 $pointdist(r_i, r_a)$ 값으로 오름차순 정렬한 순서값
 $rankdown_curvature(r_t) : r_t, eh \leq i \leq sh$,를 $k-곡률(r_i)$ 값으로 내림차순 정렬한 순서값
 $rank(r_i) = rankup_distance(r_i) + rankdown_curvature(r_i)$
 find r_t such that

$$rank(r_i) = \text{Min}\{rank(r_t)\}$$

단계3에서 찾은 기준점 r_a 와 끝점 r_t 를 중심으로 부품을 분할한다. 분할된 두 영역은 부품 집합 S에 추가되어 단계3을 반복 수행하며 만약 S내에 모든 부품이 유사 불록하다면 종료한다. 그림5는 입력 영상에서 출력 영상까지 수행되는 과정을 나타낸다. 그림5(a)는 입력 영상을 그림5(b)는 출력 결과를 나타낸다. 그림5(c) 홀 탐지와 분할을 그림4(d)는 분리된 영역에서의 분할 과정을 나타낸다.

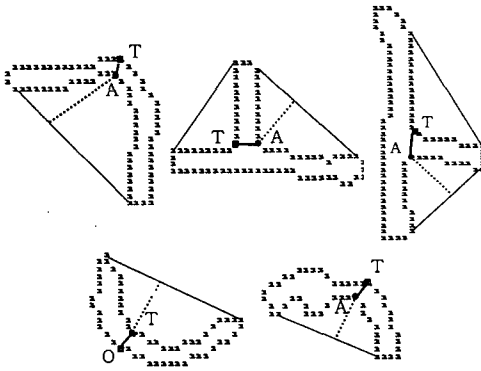
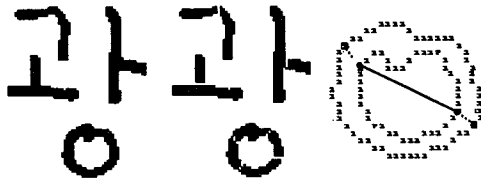


그림 5. 전체적 흐름도

3. 기존 방법과의 비교 및 실험

그림6은 본 논문의 알고리즘을 PE92 데이터베이스에 적용하여 얻어진 결과를 나타낸다.

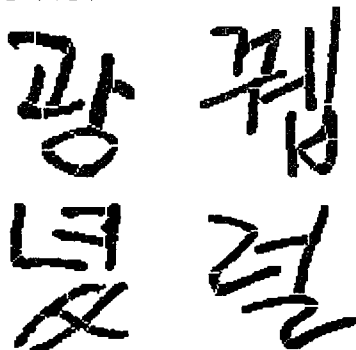


그림 6. 결과

본 논문은 [박정선99] 논문과 몇 가지에 대해 비교 실험하였다. 표1과 같이 수행시간과 분할된 부품 개수는 비슷하였지만, 정확한 분할 측면에서는 월등한 성능 향상이 있었다.

표1. 성능 비교

	수행시간	분할된 부품개수
[박정선99]	0.24	9.04
제안한 방법	0.23	9.02

제안한 방법에서 나타난 문제점은 'X' 모양에서의 분할이다: 그림7 '남'자와 '남'의 초성 'ㄴ'과 중성 'ㅌ'가 연결된 지점에서 'X' 모양과 비슷하게 형성된 부품을 초성과 중성을 분할하지 않고, 대각선으로 분할된 것을 볼 수 있다. 이것은 한글의 구성 지식을 첨가하지 않은 상태에서 분할하기 때문에 이런 부분은 분할 한계를 가지고 있다. 이 문제를 해결하기 위해서는 지식 기반의 후처리가 필요하다 하겠다.

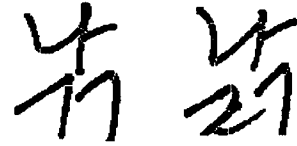


그림 7. 'X' 모양 분할 데이터

4. 결론 및 향후 연구 과제

문자 인식에 있어서 최 추출의 근본적인 문제를 해결하기 위해서는 원 영상을 훼손시켜 추출한 세선화 방법보다는 원 영상 그대로의 정보를 유지하고 있는 모양 분해 알고리즘이 더 적합하다. 그런 의미에서 한글에 모양 분해 알고리즘을 적용한 것은 당연하다고 하겠다. 본 논문의 알고리즘은 2.2절에서 기술한 것처럼 두 개의 파라미터만으로 조절가능하며 간단한 프로시저로 구성되어 있다. 하지만, 'X' 모양 부분에서의 분할은 향후 해결해야할 문제점으로 남아 있다. 그리고 필기 한글에 국한하여 수행하지 않고 모든 문자에 적용하는 것 또한 향후 연구 과제로 남아 있다.

Reference

[김수형98] 김수형, 정선화, 오일식, "필기 한글 문자의 오프라인 인식에 관한 사례 연구," 한국 정보과학회 '98 가을 학술발표논문집(II), 아주대학교, pp.396-398, 1998.
 [박정선99] 박정선, 오일식, "필기 한글 패턴을 위한 두 단계 모양 분해 알고리즘," 한국 정보과학회 '99 가을 학술발표논문집(III), 광운대학교, pp.464-466, 1999.
 [이성환93] 이성환, 박희선, "한글 인식의 사례연구 : 최근 5년 동안의 연구 결과를 중심으로," 제1회 문자인식 워크샵 발표 논문집, 충북대학교, pp.3-46, 1993.
 [Jiang93] J. Jiang, W. Kim, and H. Tominaga, "Recognition and representation of text characters using rubber band," proceedings of 4th International Conference on Document Analysis and Recognition, pp.955-958, 1993.
 [Latecki99] L.J. Latecki and R. Lakamper, "Convexity rule for shape decomposition based on discrete contour evolution," Computer Vision and Image Understanding, Vol.73, No.3, pp.441-454, 1999.
 [Lin95] J.Y. Lin and Z. Chen, "A Chinese-character thinning algorithm based on global features and contour information," Pattern Recognition, Vol.28, No.4, pp.493-512, 1995.
 [Rosin2000] P.L. Rosin, "Shape partitioning convexity," IEEE Tr. on Systems, Man, and Cybernetics, Vol.30, No.2, pp.202-210, 2000.
 [Tseng92] L.Y. Tseng and C.T. Chuang, "An efficient knowledge-based stroke extraction method for multi-font Chinese characters," Pattern Recognition, Vol.25, No.25, pp.1445-1458, 1992.
 [Teh89] C.H. Teh and R.T. Chin, "On the detection of dominant points on digital curves," IEEE Tr. on Pattern Analysis and Machine Intelligence, Vol.11, No.8, pp.859-872, August 1989.
 [Zimmer97] Y. Zimmer, R. Tepper, and S. Akselrod, "An improved method to compute the convex hull of a shape in a binary image," Pattern Recognition, Vol.30, No.3, pp.397-402, 1997.