

모양 분해를 이용한 필기 한글 문자의 골격선 추출

(Extraction of Skeletons from Handwritten Hanguk Characters using Shape Decomposition)

홍기천[†] 오일석^{**}

(Ki-Cheon Hong) (Il-Seok Oh)

요약 필기 한글 문자 인식을 위한 획 추출 방법으로 많이 사용되는 세션화는 패턴을 왜곡시키는 문제점을 안고 있다. 본 논문은 모양 분해 알고리즘을 사용한 한글 문자의 골격선 추출 방법을 제안한다. 먼저 모양 분해 알고리즘을 사용하여 입력 패턴을 유사 불복한 부품 집합으로 분해한다. 모양 분해된 패턴에서 결합 부품을 탐지하고, 이 부품과 인접한 부품들로부터 골격선을 구한다. 그 다음 결합 부품과 인접하지 않은 부품들에 대한 골격선을 추출하고 골격선의 연결성을 보장하기 위해서 선분 연장을 수행한다. 본 논문에서 추출한 골격선과 세션화로 추출한 골격선을 비교하기 위하여 다섯 가지 비교 기준을 설정하고, 이를 기반으로 비교 분석하였다. 본 논문에서 제안한 방법이 여러 기준에서 세션화-기반 방법보다 우수함을 보였다.

Abstract The thinning process which is commonly used in extracting skeletons from handwritten Hanguk characters has a problem of distorting the original pattern shapes. This paper proposes a method of skeleton extraction using a shape decomposition algorithm. We decompose the character pattern into a set of near convex parts using a shape decomposition algorithm. From the shape-decomposed pattern, we detect the joint parts and extract the skeletons from the parts incident to the joint parts. Then the skeletons not incident to the joint parts are extracted. Finally, the process of skeleton extension is performed to ensure the connectivity. We setup five criteria for the comparison of quality of skeletons extracted by our method and the thinning based method. The comparison shows the superiority of our method in terms of several criteria.

1. 서론

한글 인식에 관한 연구는 1970년대 초부터 시작되었으며, 인쇄 한글 인식 문제를 중심으로 연구되어 왔다 [1]. 그 결과 최근에는 인쇄 한글 인식을 위한 상용 시스템들이 많이 선보이고 있다 [2]. 그러나 필기 한글 인식의 경우에는 아직도 많은 난제들을 가지고 있기 때문

에 실용화에는 아직 거리가 멀다 [3]. 이러한 문제들은 다양한 필기자에 따른 글자 모양의 심한 변화와 동일한 필기자인 경우에도 글자의 위치나 필기 도구에 따른 심한 변화 등에 기인한다. 또한 한글 구성상 2차원 상에서 자소의 분리와 인식을 처리해야 한다는 문제로 어려움을 더해주고 있다.

한글은 부류수가 방대하고 글자간 유사성이 높기 때문에 문자 패턴을 구성하는 획(stroke)을 추출하고 이를 분석하여 인식하는 방법이 주류를 이루고 있다. 때문에 한글 패턴에 대한 성능 높은 획 추출 알고리즘은 매우 중요하다. 한문도 부류수가 방대하다는 이유 때문에 획 분석에 의한 방법이 많이 사용되고 있다. 한문에 대한 획 추출은 많은 연구가 수행되어 있다 [4]. 하지만 획 추출의 중요성에도 불구하고 필기 한글 패턴에서 높은 성능의 획 추출에 대한 연구는 거의 없는 실정이다. 현

* 이 연구는 한국과학재단 핵심과제 981-1212-036-2 지원을 받았음

† 학생회원 : 전북대학교 컴퓨터과학과
kchong@cs.chonbuk.ac.kr

** 종신회원 : 전북대학교 컴퓨터과학과 교수
영상정보신기술연구소
isoh@moak.chonbuk.ac.kr

논문접수 : 2000년 1월 12일

심사완료 : 2000년 3월 7일

제까지 개발된 필기 한글 인식 방법은 대부분 획 추출을 위해 세선화(thinning)에 의존하고 있다

한글 인식을 위해서 세선화 연산을 사용한 연구가 많이 있다. 이 연구들은 세선화 영상으로부터 끝점, 교차점, 굴곡점등과 같은 특징점을 추출하였다. 김진형은 추출된 특징점들을 이용하여 세선화된 패턴을 획들의 집합으로 구성하고 계층 그래프를 생성하여 그래프 매칭을 수행하였다 [5]. 김항준은 문자의 오른쪽-위 부분에서 수직 모음 인식을 위해서 특징점을 사용하였다 [6]. 이은주는 추출된 특징점을 이용하여 수직 모음을 추정하는 방법을 사용하였다 [7]. 김태균은 추출된 특징점들을 이용하여 입력 패턴을 그래프화하였다 [8]. 박래홍은 HMM을 이용하여 필기 한글과 영숫자를 인식하였다. HMM의 인수 중 하나인 관찰열(observation sequence)로서 특징점을 이용하였다 [18].

위에서 살펴본 바와 같이 한글 인식 방법 대부분이 세선화에 의존하고 있기 때문에 세선화 과정에서 초래되는 패턴의 왜곡현상과 같은 문제점을 안고 있다. 즉, 획 추출 과정에서의 왜곡으로 인해 최종 인식률에 나쁜 영향을 미치고 있다. 획 추출 과정에 대해 보다 높은 성능을 갖는 모듈을 개발하여, 인식 시스템에서 획 추출부분을 이 모듈로 대체하면 인식률을 향상시킬 수 있을 것이다.

황영섭은 세선화의 이러한 문제점을 지적하고, 문자 영상을 단위 영상으로 분할되었다는 가정하에, 이들을 조합하여 자소 후보를 만들어 신경망으로 인식하였다 [9]. 이 방법은 세선화를 사용하지 않음으로써 패턴 왜곡이나 정보손실등의 문제를 피하였다. 하지만 이 방법에서 중요한 문제인 단위 영상으로의 분할 및 자소 단위로의 분할에 대한 문제를 다루지 않고, 자소간에 접촉이 없다는 가정을 사용하였다.

조성배는 영상의 세선화를 위한 지식기반 시스템을 제안하였다 [10]. 이 방법에서는 문자의 세선화 과정에서 생기는 왜곡 현상 중에서 인접한 획에 의해 발생하는 가획 현상, 원 영상의 접촉된 획에 의해 발생하는 왜곡 현상, 교차 영역에서 발생하는 Y형태 왜곡 현상과 같은 세 가지의 왜곡 현상을 많이 해결하였다. 그러나 획의 방향이 급격히 변하는 지점에서 필요없는 가지가 생기는 문제는 해결하지 못했다.

서장원은 명암 영상으로부터 획을 추출하였다 [11]. 이 방법으로 추출된 골격선은 세선화 방법에서 초래되는 가획을 감소시켰다. 그러나 획이 접촉된 부분에서는 아직도 계곡과 같은 왜곡현상이 발생한다.

박정선은 두 단계 모양 분해(shape decomposition)

알고리즘을 사용하여 필기 한글 패턴을 유사 볼록(near convex)한 부품들로 나누었다 [12]. 첫 번째 단계는 패턴의 외곽선상에 존재하는 점들 중, 오목 특징점과 볼록 특징점을 탐지하고, 이로부터 T-접점과 B-접점을 추출하여 부품들로 분해하였다. 두 번째 단계는 첫 번째 단계에서 탐지하지 못했던 특징점을 찾기 위하여 볼록 쉘(convex hull) 알고리즘을 적용하여 특징점을 추출하였다. 이 단계는 첫 번째 단계에서 탐지하지 못했던 T-접점과 B-접점을 탐지하고 결합 부품을 탐지하였다. 이 논문은 필기 한글 패턴에 대해서 모양 분해를 수행한 후, 골격선(skeleton) 추출 과정을 다루지 않았다.

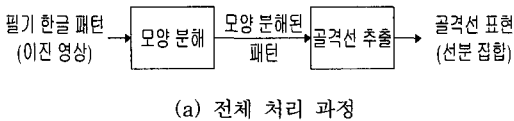
본 논문은 모양 분해 알고리즘을 이용하여 이미 유사 볼록 부품으로 분해된 필기 한글 패턴으로부터 골격선을 추출하는 문제를 다룬다. 모양 분해를 위해서 본 논문은 [12]를 사용하였다. [12]는 필기 한글 패턴에 대한 모양 분해를 시도한 최초의 논문으로서 비교적 좋은 성능을 실험으로 입증하였다.

본 논문에서 제안한 골격선 추출 알고리즘은 크게 두 단계로 구성된다. 첫 번째 단계는 결합 부품(joint part)의 탐지이고, 두 번째 단계는 탐지된 결합 부품을 중심으로 각각의 유사 볼록 부품 집합으로 모양 분해된 패턴으로부터 골격선을 추출하는 것이다. 각 골격선은 두 개의 끝점으로 정의되는 선분(line segment)이다. 결합 부품을 탐지하는 이유는 필기 한글 패턴내에는 획과 획이 서로 교차하는 지점이 자주 발생하기 때문이다. 이러한 지점은 여러 획이 복잡하게 만나는 곳으로서 특별한 주의없이 골격선을 추출하면 문자 모양을 잘 표현하는 결과를 얻을 수 없다. 때문에 이러한 지점은 여러 획을 결합해주는 역할을 하는 결합 부품으로 탐지한다. 이들 결합 부품의 중점은 인접해 있는 다른 부품들에게 하나의 끝점을 제공한다. 결합 부품과 인접한 부품들의 또 다른 끝점은, 부품을 둘러싸고 있는 외곽선상의 점들 중에서 결합 부품의 중점을 기준으로 가장 거리가 먼 점을 선택하여 결정한다. 이 두 점으로 정의되는 선분이 인접 부품을 표현하는 골격선이 된다.

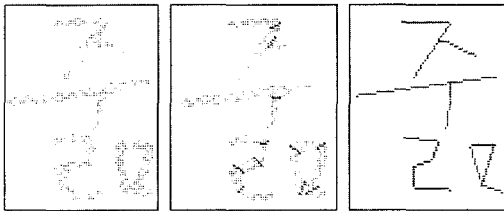
결합 부품과 인접하지 않은 부품들에 대한 골격선을 구하기 위하여, 부품들의 외곽선상에 존재하는 점들간의 거리를 계산하여, 이 중 가장 거리가 먼 두 개의 점을 선택한다. 이 두 점으로 정의되는 선분이 골격선이 된다. 이런 방식으로 골격선을 찾고 나면, 두 개의 획이 T 형태로 만나는 T-접점에서는 이웃한 두 개의 부품에서 추출된 골격선이 만나지 않는다. 이러한 문제를 해결하기 위하여 골격선을 이루는 선분을 연장하는 과정을 거친다.

기준에 많이 쓰이는 세선화 알고리즘으로 추출된 골격선과 본 논문에서 추출한 골격선을 다섯 가지의 기준을 설정하여 비교 분석하였다. 비교 대상으로 사용한 세선화 알고리즘은 Zhang-Suen [13]과 SPTA [14]이다. 이 두 알고리즘은 국내 문자 인식 그룹에서 많이 사용하고 있는 방법들이다. 비교 실험 결과, 본 논문에서 제안한 골격선 추출 방법이 잡영 가지 발생, 획의 직선성 유지, 획의 길이 유지, 획의 교차 지점 처리, 복원성 기준에서 훨씬 나음을 보였다.

2 장에서는 최초 입력인 필기 한글 패턴으로부터 최종 출력인 골격선 집합을 추출하는 전체 과정을 보이고, 이 중 본 논문의 연구 범위에 대해 기술한다. 3 장에서는 본 논문이 제안한 골격선 추출 방법에 대해서 자세히 기술한다. 4 장에서는 기존의 세선화 기반 골격선 추출 방법과 본 논문에서 제안한 방법을 PE92 데이터베이스에 대한 실험 결과를 가지고 비교 분석한다. 마지막으로 제 5 장에서는 결론과 향후 연구 과제에 대해서 논한다.



(a) 전체 처리 과정



(b) 처리 과정 예시

그림 1 모양 분해를 이용한 필기 한글 패턴의 골격선 추출 과정

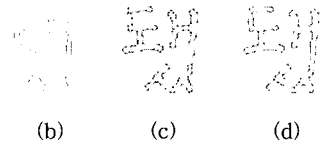
2. 모양 분해

그림 1은 이진 영상으로 표현되는 필기 한글 패턴으로부터 골격선을 추출하는 전체 처리 과정 및 예제 패턴을 보여준다. 모양 분해 단계는 이진 영상에 대해서 모양 분해 알고리즘을 이용하여 유사 블록한 부품들의 집합으로 분해한다. 골격선 추출 단계는 분해된 한글 패턴을 입력으로 하여 선분으로 표현되는 골격선을 추출한다. 본 논문은 골격선 추출 단계를 위한 알고리즘을

제시한다. 모양 분해를 위해서는 특정한 알고리즘에 국한될 필요없이 유사 블록으로 분해해 주는 임의의 알고리즘을 채택하여 사용할 수 있다. 본 논문에서는 [12]에서 제안한 알고리즘을 채택하였고, 실험에서도 이 알고리즘에서 출력된 패턴을 입력으로 사용하였다.

모양 분해 알고리즘은 다음 다섯 단계로 구성되어 있다. 보다 자세한 사항은 [12]를 참조하기 바란다.

(1) 특징점 탐지: 입력 패턴으로부터 체인 코드를 구한 다음, 이 체인 코드에 우세점(dominant point) 탐지 알고리즘 [15]을 적용하여 방향이 급격히 변하는 점을 추출한다. 그러나 이 알고리즘은 너무 많은 우세점을 탐지하는 문제점이 있다. 이렇게 많이 발생한 우세점들 중 미세한 돌출 부분에서 생기는 것들을 제거하는 과정이 필요하다. 이를 위하여 한 우세점에 대해 좌우 우세점과의 각도를 계산하여 이 각도가 임계값 T 이상이면 잡음에 의한 우세점이라 간주하고 제거한다. 이 과정을 거쳐 남아있는 우세점을 특징점이라 정의하고, 이들을 다음 단계에서 T-점점과 B-점점 탐지에 사용한다. 그림 2는 입력 패턴으로부터 특징점 탐지 과정을 보여준다.



(a) (b) (c) (d)

그림 2 특징점 탐지 (a) 입력 패턴, (b) 외곽선 추출 (체인코드 형성), (c) 우세점 추출(Teh_Chin 알고리즘), (d) 특징점 추출(검은점: 오목 특징점, 회색점: 볼록 특징점)

(2) T-점점 탐지: T-점점은 ‘┌’, ‘┐’, ‘└’ 등과 같은 패턴내의 두 개의 획이 서로 T형태로 만나는 지점에서 형성된다. T-점점은 오목 특징점과 오목 특징점의 쌍에 대해 기하학적 조건을 검사하여 탐지한다.

(3) B-점점 탐지: B-점점은 ‘ㄱ’, ‘ㄴ’, ‘ㄷ’, ‘ㄹ’ 등과 같은 패턴내에서 방향이 급격히 변하는 지점에서 생성된다. B-점점은 오목 특징점과 볼록 특징점의 쌍에 대해 기하학적 조건을 검사하여 탐지한다. 그림 3은 탐지된 T-점점과 B-점점을 이용하여 분할한 예제를 보여준다.

(4) 유사 블록 검사: 이 단계에서는 위의 (3) 단계까지 수행한 후 얻어진 부품들이 유사 블록인지 판정하기 위하여 볼록 현 알고리즘을 사용하여 검사를 수행한다. 필기 한글 모양 특성상 위의 (3) 단계까지 만을 가지고 분할이 완료되는 경우도 있지만 특징점의 불완전한 탐



(가는 선: T-접점 분할, 굵은 선: B-접점 분할)

그림 3 T-접점과 B-접점 분할

지에 의해 유사 블록 부품으로 분해되지 않는 경우도 종종 발생한다. 예를 들어, 그림 3의 '됐'의 '냈'에서 중성과 중성이 접촉된 부분이 분해가 되지 않았다. 이러한 부분을 분해하기 위하여 블록 쉼 알고리즘 [16]을 이용하여 분해된 모든 부품들을 대상으로 블록성을 판정한다. 만약 유사 블록하지 않다고 판정된 부품에 대해 다음 단계를 이용하여 분할을 한번 더 시도한다.

(5) 오목 영역 중심으로 분할: 블록 쉼 알고리즘은 원래의 부품을 포함하면서 이 부품이 가질 수 있는 가장 작은 블록 쉼 영상을 구하기 위하여 쓰여진다. 블록 쉼 영상을 구성하는 화소는, 원래 부품에 속하는 화소와 부품의 오목한 부분을 채우는 화소의 두 가지로 구성된다. 후자의 영역을 오목 영역이라 한다. 분할은 오목 영역의 가장 깊은 점을 중심으로 수행된다. 오목한 영역이 한 개일 때는 B-접점으로 분할하고, 두 개 이상일 때는 T-접점으로 분할한다.

그림 4는 위에 기술한 모양 분해 알고리즘을 수행한 후, 분할된 문자 패턴의 예를 보여준다. T-접점의 끝점은 '@', B-접점의 끝점은 '&', 분할경로는 '*'로 표시하였다.

3. 골격선 추출 알고리즘

이 논문에서 제안하는 골격선 추출 알고리즘은 그림 4와 같은 유사 블록 부품 집합으로 분해된 패턴을 입력으로 받아, 각 부품을 대표하는 선분(line segment)을 계산하여 골격선 표현을 추출하는 것이다. 그림 4에서 보듯이 부품의 유형이 매우 다양하기 때문에 골격선을 구하는 것은 단순한 문제가 아니다. 부품을 분할한 접점들을 기준으로 유형을 나누어 보면 독립 부품, 하나의 접점을 갖는 부품, 여러 개의 접점을 갖는 부품으로 나눌 수 있다.

독립 부품은 외곽선상에 어떠한 접점도 가지고 있지 않는 부품을 의미한다. 'l', 'r'와 같이 자소가 하나의 획으로 구성되는 경우에 많이 발생한다. 이러한 형태의 부품은 선분의 양 끝점을 계산하는 것이 단순하다.

접점을 하나만 가진 부품의 예로는 그림 4의 '뺨'자에서 'a', 'c', 'b'등으로 표시된 부품과 '춌'자에서 'a'로 표시된 부품이 있다. 이 경우 '뺨'의 'a', 'c', 'b'는 접점에서 끝점 하나를 계산하고 다른 끝점은 획의 반대편에서 계산하면 된다. 반면에 '춌'자의 'a'로 표시된 부품은 부품 영역의 끝이 아닌 중간 변에 T-접점이 존재한다. 이 경우는 획의 양 끝에서 선분의 두 끝점을 계산하고, T-접점은 인접 부품에서 추출된 선분과의 연결성을 확보하는데 사용한다.

마지막으로 두 개 이상의 접점을 포함하는 부품의 경우를 생각해 보자. 이러한 형태의 부품은 접점의 개수와 위치가 매우 다양하다. 그림 4의 '뺨'자에서 'i', 'o'로 표

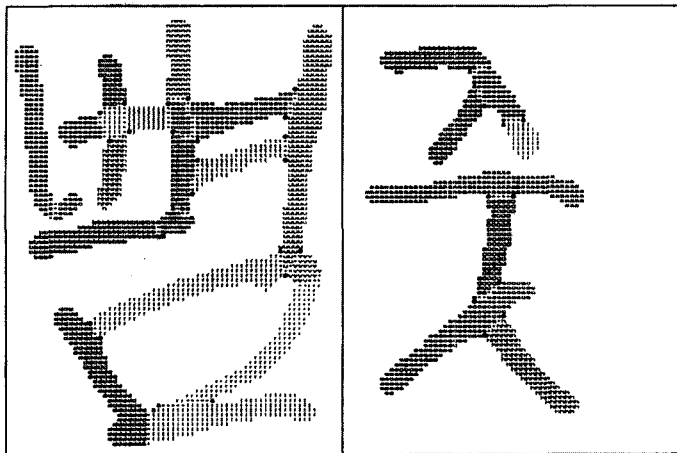


그림 4 모양 분해 결과

시된 부품들은 접점들이 서로 한 점을 공유하면서 부품을 완전히 둘러싸고 있다. 이러한 부품은 인접한 부품들을 한 점에서 결합해주는 역할을 한다. 이러한 부품을 **결합 부품(joint part)**이라 하자. 'u'로 표시된 부품의 밑 부분에서는 두 개의 접점이 한 점을 공유하지만, 다른 쪽은 공유를 하지 않고 있다. 또한 '줏'자의 'c'로 표시된 부품은 두 개의 접점을 포함하는데 모두 부품의 중간 변에 위치하고 있고 이 두 접점은 평행하다. 이런 지점은 세 개 이상의 획이 한 점을 중심으로 교차하는 형태를 이룬다. 이러한 지점을 부주의하게 처리하면, 쓸데 없이 많은 선분이 복잡하게 연결된 형태의 골격선이 생성된다. 이러한 지점의 효과적인 처리를 위하여 제안한 알고리즘은 이러한 지점에서 결합 부품을 탐지한다. 이 결합 부품은 하나의 결합점을 제공하고 이 점은 인접한 획들이 공유하는 끝점으로 사용된다.

이 이외에도 '뽻'자의 's', 't', 'f', 'r', '줏'자의 'g'와 같이 부품의 양 끝에 접점이 존재하는 경우도 있다. 또한 '뽻'자의 'u', 'p', 'd'등과 같이 부품의 변에 접점이 한 개 이상 존재하는 경우도 있다.

앞에서 살펴본 다양한 경우를 고려하여, 그림 5에서 보는 바와 같은 두 단계를 거치는 골격선 추출 알고리즘을 제안한다. 이 알고리즘은 이 두 단계를 통해 다양한 경우를 체계적으로 처리한다.

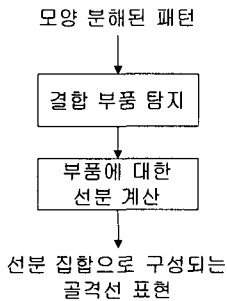


그림 5 제안한 골격선 추출 알고리즘의 처리 단계

3.1 결합 부품 탐지

결합 부품을 효과적으로 탐지하기 위해 두 가지 경우로 나누었다. 첫 번째는 두 개 이상의 접점(T-접점 또는 B-접점)이 끝점을 공유하는 경우이다. 두 번째는 끝점 공유는 없지만 접점간의 거리가 가깝고, 접점의 양 끝점을 서로 이어 생성되는 사각형의 면적이 임계값 이하인 경우이다. 이러한 구분과 아래 알고리즘 설계는 PE92 데이터베이스에 있는 많은 샘플들에 대해 2 장의 모양 분해를 수행하고 그 결과를 주의 깊게 관찰하여

수행하였다. 관찰 결과, 결합 부품 탐지의 필요성을 결정하였고, 탐지 과정을 위의 두 가지 경우로 나누면 획이 세 개 이상 복잡하게 만나는 지점의 대부분을 처리할 수 있음을 알 수 있었다.

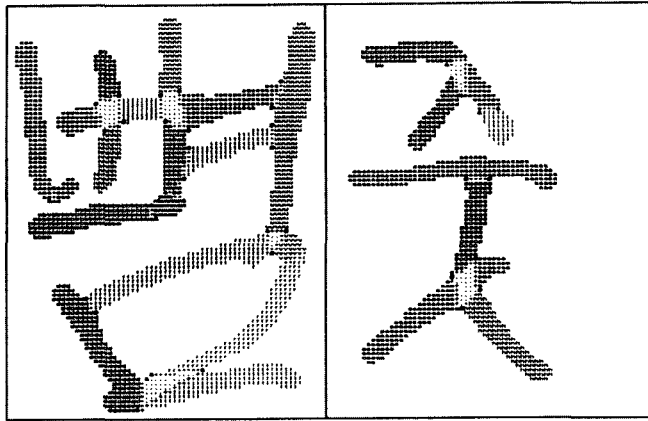
(1) 끝점 공유를 통한 탐지

모양 분해된 패턴의 접점들을 조사하여 끝점을 공유하는 경우를 모두 찾는다. 이 때 끝점을 공유하는 접점들이 폐곡선을 이루는지 여부에 따라 두 가지 경우가 있다. 그림 4의 '뽻'자에서 부품 'i'와 'o'를 둘러싸고 있는 네 개의 접점이 폐곡선을 이루고 있다. 이 경우는 부품 'i'와 'o' 자체를 결합 부품으로 결정한다. '뽻'자의 부품 'u'의 아래 부분에 두 개의 접점이 한 개의 끝점을 공유하고 있다. 이 경우도 이 두 개의 접점을 중심으로 결합 부품으로 탐지하는데, 접점 두 개만으로는 완전한 결합 부품을 형성하지 못하기 때문에 공유되지 않는 두 개의 끝점에 새로운 분할 경로를 삽입하여 새로운 부품을 만들고 이 부품을 결합 부품으로 결정한다. 그림 4(a)는 새로운 분할 경로는 '+', 탐지된 결합 부품은 '.'로 표시하였다.

(2) 거리와 면적에 의한 탐지

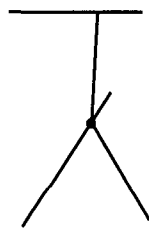
위의 경우는 접점들이 끝점 공유의 조건을 가지고 결합 부품을 탐지하였다. 그러나 많은 필기 한글 패턴에서는 여러 획이 분기하는 지점에서 생성된 접점들이 항상 끝점을 공유하는 것은 아니다. 즉, 3 개 이상의 획이 분기하는 지점에 생성된 접점들이 끝점을 서로 공유하지 않으면서 결합 부품으로 탐지해야 하는 경우도 많다는 것이다. 이러한 경우는 분기 지점에 위치한 접점들이 일정한 거리 이내에 있으며, 접점들이 형성하는 사각형의 면적이 원래 부품의 면적의 절반 이상을 차지하지 않는다는 공통점을 발견할 수 있다. 사각형의 면적이 원래 부품의 면적의 절반 이상을 차지하게 되면 원래 부품의 면적에서 사각형의 면적을 뺀 나머지 부분을 무시해도 추출된 골격선과 원래 입력 패턴과의 차이는 거의 없다. 예를 들어, '줏'자의 'u'와 's'이 만나는 지점에서 세 개의 획이 한 지점에서 만나는 경우 결합 부품을 탐지하지 않으면 그림 6(d)의 바람직하지 않은 골격선 결과를 얻는다. 이 경우는 부품 'c'에서 두 개의 마주보는 접점이 일정 거리 이내에 있으며 접점들이 형성하는 사각형의 면적이 원래 부품의 면적보다 절반 이하이다. 이러한 현상은 '뽻'의 부품 'i'과 '줏'의 부품 'e'에서도 발생하였고 많은 필기 한글 패턴에서 관찰되었다.

이로 미루어 볼 때 접점들이 끝점을 공유하지 않는 경우에 결합 부품을 탐지하기 위해서는 접점간의 거리와 면적의 조건을 동시에 만족하는 부품만 결합 부품으

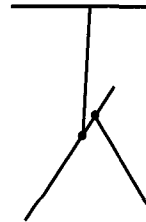


(a) 문자 '뺨'

(b) 문자 '춧'



(c) 결합 부품을 탐지한 경우



(d) 결합 부품을 탐지하지 않았을 경우

그림 6 결합 부품 탐지

로 탐지한다. 즉, 두 개 접점간의 거리가 임계값 T_1 보다 짧으면 이 두 개 접점의 양 끝점을 이어서 생성되는 사각형의 면적을 조사한다. 만약 이 면적이 임계값 T_2 보다 작으면 이 두 접점으로 만들어진 사각형을 결합 부품으로 탐지하였다.

두 접점간의 거리 임계값 T_1 으로는 평균 획 두께를 사용하였다. 이 거리 조건을 만족하는 부품을 대상으로 두 개 접점의 양 끝점을 서로 이어서 형성되는 사각형의 면적과 원래 부품의 면적을 비교한다. 사각형의 면적은 사각형내의 화소의 수이고 원래 부품의 면적은 부품을 구성하는 화소의 수이다. 만약 사각형의 면적과 원래 부품 면적의 비율을 조사하여 임계값 T_2 이하이면 결합 부품으로 탐지하였다. 여러 문자에 대해서 실험해 본 결과 임계값 T_2 를 40%로 결정하였다. 즉, 사각형의 면적이 원래 부품 면적의 40%이하이면 결합 부품으로 탐지한다.

위의 거리와 면적 조건을 동시에 만족하는 경우는 그

림 4의 문자 '춧'의 종성 'ㅈ'과 초성 'ㅈ', 그리고 문자 '뺨'의 'ㅁ'의 아래 부분이다. 그림 4의 '뺨'에서 부품 'm'의 경우는 거리 조건은 만족을 하지만 두 개의 접점이 형성하는 사각형의 면적이 임계값 T_2 이상이므로 결합 부품으로 탐지하지 않았다.

위의 거리와 면적 조건을 만족한 두 개 접점만으로는 완전한 결합 부품을 형성하지 못하기 때문에 접점의 끝점을 중심으로 새로운 분할 경로('+'로 표기)를 삽입한다. 그림 6(b)의 '뺨'자의 부품 'l', '춧'자의 부품 'c'와 'e'에서 이렇게 만들어진 새로운 결합 부품을 볼 수 있다.

3.2 선분 생성

이 절에서는 위에서 추출한 결합 부품 (그림 6)을 이용하여, 각각의 부품에서 골격선 표현을 구성할 선분을 추출하는 과정에 대해서 기술한다. 이 과정을 효과적으로 수행하기 위해 부품을 결합 부품에 인접해 있는 지에 따라 인접한 부품 (그림 6의 '뺨'자에서 부품 'g','e','h','j','n','p','s','u','f','v','d','r','l'과 '춧'자에서

부품 'b', 'd', 'e', 'g', 'c', 'h')과 인접하지 않은 부품 (그림 6의 '뺨'자에서 'a', 'c', 'b', 'm', 't'와 '췌'자에서 부품 'i', 'a')로 구분하여 처리한다. 이 과정은 크게 결합 부품과 인접한 부품에서 선분 추출, 결합 부품과 인접하지 않은 부품들에서 선분 추출, 그리고 선분간의 연결성 확보를 위한 선분 연장의 세 단계로 구성된다.

(1) 결합 부품과 인접한 부품

결합 부품의 중점은 인접한 부품들에 대해서 하나의 끝점을 제공한다. 그림 7에서 결합 부품의 중점은 'o'로 표시하였다. 그림 6(a)에서 결합 부품과의 인접성에 따라 다양한 경우가 있을 수 있다. 부품 'j'는 양끝이 결합 부품에 인접한 경우이다. 부품 'g', 'e', 'h', 'n'의 한 끝은 결합 부품에 인접하고 다른 끝은 획의 끝에 해당한다. 또한 부품 'p', 's', 'u', 'v', 'f', 'r', 'l'은 한끝은 결합 부품에 인접하고 다른 끝은 접점에 해당한다. 이 중 부품 'p'는 변에 또 다른 접점을 가지고 있다.

이러한 다양한 경우를 일관성있게 처리하기 위해 제안한 알고리즘은 다음과 같은 과정을 거친다. 하나의 부품에 대해 결합 부품의 중점으로부터 가장 먼 점을 찾는다. 이를 위해 부품을 싸고 있는 외곽선상의 모든 점까지의 거리를 계산하여 가장 먼 점을 찾을 수도 있고, 외곽선에서 추출한 굴곡점만을 대상으로 거리를 계산하여 가장 먼 점을 찾을 수도 있다. 본 논문에서는 계산 효율을 위하여 모양 분해 과정에서 찾아 놓은 우세점(dominant point)를 굴곡점으로 사용하여 굴곡점 (그림 7에서 '\$'로 표시)중에서 가장 먼 점을 찾았다. 접점을 이루는 두 끝점도 굴곡점이다. 이 가장 먼 점을 f 라 하자.

현재 그림 6에서 '뺨'과 '췌'가 만나는 지점의 결합 부품을 처리한다고 가정하자. 부품 'p'의 경우는 가장 먼 점 f 가 접점의 끝점('o'로 표시된 점)에 해당한다. 이를 경우 1이라 하자. 부품 'j'의 경우는 f 가 접점의 끝점에 해당하고 이 접점이 다른 결합 부품에 속해 있다. 이를 경우 2라 하자. 부품 'n'의 경우는 f 가 접점에 속하지 않는 굴곡점이다. 이를 경우 3이라 하자.

위에 살펴본 세 가지 경우가 결합 부품과 인접한 모든 상황을 포함한다. 이제 세 가지 경우의 처리 방법을 기술한다.

① 경우 1: 이 경우에는 결합 부품의 중점과 f 점이 속한 접점의 중점을 두 끝점으로 하는 선분을 생성한다. 그림 4(a)에서 's', 'p', 'f', 'v', 'r'로 표시된 부품이 여기에 속한다. 그림 7에서는 이렇게 생성된 선분을 '1'로 표시한다.

② 경우 2: 이 경우는 두 개의 결합 부품의 중점을 두 끝점으로 하는 선분을 생성한다. 그림 6(a)의 'j'로

표시된 부품이 여기에 속한다. 그림 7에서는 이러한 선분을 '2'로 표시한다.

③ 경우 3: 이 경우는 결합 부품의 중점과 f 를 두 끝점으로 하는 선분을 생성한다. 그림 6(a)의 'g', 'e', 'h', 'n', 'u', 'd', 'l'로 표시된 부품이 여기에 속한다. 그림 7에서는 이러한 선분을 '3'으로 표시한다.

(2) 결합 부품과 인접하지 않은 부품

이 경우는 결합 부품으로부터 한 끝점을 제공받지 못하므로 두 개의 끝점을 찾아야 한다. 이를 위해 앞에서와 마찬가지로 부품의 외곽선상에 존재하는 굴곡점들을 대상으로, 그들간의 거리를 계산하여 가장 먼 두 개의 점을 선택한다. 이 두 점을 f_1, f_2 라 하면, f_1 과 f_2 가 접점에 속하는지의 여부에 따라 다음 세 가지 경우로 나눌 수 있다.

① 경우 4: $f_1 \in$ 접점, $f_2 \in$ 접점

이 경우는 두 점 모두 접점 상의 점이므로, 각 접점의 중점을 양 끝점으로 하는 선분을 생성한다. 그림 6(a)의 'm', 't'로 표시된 부품이 여기에 속한다. 이러한 선분은 그림 7에서 '4'로 표시하였다.

② 경우 5: $f_1 \in$ 접점, $f_2 \notin$ 접점 또는 $f_1 \notin$ 접점, $f_2 \in$ 접점
 접점의 중점과 접점에 속하지 않는 굴곡점(f_1 또는 f_2)을 끝점으로 하는 선분을 생성한다. 그림 6(a)의 'a', 'b', 'c'로 표시된 부품이 여기에 속한다. 이러한 선분은 그림 7에서 '5'로 표시하였다.

③ 경우 6: $f_1 \notin$ 접점, $f_2 \notin$ 접점

이 경우는 f_1 과 f_2 를 양 끝점으로 하는 선분을 생성한다. 그림 6(b)의 '췌'자의 부품 'a'가 여기에 속한다.

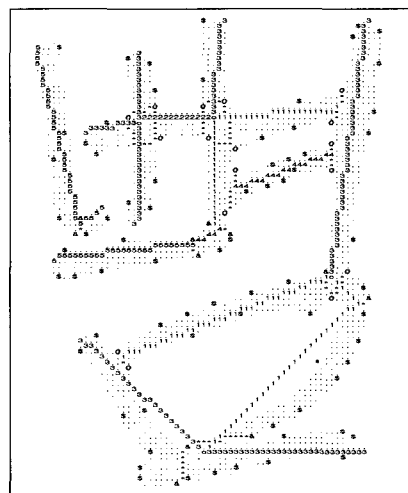
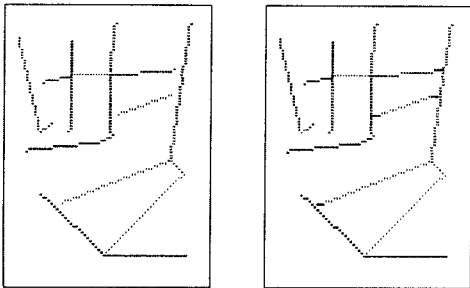


그림 7 선분 생성

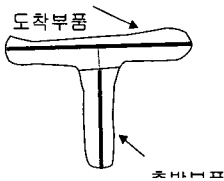
(3) 선분 연장

이제까지 모든 부품에 대해 그것을 대표하는 선분을 추출하였다. 그림 8(a)는 선분 추출 결과를 보여준다. 즉, 그림 7에서 '1'-'6'사이의 숫자로 표시된 선분만을 그린 것이다. 여기에서 한 가지 문제는 한글의 자소 중, 'l', 'r', 'o'와 같은 T-접점 영역에서 획의 연결성이 끊긴다는 것이다. 그림 8(a)에서 선분 's'와 'u', 't'와 'p', 't'와 'u', 'f'와 'd'가 선분의 끊김이 발생하는 곳이다.

추출된 선분들이 원래 패턴의 연결성을 그대로 유지하기 위해서는 선분 연장이 필요하다. 선분 연장을 수행하기 위해서, 연장 선분이 출발할 부품과 도착할 부품을 정해야 한다. 이를 위해 접점의 방향을 이용하였다. 접점에 인접한 두 개의 부품 중, 접점의 방향과 유사한 부품을 도착 부품, 나머지 부품을 출발 부품으로 정하였다(그림 8(c)). 출발 부품에 해당하는 선분의 끝점을 도착 부품에 해당하는 선분까지 연장하였다. 그림 8(b)에서 '&'로 표시된 부분이 선분 연장을 수행한 부분이다.



(a) 선분의 끊김 현상 (b) 선분 연장을 수행한 결과



(c) 출발 부품과 도착 부품

그림 8 선분 연장

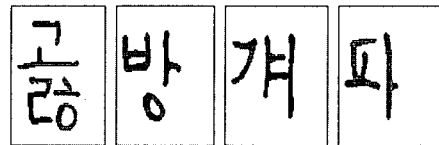
4. 골격선 비교 실험 및 분석

이 장에서는 기존의 세선화 방법과 본 논문에서 제안한 방법을 비교 분석한다. 추출된 골격선 비교를 위해서 다섯 가지 기준을 설정하였다. 실험 데이터로는 PE92

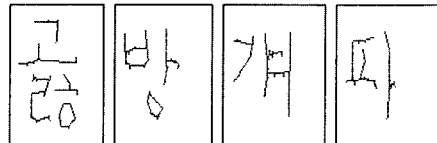
필기 한글 데이터베이스에 있는 패턴을 사용하였다.

4.1 세선화 기반 골격선 추출

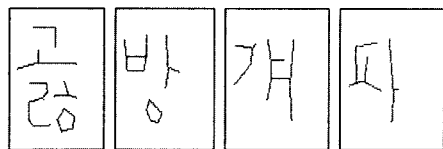
기존의 세선화-기반 골격선 추출 단계는 다음과 같다. 먼저, 이진화된 입력 패턴에 세선화 연산을 적용시킨다. 세선화된 패턴에서 끝점과 분기점의 특징점을 추출한다. 추출된 특징점들을 기준으로 하여 체인 코드 형태로 변환한 후, 이에 직선 근사(linear approximation) 연산을 적용하여 직선 선분(line segment) 집합으로 변환한다. 세선화 연산을 위해서는 Zhang-Suen [13]과 SPTA(Safe Point Thinning Algorithm) [14]을 사용한다. 이 두 방법은 국내 문자 인식 그룹에서 많이 사용하는 세선화 방법이다. 비교를 위해서 Zhang-Suen 방법은 [5]에서 사용한 프로그램 코드를 구하여 사용하였고, SPTA 방법은 본 연구팀이 이미 구현하여 사용하고 있는 방법이다. 그림 9은 원래 이미지와 Zhang-Suen 세선화 연산과 SPTA 세선화 연산으로부터 추출한 골격선의 예제이다.



(a) 원래 이미지



(b) Zhang-Suen 세선화 기반



(c) SPTA 세선화 기반

그림 9 세선화에 의한 골격선 예제

4.2 골격선 비교를 위한 기준

비교를 위해 이 논문이 설정한 다섯 가지 기준은 다음과 같다. 이들 기준은 추출된 골격선을 이용할 다음 단계, 즉 인식 단계, 자모 분할 단계, 또는 각종 특징 추

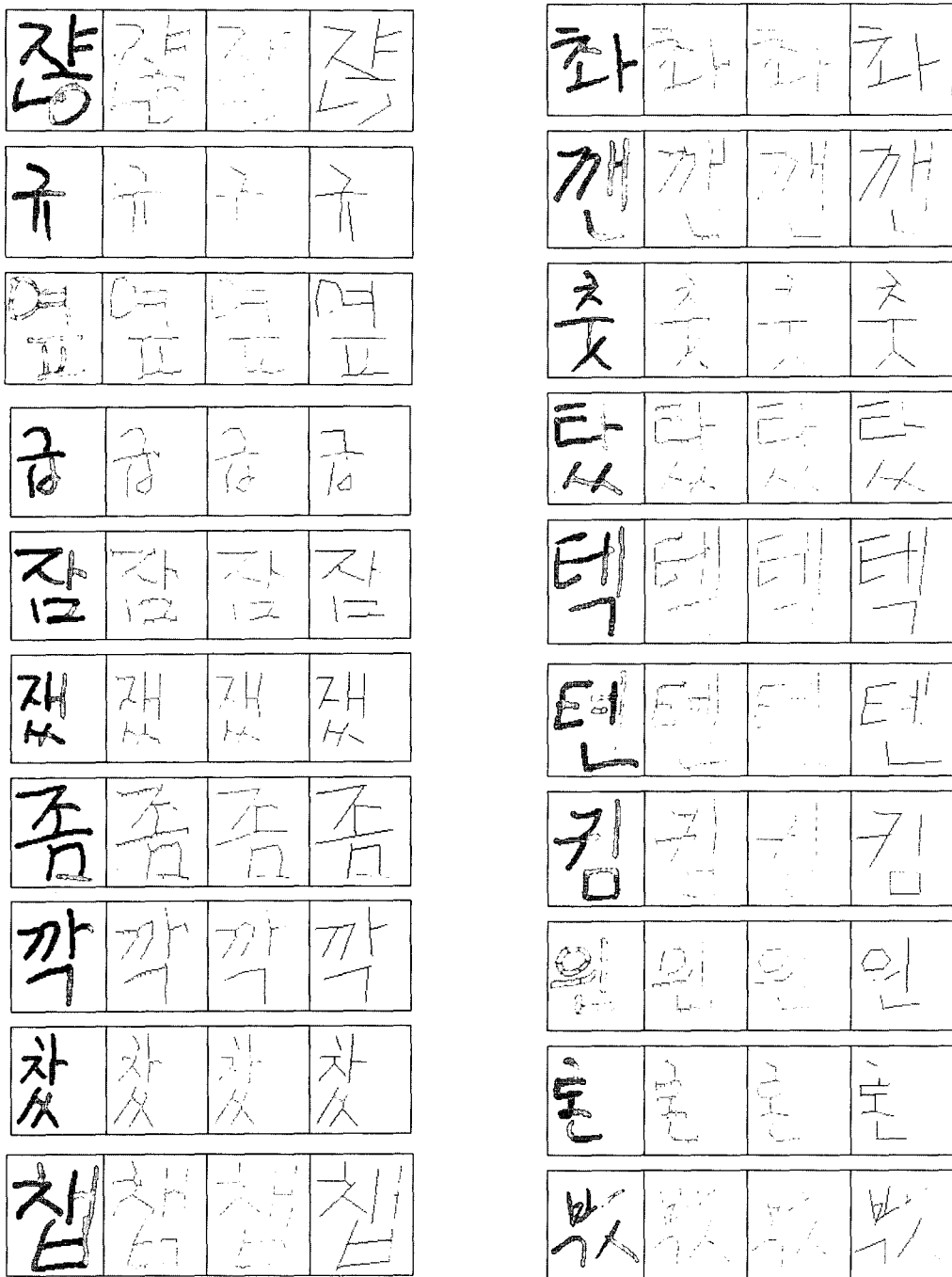


그림 10 세 가지 방법으로 추출된 골격선 (맨 왼쪽: 원래 패턴, 왼쪽에서 두 번째: Zhang-Suen 세션화, 왼쪽에서 세 번째: SPTA 세션화, 맨 오른쪽: 본 논문 방법)

출 단계의 성능과 밀접한 관련이 있다.

(1) 잡영 가지(noise branch) 발생: 필기체 데이터 획득시 잡음으로 인해 외곽선에 작은 돌출이나 홈이 자주 발생한다. 이 기준은 이러한 외부 잡영에 대해서 얼마나 둔감하게 작동하는지를 측정한다. 잡영 가지가 많은 경우 이들을 제거하는 전처리가 필요한데, 제거과정에서 실제로 유용한 획도 제거되는 가능성을 안고 있다.

(2) 획의 직선성(straightness) 유지: 한글의 모음 중 'ㄷ', 'ㄱ', 'ㄴ' 등과 같이 두 개의 획이 T-접점의 연결 형태로 접촉하는 경우가 많다. 이러한 경우, 두 개의 획이 접촉하는 부분에서 하나의 직선 획이 둘 또는 세 개의 획으로 분할되어 직선성이 유지되지 않는 경우가 발생할 수 있다. 이는 선분 개수를 증가시켜 인식시간이 길어지는 문제와 획의 방향이 많이 변하는 경우에는 오인식을 초래할 수 있다.

(3) 획의 길이 유지: 이 기준은 추출된 골격선 위에 원래 이미지를 겹쳐 놓고 보았을 때, 추출된 골격선의 길이가 원래 패턴 획의 길이와 얼마나 유사한 지를 측정한다. 원래 짧은 획의 경우는 골격선 추출 과정에서 길이가 짧아지면 'ㄷ'을 'ㄱ' 등으로 오인식하는 원인이 될 수 있다.

(4) 획교차(stroke crossing): 이 기준은 원래 패턴에서 획들이 서로 교차하는 지점을 골격선이 잘 표현하고 있는 지를 가늠할 수 있는 기준이다. 획이 교차하는 곳에서 패턴 왜곡이 가장 많이 발생한다. 또한 이 지점이 문자에 대한 정보를 많이 가지고 있기 때문에 제대로 처리하지 않으면 오인식을 일으킬 가능성이 높다.

(5) 복원성(reconstructability): 추출된 골격선을 구성하는 각각의 선분에 대해 원래 패턴의 어느 영역에 해당하는지를 명시적으로 알 수 있는지를 파악하기 위한 기준이다. 각 선분에 해당하는 패턴의 영상 영역을 알 수 있다면 이 선분에 대한 다양한 특징을 추출하여 인식등에 유용하게 사용할 수 있다.

4.3 비교 실험 결과

본 논문에서 비교 실험한 결과를 정량으로 제시하는 것은 쉽지 않다. 4.2절에서 설정해 놓은 기준은 인식등의 단계에 대한 성능과 밀접한 관련이 있는 실제적인 기준들이다. 이들 기준을 가지고 많은 패턴에 대해 비교 실험함으로써 객관성을 유지하려고 노력하였다. 비교 실험을 위해 사용된 데이터는 PE92 데이터를 사용하였다. 이 데이터는 대부분 난필로 쓰여졌으며 자소끼리의 접촉이 다양하다는 특징을 가지고 있다. 그래서 본 논문에서 비교 실험을 위해서 제시한 예제는 획들이 서로 교차하는 지점에서 세선화 연산시 왜곡이 발생할 수 있는

예제들을 중심으로 다루었다. 또한 'ㅇ'과 'ㅁ'과 같은 자소는 한글 인식에 중요한 정보를 제공하기 때문에 이러한 자소에 대한 예제도 다루었다. 그림 10은 원래 문자 패턴, Zhang-Suen 세선화 방법을 사용한 골격선, SPTA 세선화 방법을 사용한 골격선, 그리고 본 논문에서 제안한 방법을 이용한 골격선을 보여준다.

(1) 기준 1 : 잡영 가지 발생 그림에서 보듯이 Zhang-Suen 세선화 방법은 너무 많은 잡영가지를 생성한다. SPTA와 제안한 방법은 비교적 적은 잡영가지를 생성한다. SPTA 경우는 획 방향이 급격히 변하는 지점에서 필요없는 선분을 발생시키는 경우가 발생하였다. 대표적인 예로서 '잠'자의 'ㅈ'에서 필요없는 긴 획이 발생하였다. 이러한 잡영가지는 선분 개수를 증가시킴으로써 인식시간을 길게하는 요인이 될 수 있다. 또한 잡영을 제거하는 과정에서 실제 유용한 선분까지 제거하는 오류를 범할 가능성도 높다.

(2) 기준 2 : 획의 직선성 유지 그림 11은 그림 10의 문자 '좌'에서 모음 'ㅅ'만 따로 떼어놓은 것이다. 'ㄱ'의 수평 획과 'ㄷ'의 수직획이 Zhang-Suen 방법과 SPTA 방법으로 추출된 골격선에서는 세 개 혹은 그 이상의 선분으로 표현되어 원래 획이 가지고 있는 직선성을 잃어버렸다. 그러나 본 논문에서 제안한 방법으로 추출된 골격선에서는 원래 패턴의 직선성을 그대로 유지하고 있다. 이러한 현상은 그림 8에서 제시한 거의 모든 패턴에서 나타나므로 기준 2는 매우 중요하다고 할 수 있다. 'ㅅ'에 대해 Zhang-Suen 방법은 25개, SPTA는 8개, 그리고 제안한 방법은 4개의 직선을 생성하였다. Zhang-Suen과 SPTA 방법과 같이 필요 이상의 선분을 출력하면, 인식기에 많은 부담을 준다. 필기 패턴 간에 존재하는 변화가 더 심해져서 매칭 오류를 발생시킬 가능성이 높아지고, 또한 인식시간도 인식기에 따라 선형 또는 지수적으로 증가할 수 있다.

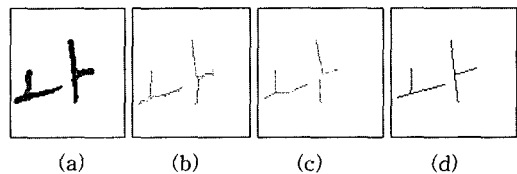


그림 11 획의 직선성 유지에 대한 예제 (a) 원래 이미지, (b) Zhang-Suen 방법, (c) SPTA 방법, (d) 본 논문의 방법

(3) 기준 3 : 획의 길이 유지 세선화 방법은 기본

적으로 획 외곽으로부터 화소를 제거하며 획의 중앙선 (medial line)을 찾아가는 연산을 수행하므로 획의 길이가 원래 입력 패턴이 가지고 있던 것보다 짧게 추출된다. 그러나 제안한 방법은 분해된 부품들의 외곽선상에 있는 점들 중에서 양 끝점을 찾아 선분을 계산하므로 원래 획 길이를 그대로 유지한다. 한글에서는 ‘ㄱ’과 ‘ㄷ’의 수평획등이 필기시 짧게 쓰여지는 경우가 많다. 이런 경우 골격선 추출과정에서 획 길이가 짧아진다면 이들 획이 잠영가지로 처리되어 오인식을 발생시킬 가능성이 있다. 그림 9에서 ‘파’자가 ‘피’자로 오인식될 수 있다.

(4) 기준 4 : 획 교차 영문이나 한문의 경우 문자 특성상 획 교차가 많이 발생하는데(X, t, 木, 七 등), 한글은 획간의 교차가 없다. 하지만 필기의 경우에는 그림 12에 보는 것과 같은 교차가 종종 발생한다. 이 경우 교차 부분에 문자에 대한 정보가 많이 포함되어 있으므로 추출된 골격선이 이 지점을 잘 표현할 수 있어야 한다. 그림 12는 그림 10의 문자 ‘컴’에서 ‘구’부분만 따로 떼어 놓은 것이다. 여기에서 수평 획은 하나의 획으로 구성되어 있다. 그러나 Zhang-Suen 방법과 SPTA 방법은 세 개 혹은 그 이상의 획으로 분리되었다. 특히 SPTA 에서는 교차 중앙 지점에서 짧은 획이 하나 발생하여 이 획이 인접한 네 개의 획을 이어주는 역할을 한다. 이러한 경우, 인식 과정에서 ‘컴’자를 ‘킴’자로 오인식하기 쉽다. 이러한 현상은 그림 10에서 ‘규’, ‘급’, ‘짚’, ‘찰’, ‘춧’, ‘닷’, ‘빱’자에서도 나타나고 있다. 제안한 방법에서는 획이 교차하는 지점에서 결합 부품을 탐지하고 이 결합 부품이 인접한 부품들이 공유하는 한 개의 점을 제공한다. 이 점을 중심으로 네 개의 선분이 연결되는 형태로 명확하게 출력되었다. 선분의 개수는 Zhang-Suen은 21개, SPTA는 8개, 제안한 방법은 5개이었다. 그림 10의 문자들 중에서 ‘찰’자의 ‘ㅉ’을 제외하고는 모든 경우 세선화에서 발생하는 문제를 해결하였다. 이는 인식기 측면에서 보면 큰 향상이라고 말할 수 있다.

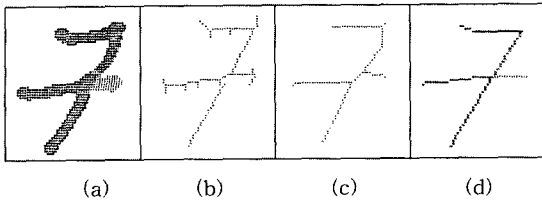


그림 12 획 교차에 대한 예제 (a) 원래 패턴, (b) Zhang-Suen 방법, (c) SPTA 방법, (d) 본 논문의 방법

기준 5 : 복원성 골격선은 원래 패턴 영상이 가지고 있는 정보를 손실한다. 그러나 제안한 방법은 패턴을 모양 분해하고, 분해된 부품으로부터 골격선을 추출하기 때문에 필요하면 언제든지 원 영상내에서 부품 영역을 참조할 수 있다. 그림 13은 그림 10의 ‘컴’자와 ‘원’자의 ‘ㅁ’과 ‘ㅇ’의 모양 분해된 부품과 여기에서 추출한 선분들을 보여준다. 각 부품을 알파벳으로 구별하고 여기에서 추출된 선분도 같은 기호로 표시하였다. 이 그림에서 볼 수 있듯 골격선을 구성하는 각 선분은 해당 부품을 명시적으로 알고 있다. 때문에 필요한 경우에는 해당 부품 영역을 참조하여 다양한 특징을 추출할 수 있다. 특히 ‘ㅁ’과 ‘ㅇ’을 구별하는데 유용한 획의 굴곡성을 추출하는데 사용할 수 있다. 그림 13에서 ‘ㅁ’의 경우에는 각 부품에 굴곡성이 없는 반면 ‘ㅇ’의 경우에는 부품들이 굴곡성을 갖고 있고 이들이 일관성있게 연결되어 있으므로, 이러한 굴곡성 특징을 사용하여 ‘ㅁ’과 ‘ㅇ’을 효과적으로 구별할 수 있다. 세선화의 경우에는 각각의 선분이 차지하는 원래 패턴에서의 영역을 명시적으로 알 수 없으므로 이러한 특징 추출이 쉽지 않다. 제안한 방법의 이러한 성질은 인식 단계에 매우 유용하게 사용할 수 있다.

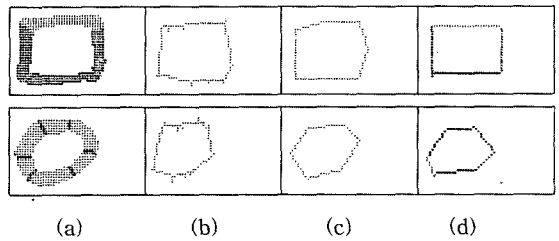


그림 13 복원성에 대한 예제 (a) 원래 패턴, (b) Zhang-Suen방법, (c) SPTA 방법, (d) 본 논문의 방법

표 1 골격선을 구성하는 평균 선분 개수

	Zhang-Suen 방법	SPTA 방법	본 논문의 방법
선분 개수	68개	17.15개	10.75개

표 1은 골격선을 구성하는 선분의 개수에 대한 통계이다. 제안한 방법은 문자 패턴당 평균 10.75개를 생성한 반면, SPTA는 약 17.15개, Zhang-Suen은 68개의

선분을 생성하였다. 앞에서 언급했듯이 쓸데없이 많은 선분은 인식기의 속도를 저하시키는 요인이 되며 또한 인식 오류를 발생시키는 요인도 된다.

5. 결론 및 향후 연구 과제

본 논문에서는 모양 분해 알고리즘을 이용한 필기 한글 문자의 골격선 추출 방법을 제안하였다. 대부분의 필기체 문자 처리 과정 중 전처리로서 사용되는 세선화 기반 골격선 추출 방법과 본 논문에서 제안한 골격선 추출 방법을 잡영 가지의 발생, 획의 직선성 유지, 획의 길이 유지, 획교차, 복원성과 같이 다섯 가지의 기준을 설정하여 비교하였다. 비교 실험 결과, 본 논문에서 제안한 골격선 추출 방법이 세선화 기반 골격선 추출 방법보다 우수함을 보였다. 따라서 본 논문에서 제안한 방법이 전처리로서 사용될 경우, 인식률의 향상을 보일 것으로 기대된다. 그러나 본 논문의 방법은 모양 분해와 골격선 추출 단계를 거치기 때문에 수행 시간 면에서는 세선화 방법보다 훨씬 떨어지는 단점이 있다.

향후 연구 과제로는, 추출한 골격선을 이용하여 인식 방법을 개발하는 것이다. 또는 기존 골격선을 이용한 인식기에서 골격선 추출모듈만을 본 논문의 모듈로 대체하여 인식률 향상을 측정해 볼 수 있다.

참고 문헌

- [1] 이성환, 박희선, "한글 인식의 사례 연구 : 최근 5년 동안의 연구 결과를 중심으로", 제1회 문자인식 워크샵 발표 논문집, 충북대학교, pp.3-46, 1993.
- [2] 한선화, 이충식, 이준호, 김진형, "문자 인식 기술을 이용한 데이터베이스 구축", 정보처리 논문지, Vol.6, No.7, pp.1713-1723, 1999.
- [3] 김수형, 정선화, 오일석, "필기 한글 문자의 오프라인 인식에 관한 연구", 한국정보과학회 98 가을 학술발표논문집(II), 아주대학교, pp.396-398, 1998.
- [4] K. Lin, Y.S. Huang, and C.Y. Suen, "Identification of fork points on the skeletons of handwritten Chinese characters," *IEEE Transactions on PAMI*, Vol.21, No.10, pp.1095-1100, 1999.
- [5] 김호연, 서장원, 권재욱, 김진형, "필기 한글 인식에 관한 연구:KAIST의 연구를 중심으로", *인지과학회*, Vol.6, No.4, pp.5-20, 1995.
- [6] H. J. Kim, P. K. Kim, "Recognition of off-line handwritten Korean characters," *Pattern Recognition*, Vol.29, No.2, pp.245-254, 1996.
- [7] 이은주, 권오석, 김태균, "필기체 한글에서 자모 분리 와 인식", *한국정보과학회 논문지*, Vol.15, No.6, pp.526-534, 1988.
- [8] 김태균 et.al, "Stroke 조합에 의한 필기체 한글의 표현 과 인식", *전자공학회논문지*, Vol.25, No.1, pp.18-26, 1988.
- [9] 황영섭, "제약조건 만족 그래프를 이용한 필기 한글 인식", *포항 공과 대학교 박사 학위 논문*, 1997.
- [10] 한동협, 조성배, "오프라인 필기 한글 영상의 세선화를 위한 지식기반 시스템에 관한 연구", 제10회 영상처리 및 이해에 관한 워크샵, pp.163-168.
- [11] J. W. Suh, J. H. Kim, "Stroke extraction from gray-scale character image," *IWFHR'96*, pp.593-598, 1996.
- [12] 박정선, 오일석, "필기 한글 문자의 모양 분해", *한국정보과학회 논문지*, (submitted).
- [13] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital pattern," *CACM*, Vol.27, No.3, pp.236-239, 1984.
- [14] N. J. Naccache and R. Shinghal, "SPTA : a proposed algorithm for thinning binary patterns," *IEEE Transactions on System, Man, and Cybernetics*, Vol.SMC-14, No.3, pp.409-418, May/June 1984.
- [15] 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, p.859-872, 1989.
- [16] 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.
- [17] Il-Seok Oh, Jin-Seon Lee, Tae-Woong Yoo, Jung-Seon Park, Ki-Chun Hong, "Region-based shape decomposition algorithm for Hangul character patterns," *Proceedings of 6th IWFHR*, Vol.6, pp.557-566, Taejeon, Korea, 1999.
- [18] W. S. Kim, R. H. Park, "Off-line recognition of handwritten Korean and alphanumeric characters using hidden Markov models," *Pattern Recognition*, Vol.29, No.5, pp.845-858, 1996.



홍 기 천

1995년 전북대학교 전산통계학과 학사.
1997년 전북대학교 전산통계학과 석사.
2000년 현재 전북대학교 전산통계학과 박사과정. 관심분야는 패턴 인식, 컴퓨터 비전

오 일 석

정보과학회논문지 : 소프트웨어 및 응용 제 27 권 제 5 호 참조