

필기체 인식을 위한 한글 자소분리

A Hangul Element Separation for the Hand-written Character Recognition

백 남 우

(제천기능대학, 조교수)

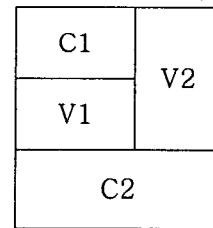
본 연구는 필기체 한글 문자를 인식하기 위하여 한글 문자구조를 6개 기본구조로 분류한다. 각각의 한글 자음과 모음을 7-세크먼트, ' / ' (Left-Incline), ' \ ' (Right-Incline), ' - ' (Left-Right), ' | ' (Up-Down), ' C ' (Circle), ' ㄱ ' (Righr-down), ' ㄴ ' (Down-Right) 분리한다. 분리된 7-세크먼트에 대해 한글이 쓰여지는 위치에 따라 8개의 기본구조로 정의하여 세크먼트를 분리하여 레벨화한다. 따라서 본 연구는 문자를 자소(자음과모음)로 하여 7-세크먼트로 분리하는 필기체 자소분리 구조를 제시한다.

목 차

- I. 서론
- II. 입력문자 구조
- III. 세크먼트정의와 추출
 - 3.1 세크먼트 정의
 - 3.2 세크먼트의 분리 및 추출
 - 3.3 분류문자
- IV. 자소도출 및 합성
- V. 결론

I. 서론

한글 문자의 기본 자소는 주로 직선으로 구성되어 있고, 그 위에 직선 성분이 추가되어 복잡한 자소로 변환해 가는 필기체 한글 문자 인식에 대한 연구가 활발히 진행되고 있지만 그 인식 수법이 아직 확립되어 있지는 않다. 본 논문에서는 한글의 특징이 기본적으로 직선과 동그라미로 구성되어 있는 점을 활용하여 한글을 6개의 문자구조로 분류하고, 그 문자구조를 7종류 세크먼트(/, \, |, -, O, ㄱ, ㄴ)로 정의하여, 쓰여지는 문자로부터 segment를 추출한다.



<그림 1> 기본구조

II. 입력문자 구조

읽어들인 문자Data는 640×378 dot로 Digital화된다. 이 문자 정보에 대하여 5×5의 25개점을 1개의 mass로 하여 점 Data가 13개 이상이면 이 mass전체를 1로, 12개 이하이면 0으로 변환한다. 그리고 문자가 항상 중심에 오도록 중심 보정을 행한다. 그리고 다시 이 정보에 대하여 4×4로 Data를 압축하면 82×160문자 Data에 대한 선의 폭을 '1'로 하기 위한 세션화 algorithm을 적용한다. 그림1의 기본구조에서 C1은 초성 자음, V1은 중성 횡모음, V2는 중성 종모음, C2는 중성 자음을 각각 나타낸다.

6형의 문자구조는

- 제 1 형 : C1 + V1
 - 제 2 형 : C1 + V1 + C2
 - 제 3 형 : C1 + V1
 - 제 4 형 : C1 + V1 + C2
 - 제 5 형 : C1 + V1 + V2
 - 제 6 형 : C1 + V1 + V2 + C2
- 로 분류한다.

III. Segment 정의와 분리

1. Segment 정의

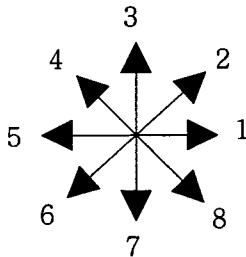
본 연구에 있어서 segment는 필기체 한글을 써 내려가는데 있어서 일 필로 쓰기 시작하여 떨어트리지 않고 써내려갈 수 있는 정보에 착안하여 표1과 같이 정의한다.

<표 1> Segment의 정의

순번	Segment	Label	의미
1	/	LI	Left-Incline
2	\	RI	Right-Incline
3	—	LR	Left-Right
4		UD	Up-Down
5	o	C	Circle
6	ㄱ	RD	Right-Down
7	ㄴ	DR	Down-Right

2. Segment의 분리 및 추출

segment 추출은 양자화 된 8개의 방향성분을 이용하여 입력문자에 대하여 탐색을 통하여 분리 추출한다. 여기에서 방향성분은 그림 2에 보이는 바와 같이 번호로 나타낸다.



<그림 2> 방향성분

3x3 마스크를 이용하여 같은 segment에 대하여 방향성분을 구하는 것에 의하여 segment를 추출한다. 각각의 segment에 대한 rule은 표2이고 자소의 레벨화는 표3과 같다.

<표 2> Label의 Rule

Label	방향성분 및 각도(θ)
LI	6 이 10개 이상존재 & 210°-240°
RI	8 이 10개 이상존재 & 300°-330°
LR	1 이 10개 이상존재 & 350°-20°
UD	7 이 10개 이상존재 & 240°-300°
C	6,7,8,2,3 이 각각 3개 이상 존재 & 360°
RD	1 이 10개 이상존재 & 350°-20°
	7 이 10개 이상존재 & 240°-300°
DR	1 이 10개 이상존재 & 350°-20°
	6 이 10개 이상존재 & 210°-240°
	7 이 10개 이상존재 & 240°-300°
DR	7 이 10개 이상존재 & 240°-300°
	1 이 10개 이상존재 & 350°-20°
	7 이 10개 이상존재 & 240°-300°
DR	7 이 10개 이상존재 & 240°-300°
	2 가 10개 이상존재 & 30°-60°

그림 2는 본 연구에서 segment를 추출하기 위한 8 방향성

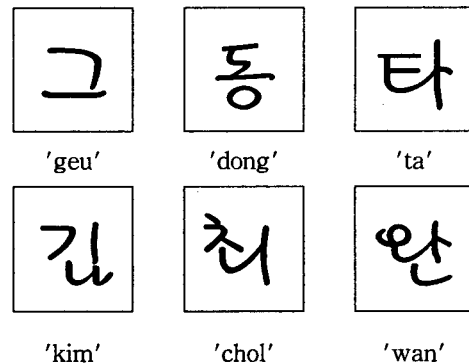
분을 나타낸다. 각도는 중앙을 중심으로 하여 왼쪽에서 오른쪽으로, 위에서 아래 방향으로 출발하여 각 세크먼트에 이르는 각도로 반시계 방향으로 한다. segment C의 경우에는 시점좌표가 같은 경우 그리고 방향성분의 6, 7, 8, 2, 3 이 사용된 정보를 이용하여 추출하고 각도는 360° 로 정의한다.

<표 3> 자소의 label화

자음자	모음자
ㄱ ; G	ㅏ ; A
ㄴ ; N	ㅑ ; YA
ㄷ ; D	ㅓ ; EO
ㄹ ; L	ㅕ ; YEO
ㅁ ; M	ㅗ ; O
ㅂ ; B	ㅛ ; YO
ㅅ ; S	ㅜ ; U
ㅇ ; NG	ㅠ ; YU
ㅈ ; J	ㅡ ; EU
ㅊ ; C	ㅣ ; I
ㅋ ; K	
ㅌ ; T	
ㅍ ; P	
ㅎ ; H	

3. 분류문자

본 연구에서 분류 문자의 예를 표시함.



<그림 3> 실험용 문자표시

IV. 자소 도출 및 합성

자소는 추출된 7개의 segment 속성과 필순을 이용하여 미리 작성된 지식에 의하여 결정한다. 각 방향으로부터 발생 가능한 자소간의 관계를 표4에 보인다. 이 표에 있어서 사각안에 있는 숫자는 방향성분이며, 필순에 따른 segment가 나타나는 순서번호를 나타낸다. 여기에서 1에는 방향성분 1에서 시작하여 생성되는 자소인 'ㄱ', 'ㅌ', 'ㅍ', 'ㄷ', 'ㅈ', 'ㄹ', 'ㅊ', 'ㅎ', 'ㅡ', 'ㅜ', 'ㅣ', 'ㅋ', 'ㅠ'을 보이고 있고, (b)는 방향성분 7에서 시작하여 생성되는 자소인 'ㄴ', 'ㅂ', 'ㅁ', 'ㅅ', 'ㅎ', 'ㅣ', 'ㅏ', 'ㅑ', 'ㅓ', 'ㅕ'를 보이고 있으며, (c)는 방향성분 6에서 시작되는 자소인 'ㅈ', 'ㅇ'을 보이고 있다. 그리고 (d)는 방향성

분 8에서 시작하는 자소 ‘ㄱ’, ‘ㅋ’, ‘ㆁ’, ‘ㅎ’을 보이고 있으며 (e)는 방향성분이 2부터 시작되어 생성되는 자소 ‘ㄷ’, ‘ㄱ’, ‘ㄴ’을 각각 보이고 있다. (d)와(e)는 비스듬하게 경사진 segment까지 고려하여 나타낸 관계이다.

<표 4> 각 방향 성분으로 구성되는 자소

	자음	모음	
a)	ㄱ	1,7 / 1,6	— 1
	ㄷ	1,7,1 / 1,7,2	ㅌ 1,7
	ㄹ	1,7,1,7,1 / 1,6,1,7,2	ㄴ 1,7
	ㅈ	1,6,8	ㅊ 1,1,7 / 1,2,7 / 1,8,7
	ㅊ	1,1,6,8	ㅌ 1,7,7 / 1,6,7 / 1,7,8
	ㅌ	1,1,7,1 / 1,1,7,2 / 1,7,1,2	
	ㅍ	1,7,7,1	
	ㅎ	1,1,6,7,8,1,2,3,4,5	
b)	ㄴ	7,1 / 7,2	ㄷ 7
	ㄷ	7,1,7,1 / 7,1,1,7	ㅌ 7,1
	ㅌ	7,7,1,1 / 7,1,7,1	ㅊ 7,1,1 / 7,1,2 / 7,1,8
	ㅊ	7,1,6,8	ㅌ 7,1
	ㅎ	7,1,6,7,8,1,2,3,4,5	ㅊ 7,7,1 / 7,6,1 / 7,8,1
c)	ㅅ	6,8	
	ㅇ	6,7,8,1,2,3,4,5	
d)	ㄱ	8,6	
	ㅋ	8,1,6	
	ㆁ	8,6,1	
	ㅎ	8,1,6,7,8,1,2,3,4,5	
e)	ㄱ	2,7	
	ㄷ	2,7,2 / 2,8,2	
	ㅌ	2,7,2,2	

segment가 쓰여지는 순서는 일반적으로 왼쪽에서 오른쪽으로, 위에서 아래로 쓰여진다. 그리고 1개의 자소는 1개부터 4개까지의 segment를 가지고 있으므로 자소임을 확인하기 위해서는 적어도 4개 이하의 segment가 필요 한 것을 알 수 있다. 그러므로 자소를 판별·도출하기 위해서는 구해진 segment의 처음 4개가 필요하다. 이 4개의 segment를 모두 사용하여 1개의 자소로서 성립하는가를 판단하여 성립하면 1개의 자소로서 확정시킨다. 1개의 자소로서 성립하지 않으면 3개, 2개, 1개로 한 개씩 segment를 감소시켜 자소가 만들어 질 때까지 반복한다. 1개의 자소가 완성되면, 다음 4개의 segment를 가져와서 같은 routine을 반복하여 구해진 segment를 모두다 사용했을 때 멈춘다. 표5에 7개의 segment에 의한 각 자소에 대한 정보를 보인다. 예를 들면, 자소 ‘ㄹ’의 경우 쓰여지는 순서에 따라서 첫 번째 획이 RD, 두 번째 획이 LR, 그리고 세 번째 획이 DR로 구성되어 있는 것을 의미한다. ‘ㄱ’, ‘ㄷ’, ‘ㅌ’, ‘ㅎ’에 대하여는 각각 잘 쓰여지는 패턴을 고려하여 두 가지의 segment 정보를 표현하고 있다.

<표 5> 자소간의 Segment 정보

자음자	모음자
ㄱ ; RD	ㅏ ; UD+LR
ㄴ ; DR	ㅑ ; UD+LR+LR
ㄷ ; LR+DR	ㅓ ; LR+UD
ㄹ ; RD+LR+DR	ㅕ ; LR+LR+UD
ㅁ ; ① UD+RD+LR ② DR+RD	ㅗ ; UD+LR
ㅂ ; ① UD+UD+LR+LR ② UD+LR+UD+LR	ㅛ ; UD+UD+LR
ㅅ ; LI+RI	ㅜ ; LR+UD
ㅇ ; C	ㅠ ; LR+UD+UD
ㅈ ; RD+RI	ㅡ ; LR
ㅊ ; (UD,LR,RI)+RD+RI	ㅣ ; UD
ㅋ ; RD+LR	
ㅌ ; ① LR+LR+DR ② LR+DR+LR	
ㅍ ; LR+UD+UD+LR	
ㅎ ; (LR,UD,RI)+LR+C	

한편 자소를 도출 할 때 표4의 정보에 의하면 모음자 ‘ㅏ’와 ‘ㅑ’는 같은 정보로 판단하기가 어려워진다. 그러므로 segment 순서 정보에 지식정보를 이용함으로써 알맞은 자소를 도출하도록 한다. segment로부터 자소가 도출되면 이 자소 정보로부터 문자를 합성한다. 한글문자는 1개의 문자로서 성립되기 위해서는 반드시 자음자와 모음자 또는 자음자, 모음자, 자음자 순서로 구성되며, 그 배치 위치는 규칙성이 있는 특징이 있다. 이러한 규칙성에 따라서 생성되는 문자의 구조는 6개의 구조로 표현 가능하다. 따라서 문자를 합성해 나갈 때 먼저 자음자인가를 판단하고 형 모음자 또는 종 모음자인가를 판단하여 1개의 문자로 생성 될 때까지 합성해 나간다.

문자가 합성된 후 결과를 label화하여 출력하도록 한다. 그림 4 에는 한글 자소 합성알고리즘에 의한 플로 차트로서 ① 1st segment를 load하여 constant인가를 판단하여 Y이면 buffer1에 저장하고 N이면 다음 segment를 load하여 초성을 buffer1에 저장한다. ② next segment를 load하여 vowel인가를 판단하여 vowel이면 buffer2에 저장하고 N이면 다음 segment를 load하여 buffer2에 저장한다. 이때 \$(완성된 글자일 때 종료하라는 문자임) 인가를 확인하여 Y이면 total buffer에 모두 저장하여 출력한다. ③ \$가 아니면 next segment가 load하여 constant인가를 판단하여 Y이면 buffer3에 저장한다. 이때 \$ 인가를 확인하여 Y이면 total buffer에 모두 저장하여 출력한다. ④ \$가 아니면 next segment가 load하여 vowel인가를 판단하여 Y이면 buffer4에 저장한다. 이때 ⑤ \$ 인가를 확인하여 Y이면 total buffer에 모두 저장하고 N이면 constant로 다시가 종성을 저장하여 출력하고 끝낸다.

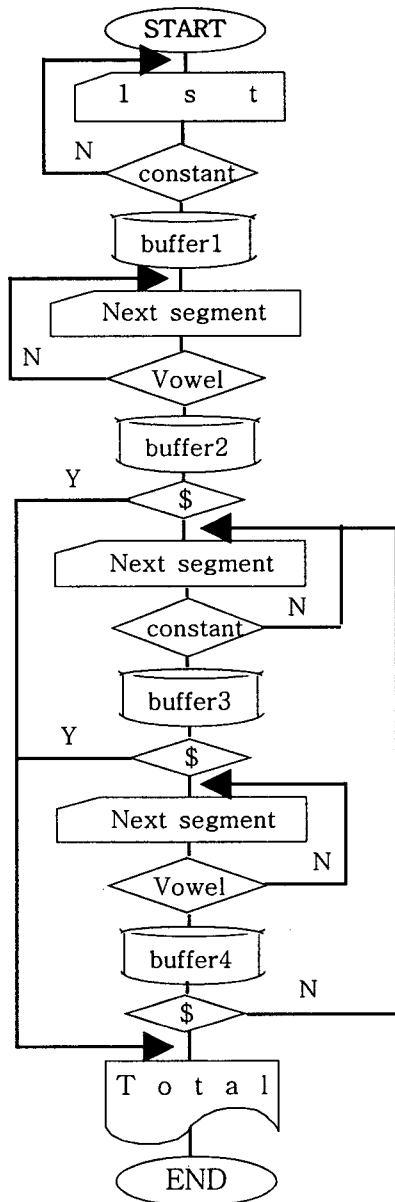
V. 결론

본 연구에서는 필기체 한글에 대하여 인식을 위한 자소분리 algorithm과 합성하는 algorithm을 제안하였다. 한글 자소에 대한 쓰여지는 순서정보를 이용하여 자소를 도출하고 도출된

자소를 이용하여 1개의 문자를 합성시켰다. 본 연구는 필기체 한글 인식을 위해 유효한 algorithm으로 고려된다. 향후 연구 계획으로는 좀더 복잡하게 자소가 붙어 있는 경우에 있어서 분리·합성시키는 방법, On-line 필기체 한글 인식 시스템 작성 등이다.

참고문헌

- ① 송정영, 문자구조 정보에 의한 필기체 한글 인식, 일본 퍼지학회, Vol. 10, No. 3, P. 506-512, 1998



<그림 4>. 한글 자소 합성알고리즘