

제스처 인식기를 포함한 통합된 온라인 한글인식기의 구현

정우식¹, 권영빈¹

¹중앙대학교 컴퓨터 공학과

Implementation of integrated On-Line Hangeul recognition system including Gesture recognition system

Woo-Sik Jung¹, Young-Bin Kwon¹

¹Dept. of Computer Science Engineering, ChungAng University

요 약

컴퓨터 발전되면서 컴퓨터는 소형화 되어져 왔다. 컴퓨터의 소형화란 사람들이 들고 다니면서 어디서든지 쉽게 사용할 수 휴대성이 만족되어야 한다. 휴대성을 만족하기 위해서 입력장치의 간편화가 요구되는데 가장 널리 알려진 키보드는 너무 크기 때문에 휴대성 만족하기에는 많은 문제를 안고 있다. 이러한 문제를 해결하기 위해서 등장한 것이 전자펜이다. 전자펜은 크기도 작고 사용법도 사람들이 많이 사용하는 펜과 비슷하기 때문에 배우기 쉽다는 장점이 있다. 그러므로 전자펜을 사용하기 위해서는 전자펜을 사용했을 때 컴퓨터가 사람인 쓴 문자나 제스처를 인식할 수 있는 기술이 필요하다. 본 논문에서는 온라인 문자인식기술에 대해서 간략히 설명을 한 뒤, 입력한 글자를 편집할 수 있는 제스처 인식기술에 대해서 설명할 것이다.

1. 서 론

컴퓨터 개발목적은 사람을 닮은 컴퓨터를 만드는 것으로 볼 수 있다. 그러므로, 사람의 능력을 컴퓨터에게 부여하는 연구는 많이 이루어져 왔다. 그 중 하나의 일환으로써 사람이 쓴 글을 컴퓨터가 인식하도록 하기 위한 연구도 함께 해왔다. 이러한 연구는 컴퓨터의 주변기기의 기술과 함께 발전하면서 입력장치를 키보드에서 펜으로, 컴퓨터를 대형화에서 소형화 휴대성을 만족하도록 발전시키고 있다[6][7].

컴퓨터가 소형화 되면서 PDA가 등장하게 되었다. PDA는 크기가 아주 작기 때문에 입력장치를 펜으로 대체하게 되었다. 이러한 펜을 이용하여 사람들은 글을 쓰고 쓴 글을 편집하여야 한다. 입력한 글의 편집을 손쉽게 하기위해서 필요한 기술이 바로 제스처 인식기술이다. 이러한 연구는 PDA의 사용을 보다 쉽고 보다 편하게 사용할 수 있도록 도와줄 것이다.

본 논문은 2장에서 기존에 구현된 한글인식기에 대해서 간략히 설명하고, 3장에서 제스처 인식기와 통합된 한글인식기에 대해서 설명한 뒤 결론을 맺는다.

2. 한글 인식기의 구현

2.1. 입력된 문자의 스트링 생성방법

입력된 문자는 인식에 영향을 미치는 필기시의 떨림을 최소화 하기 위하여 전처리 과정을 거친 후에 등간격의 점열을 생성한다. 전처

리 과정은 정규화(noramlization), 평활화(smoothing), 공간필터링(spatial filtering)의 과정을 통해서 이루어 진다. 전처리된 글자는 일반선분, 모서리 선분, 가상선으로 선분화가 된다. 다음 선분의 벡터 값이 임계값이상으로 변하였을 때의 선분을 모서리 선분으로 나타내며, 획과 획사이의 가상선에서 생성되는 선분은 가상선분으로 나타내며 마지막으로 모서리 선분도 아니고 가상선분도 아닌 경우 일반성분으로서 표현한다.

앞에서 정의한 3가지 방식으로 나뉘어진 선분들은 자신의 고유한 벡터값을 가지게 된다. 각각의 벡터값들은 0에서 31까지의 값을 가지게 된다. 이렇게 유형과 벡터값으로 분류된 선분들은 입력된 문자를 컴퓨터가 이해할 수 알파벳이 된다. 그러므로 모든 입력된 글자들은 일반선분, 모서리 선분, 가상선분의 연속된 스트링으로 표현되어 진다 [2].

2.2. 입력된 문자의 스트링 정합방법

스트링 정합방법에는 총 6개의 연산으로 이루어져 있다. 글자를 전체적인 편집비용을 계산하는데 필요한 3개의 기본연산과 자소를 분할하는데 필요한 3개의 자소분할 연산으로 나눌수 있다. 먼저 기본연산에는 교환, 삽입-a, 삭제 연산이 있다. 교환연산은 서로 같은 유형(Type)일 경우 적용되는 연산이며 삽입-a의 경우는 현재 시점에서 입력된 문자의 스트링의 유형이 연속적으로 같은 유형을 나타내고 원형자소의 경우는 연속적으로 다른 유형을 나타내는 경우에 적용된다. 마지막으로 삭제연산은 삽입-a의 연산과 반대되는 경우에 적용되는 연산이다.

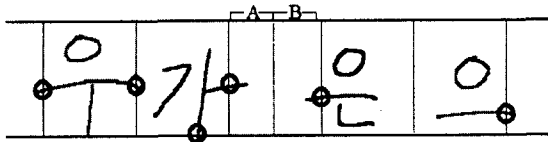
다음으로 자소분할에 필요한 연산에는 삼입-b, 삼입-c, 삭제-b 연산으로 나뉘어 진다. 먼저 삼입-b연산은 입력된 글자에 대해서 기본적으로 자소를 분할하는데 적용되는 연산이다. 삭제-b 연산은 스트리밍 생성에서 자소의 사이에 연속적으로 굴곡성분이 발생하여 삼입-b연산으로 자소간을 분할 할 수 없는 경우에 적용되는 연산이다. 마지막으로 삭제-b연산은 모음 'ㄱ'과 연결해서 쓸 수 있는 'ㄱ', 'ㄷ', 'ㅇ' 등과 같은 자음과 연결해서 쓴 경우 모음 'ㅏ'와 자음을 분할해 주는 데 적용되는 연산이다[1].

3. 제스처 인식기의 구현

이 부분에서는 제스처 인식을 적용하기 위하여 원고지 형태에서 여러글자를 인식하는 간략한 방법에 대해서 설명한 뒤, 제스처 인식에 대한 방법을 설명할 것이다.

3.1. 연속된 글자 인식

한국말은 영어와 달리 2차원적인 구조를 가지고 있기 때문에 음절과 음절을 구분하는 것이 어려운 것으로 알려져 있다. 그러므로 여기서는 원고지 형식을 이용하였다. 선상에 글자와 글자를 구분하는 선이 있는 방식으로서 글자의 입력시 다른 영역으로 넘어가는 경우를 검사 할 수 있도록 하였다. 다음 [그림-1]은 글자를 입력할 때, 영역을 넘어가는 경우를 나타내고 있다.



[그림-1] 글자 입력시 영역을 넘어간 경우

위의 그림에서 동그라미로 표시한 것들이 글자의 입력시 영역을 넘어간 경우를 나타내고 있다. 이러한 것을 처리하기 위하여 각 영역들은 자신의 고유한 번호를 가지고 있으며, 각각의 영역은 위의 그림에서 처럼 A part와 B part로 구분되어져 있다. 먼저 가로의 영역이 넘어간 경우는 선분이 시작한 위치로서 그 선분에 대한 글자를 분할하고 세로의 경우는 4가지로 구분하여 글자를 분할한다.

- S : i-1B : E : iB ⇒ i ----- (식-1)
- S : iB : E : i+1B ⇒ i ----- (식-2)
- S : iA : E : i+1A ⇒ i ----- (식-3)
- S : i-1B : E : i+1A ⇒ i ----- (식-3)

where i : 영역 번호

A, B : 영역의 A part와 B part

S, E : 선분의 시작점과 끝점

위의 식은 선분의 시작과 끝의 위치에 따라서 선분이 포함되어져야 할 영역의 위치를 나타내는 것이다. 예를 들어 (식-1)의 경우는 위의 그림에서 '은'자에 해당되는 경우이며, (식-2)는 '가' (식-3)은 '으' 그리고 마지막으로 (식-4)는 '우'의 경우에 해당된다.

3.2. 제스처의 정의

제스처의 인식을 위하여 인식할 제스처를 정의하여야 한다. 그러기 위해서 원고지에서 교정을 위하여 사용하는 교정기호를 중심으로 제스처를 정의하였다. 원고지 교정기호는 총 22개로 정의되어져 있다. 그러나 온라인특성을 고려하여 이 중에서 14개로 줄여 제스처를 정의한다. [표-1]은 정의한 제스처를 나타내고 있다[3][4].

[표-1] 정의되어진 제스처

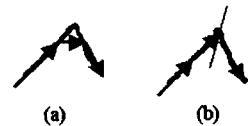
교정기호	쓰임	교정기호	쓰임
ㄱ	띄어 쓸때	ㄷ	붙여쓸때
ㄴ	다음줄로 바꿀때	ㅇ	글자를 없앨때
≡	내용을 지울때	ㄷ	순서를 바꿀때
ㄷ	앞 줄에 이올때	> <	줄을 띄울 때
ㄷ	줄을 붙일 때	∧	문장부호를 넣을때
ㄷ	오른쪽으로 옮길때	ㅏ	왼쪽으로 옮길때
ㄷ	위로 옮길때	ㅓ	아래로 옮길때

3.3. 제스처의 인식

3.3.1. 제스처의 단위벡터 생성

제스처의 전처리에는 다른 온라인 인식과 같이 평활화, 공간필터링을 거친다. 그러나 제스처는 그 가로와 세로에 대한 비율이 다양하여 정규화(normalization)을 할 수 없다. 정규화를 하게 되면 모양이 심하게 변하게 되어 인식을 더 어렵게 하기 때문이다.

정규화를 하지 못하게 됨으로써 크기에 따라서 생성되는 정보의 양이 크게 달라진다. 그렇기 때문에 먼저 단위벡터를 생성하여 같은 특징을 같은 단위벡터를 모아서 프리미티브(Primitive) 성분을 추출한다. 여기서 단위 벡터를 추출하는데 있어서 입력된 모습과 달리 추출되는 경우가 발생한다.



[그림-2] 단위벡터의 추출

위의 그림에서 (a)은 입력된 제스처에서 단위 벡터를 잘 추출하지 못한 경우이고, (b)의 경우는 굴곡점(Bending point)을 추출하여 단위벡터를 입력된 제스처의 모양과 가깝게 추출한 경우이다. 이와같은 문제를 해결하기 위해서 전처리 단계에서 굴곡점을 미리 추출하여 단위벡터를 생성할 때 굴곡점일 경우 단위벡터를 더 이상 성장시키지 않고 그 시점에서 벡터값을 생성한다.

$$|V(i-T, i) - V(i, i+T)| > Th \Rightarrow \text{accept} \quad \text{---- (식-4)}$$

where i : 현재 위치, T : 상수,

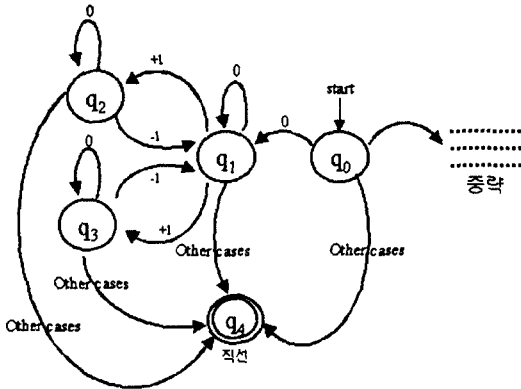
V : 두 점간의 벡터값, Th : 굴곡점 후보값 임계치

글꼴점을 추출하기 위해서 (식-4)를 이용하여 후보 글꼴점들을 추출한다. 그러면 글꼴점이 발생하는 곳은 많은 후보들이 발생하게 된다. 그 후보들 중에서 글꼴점은 후보들의 위치 값의 평균의 통해서 글꼴점을 추출한다.

3.3.2. 제스처의 프리미티브(Primitive) 성분 추출

이제까지는 프리미티브 성분을 추출하기 위한 단위 벡터의 생성방법에 대해서 알아보았다. 프리미티브 성분은 제스처를 인식하는데 가장 중요한 성분이 된다. 이 성분은 각 제스처의 원형(Prototype)의 저장형태이기도 하다.

프리미티브 성분을 추출하기 위해서는 사용되는 방법으로는 유한 오토마타(DFA)중에서 무어 기계(Moor Machine)방식을 사용하여 작성하였다. 무어 기계는 최종 상태(Final State)가 출력 결과를 알려주는 형태이다[5].



[그림-3] 프리미티브 성분 추출 오토마타의 일부분

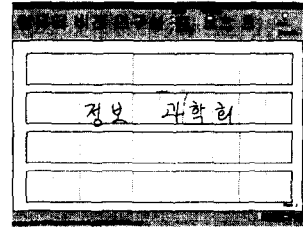
[그림-3]은 프리미티브 성분을 추출하는데 있어서 사용된 오토마타이다. 이 오토마타는 직선 프리미티브 성분을 추출하는데 사용되는 부분이다. 이 오토마타를 통해서 직선, 곡선, 마지막으로 원을 추출할 수 있게 된다.

3.3.3. 제스처의 인식

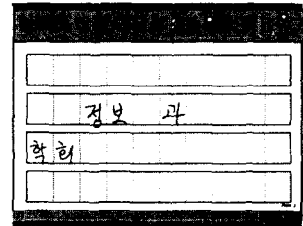
제스처가 가지는 가장 큰 특징은 획수이다. 물론 사람에 따라서 획수가 틀려질 수 있다. 그러므로 사람마다의 특징을 구분하여 획수별로 클래스를 나눈다. 클래스에는 한 제스처가 중복적으로 포함될 수 있으며, 획을 중심으로 하여 한 획으로 구성된 클래스-1, 2획으로 구성된 클래스-2 마지막으로 3획이상으로 구성된 클래스-3으로 구분될 수 있다.

그 다음은 각각의 클래스에 대해서 앞에서 설명한 프리미티브 성분을 비교하여 제스처를 인식하도록 한다. 비교하는 방법은 각각의 클래스 별로 인식하는 방법을 달리한다. 현재까지는 1획의 클래스까지 구현이 되어있는 상태이다. 1획의 클래스의 인식은 오토마타를 사용하여 1획으로 구분되어질 수 있는 제스처를 인식한다.

4.구현 결과



[그림-4] 제스처(다음줄 바꿈)를 입력한 화면



[그림-5] 제스처가 적용된 후의 화면

5.결과

이번에 제안한 제스처 인식방법은 정규화과정을 거치지 않은 상태에서 단위벡터에서 프리미티브 성분을 추출하는 방법을 사용함으로써 같은 제스처를 사용하는데 크기에 덜 민감하게 인식을 할 수 있게 하였다. 현재는 제스처를 인식하는데 필요한 성분들은 모두 추출한 상태이고 그 중에서 클래스-1의 경우는 인식할 수 있도록 되어져 있다. 아직은 클래스-2와 클래스-3의 제스처에 대해서는 완성이 되지 않은 상태이지만, 지금 구현중에 있으므로 빠른 시일내에 완성될 것으로 생각되어진다.

온라인 인식기와 제스처 인식기를 통합하면서 아직까지도 여러글자를 입력하였을 때 음절과 음절을 분할하는데는 연구가 미흡하다고 생각이 든다. 현재는 원고지 형태에 입력을 함으로써 음절을 구분하지만 앞으로는 원고지 형태가 아닌 일반적인 노트와 같은 형태에서 글자를 구분할 수 있는 연구가 필요하다고 생각이 든다.

본 연구는 한국과학재단의 핵심기초 연구 지원에 의해 이루어 졌습니다. <과제명 : "펜 컴퓨터를 위한 온라인 문자 인식(OCRPC)">

6.참고문헌

- [1] 정우식, 장항배, 권영민, 권오성, "개선된 스트링 매칭을 이용한 온라인 한글인식", Vol 11, No2, pp121-124, 1998
- [2] 권오성, 권영민. "선분정합에 의한 흘림체 온라인 한글 인식", 인지과학, Vol.3, No2, pp271-289, 1993
- [3] 최기호, 한글맞춤법 새 길라잡이, pp331-333, 토담, 1994
- [4] 김선, "맞춤법, 띄어쓰기, 원고지 사용법", pp122-142, 예문당, 1996
- [5] 김대수, "오토마타와 계산이론", 생능출판사, 1996
- [6] Sing-Tze Bow, Pattern Recognition and Image Preprocessing, Marcel Dekker, New York, 1992
- [7] C.C Tappert, Ching Y. suen, and Toru Wakahara, "The State of the Art in On-Line Handwriting Recognition",