Substroke HMM 기반 온라인 필기체 문자인식

°김 춘 영, 석 수 영, 정 호 열, 정 현 열
영남대학교 대학원 멀티미디어통신공학과

On-line Handwriting Recognition Based on Substroke HMM

°Choon-Young Kim, Soo-Young Suk, Ho-Youl Jung, Hyun-Yeol Chung
Department of Multimedia and Communication Eng., Yeungnam University
cykim@yumail.ac.kr

요 약

본 논문에서는 자연스러운 온라인 필기체 문자 인식을 위하여 획 기반 HMM(Substroke HMM)을 기반으로 한 인식 방법을 채택하고, 획 분류의 정확도 향상을 위한 전처리 과정에 대해 재샘플링 간격 조정을 통한 획 분류 실험을 통해 인식률 제고에 관한 실험을 수행하였다. 필기체 문자인식을 위한 방법으로 한 문자 전체를 HMM으로 구성하는 Whole-character HMM과 자소단위를 HMM으로 구성하는 character HMM을 주로 이용하였으나, 이러한 방법은 문자의 수에 비례하여 비교적 큰 메모리 용량과 계산량이 요구되는 단점이 있다. 이러한 단점을 개선하기 위한 획 기반 HMM은 문자를 획단위로 분류한 후 이를 HMM 모델로 구성하므로 소수의 획 기반 HMM 모델만으로 문자를 모두 표현할 수 있는 장점을 가지고 있어, 인식률의 큰 저하 없이 계산량 및 메모리 용량을 크게 줄일 수 있다. PDA상에서 수집한 완성형 한글 데이터베이스를 사용하여 획 분류 실험을 수행한 결과 평활화와 7/100 길이의 재샘플링을 수행한 경우 평활화 과정을 추가하지 않은 기존의 재샘플링 5/100 길이의 경우에 비해 정확도가 평균 3.7% 향상을 나타내었으며, 특히 첨가 에러율이 감소함을 확인할 수 있다.

1. 서 론

1980년대 초 컴팩(Compaq)사가 휴대용 컴퓨터를 발표한 이래 컴퓨터 업체들은 보다 가볍고 휴대하기 간편한 휴대용 컴퓨터를 개발하고자 노력해 왔다. 이 결과, 현재 빠른 데이터 처리 능력을 가지며, 펜 입력장치 및 통신기능을 보유한 보다 편리하고 새로운 휴대용 정보단말기로서 PDA(Personal Digital Assistant)와 스마트폰(Smart Phone)이 개발되어 널리 사용되고 있다. 컴퓨터가 소형화됨에 따라 기존의 키보드와 같은 물리적 문자입력 수단은 크기의 소형화에 한계가 있기 때문에 보다 새로운 입력 방법이 필요하게 되었고 이를 위하여 온라인 문자인식이나 음성인식이 사용되고 있다.

온라인 문자인식은 전자펜과 같은 입력 장치로 타블렛(Tablet)에 문자를 써 나가면 실시간으로 쓰여진 문자를 인식하는 방법이다. 그러나 일반 사용자들은 펜 입력 기능을 장착한 시스템의 인쇄체와 흘림체에 모두에 대해 높은 인식성능을 요구하고 있어 보다 정교한 인식 알고리즘이 필요하게 되었으며, 기본 인식모델로서는 일반적으로 HMM을 많이 이용하여 왔다. 이 경우, Whole-Character 단위로 HMM을 구성하는 경우에는 복잡한 문자를 정확히 표현하기가 어렵고, 문자 수가 증가함에 따라 요구되는 메모리 용량과 계산량이 크게 늘어나는 단점이 있다[1].

이러한 문제점을 보완하기 위해 제안된 획 기반 HMM (Substroke HMM)은 기존의 Whole-character HMM에 비하여 보다 적은 수의 HMM만으로 복잡한 형태의 문자도 효과적으로 표현할 수 있으며 인식 속도의 향상 및 메모리 용량의 감소, 인식 성능의 향상 등 여러 가지 장점을 가지고 있다[2][3].

본 논문에서는 획 기반 HMM을 한국어 온라인 문자인식에 적용하여 인식시스템을 구현하고, 전처리 과정을 통해 문자를 획으로 분리하는 과정에서 발생할 수 있는 획 단위 분류 에러를 최소화시키기 위한 방법을 강구하기 위해 재샘플링과 평활화과정을 통해 획분류의 정확도를 향상시키고자 한다. 논문의 구성은 다음과 같다. 다음 2절에서는 문자 인식 시스템의 구성에 관해 기술하고, 3절에서는 획 기반 HMM의 구성에 대해 기술한 후 4절에서는 전처리 과정 수행 후의 획 분류 에러율을 상호 비교, 분석하고 결론을 맺는다.

2. 문자 인식 시스템 구성

획 기반 HMM의 온라인 필기체 인식 시스템은 그림 1과 같은 순서로 구현될 수 있다. 먼저 전처리 과정을 통해 입력된 문자는 정규화되고, 정규화된 문자 데이터로부터 특징 파라미터를 추출한 다음 획 분할을 수행한다. 분할된 획에 대하여 획 기반 HMM을 생성하고 훈련 과정을 거쳐 인식과정을 거친다.

전처리 단계에서는 획 분할시 발생할 수 있는 에러를

최소화하기 위하여 샘플링 간격을 일정간격으로 조정하는 재샘플링(Re-sampling)을 실시하고 재샘플링된 결과에 평활화(Smoothing)를 수행한다. 이하 전처리 과정과 입력 파라미터에 대해 간략한다.

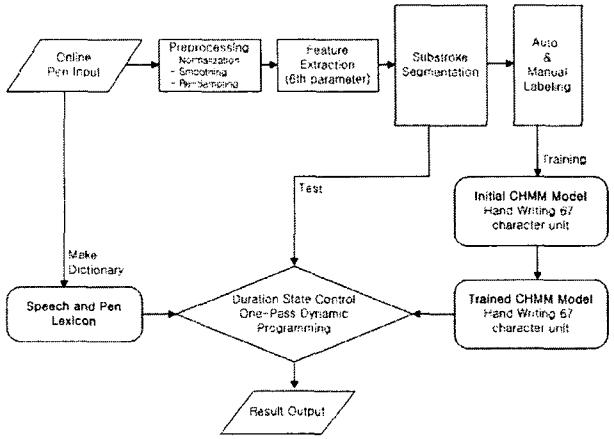

그림 3. 문자 인식 시스템 구성도

2.1 전처리

온라인 필기체 문자 데이터는 타블렛 또는 터치스크린으로부터 일정한 샘플링 간격으로 표본화된 펜의 위치정보를 나타내는 x, y 좌표값으로 구성된다. 입력되는 문자는 필자의 개성과 펜의 속도에 따라 글자형태가 다양하게 나타나기 때문에 올바른 특징 추출을 위하여 이를 전처리 과정에서 정규화할 필요가 있다. 입력 데이터의 정규화를 위하여 본 논문에서는 평활화, 크기 위치 정규화 및 재샘플링을 이용한다. 평활화는 필자가 필기하는 도중 펜이 떨려서 글씨가 울퉁불퉁하게 입력되는 문제를 이웃한 점과의 연관성을 고려하여 평탄화함으로써 해결하는 기법으로 본 논문에서는 식(1), (2)를 이용한 3포인트 평활화 방법을 이용한다[5].

$$x_i = (x_{i-1} + x_{i+1})/4 + x_i/2 \qquad (1)$$

$$y_i = (y_{i-1} + y_{i+1})/4 + y_i/2 \qquad (2)$$

재샘플링은 필기속도의 불균일과 입력 장치의 샘플링 간격의 차이에 의해 발생되는 입력 포인트 사이의 불균등한 간격을 정규화하기 위한 과정으로 입력된 열로부터 일정한 간격의 점들의 열로 새롭게 생성하게 한다[4].

그림 2는 재샘플링 방법을 나타낸다. 그림에서 j 는 재샘플링과정 전의 열, i 는 재샘플링 과정 후 새롭게 만들어지는 열을 의미한다. 재샘플링을 수행하기 전의 두 포인트 (px_{i-1}, py_{i-1}), (x_j, y_j)간의 간격(Dis)이 샘플링 간격(sth)만큼의 길이가 되기 위한 x, y좌표값의 변화분을 비례식으로 나타내면 식(3), (4)와 같다.

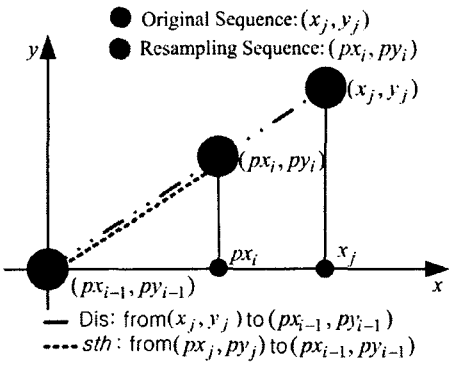

그림 4. 재샘플링

$$Dis : sth = (x_j - px_{i-1}) : (px_i - px_{i-1}) \qquad (3)$$

$$Dis : sth = (x_j - px_{i-1}) : (px_i - px_{i-1}) \qquad (4)$$

식(3), (4)를 px_i, py_i에 대해서 정리하면 식(5), (6)과 같고 이를 이용하여 재샘플링을 수행하게 된다.

$$px_i = \frac{sth \times (x_j - px_{j-1})}{Dis} + px_{j-1} \qquad (5)$$

$$py_i = \frac{sth \times (y_j - py_{j-1})}{Dis} + py_{j-1} \qquad (6)$$

$$Dis = \sqrt{(px_{j-1} - x_j)^2 + (py_{j-1} - y_j)^2} \qquad (7)$$

샘플링 간격에 따른 재샘플링의 결과는 그림 3과 같이 샘플링 간격이 너무 작으면 포인트가 조밀하게 추출되므로 획 분할시 획 순서열에 대표 벡터 외에 불필요한 벡터가 포함되는 에러가 발생할 가능성이 높다. 반대로 샘플링 간격이 너무 크면 한 획을 이루는 포인트의 수가 적을 경우 획 분할시 획 순서열에 대표 벡터에 해당하는 벡터가 삭제되는 에러가 발생할 가능성이 높아진다. 따라서 이러한 에러가 최소가 되는 최적의 샘플링 간격을 찾을 필요가 있으며, 이 최적 재샘플링 결과에 평활화를 추가하여 전처리 과정을 수행하면 보다 나은 결과를 얻을 수 있다.

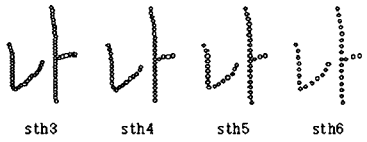

그림 5. 샘플링 간격의 변화에 따른 재샘플링의 결과

2.2 특징 파라미터

인식 시스템의 특징파라미터는 자소 사이의 구분을 가능하게 하고, 다양한 필기체의 변형을 모두 흡수할 수 있는 특징들을 추출해야 한다. 본 논문에서는 특징파라미터로서 Δx, Δy 좌표값, 국부적 각도 파라미터(sinθ, cosθ)값, 국부적 만곡 파라미터(Δsinθ, Δcosθ)값 등 총 6차의 파라미터를 추출하여 이용한다. 각 파라미터는 식(8)~(13)을

이용하여 구할 수 있다.

$$\Delta x[i] = x[i+1] - x[i-1] \quad (8)$$

$$\Delta y[i] = y[i+1] - y[i-1] \quad (9)$$

$$sin\theta_i = \frac{\Delta y[i]}{\sqrt{\Delta x[i]^2 + \Delta y[i]^2}} \quad (10)$$

$$cos\theta_i = \frac{\Delta x[i]}{\sqrt{\Delta x[i]^2 + \Delta y[i]^2}} \quad (11)$$

$$\Delta sin\theta_i = sin(\theta_{i+1} - \theta_{i-1}) \quad (12)$$

$$\Delta cos\theta_i = cos(\theta_{i+1} - \theta_{i-1}) \quad (13)$$

3. 획 기반 HMM의 구성

획 기반 HMM은 많은 종류의 문자를 소수의 획만으로 모두 표현할 수 있어 메모리가 제한된 시스템에서 모델의 수를 감소시킴으로 효율성을 증가시킬 수 있는 장점을 가지고 있다[2][3]. 한글은 일본어나 한자의 경우에 비해 비교적 간단한 획으로 구성할 수 있으며 그림4에 나타낸 바와 같이 17개 정도의 획 만으로도 모두 표현이 가능하다. 획은 그림 4에서 보는 바와 같이 pen-down상태에서는 'A-H'로, pen-up상태에서는 '1-8'로 표현할 수 있다. 'ㅇ'과 같은 획은 한 획 내에서 동일한 방향으로 연속적으로 변하는 방향 벡터의 개수가 임계값 이상일 때 'ㅇ' 획으로 결정하여 별도로 구현한다.

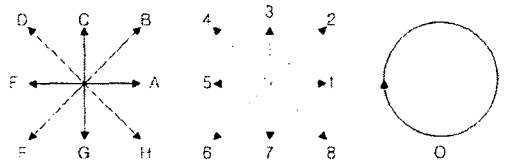

그림 6. 한글 표현에 필요한 획 벡터

표 1. 자음의 획 대표 순서열

자소	Substroke	자소	Substroke	자소	Substroke
ㄱ	AF	ㅂ	GGAA	ㅊ	AAFH
ㄲ	AGAG	ㅃ	GGAAGGAA	ㅋ	AAG
ㄴ	GA	ㅅ	FH	ㅌ	AAGA
ㄷ	AGA	ㅆ	FHFH	ㅍ	AGGA
ㄸ	AGAAGA	ㅇ	O	ㅎ	AAO
ㄹ	AGAGA	ㅈ	AFA		
ㅁ	GAGA	ㅉ	AFHAFH		

표 2. 모음의 획 대표 순서열

자소	Substroke	자소	Substroke	자소	Substroke
ㅏ	GA	ㅠ	AGG	ㅚ	GAG
ㅑ	GAA	ㅡ	A	ㅙ	GAGAG
ㅓ	AG	ㅣ	G	ㅝ	AGAG
ㅕ	AAG	ㅢ	AG	ㅞ	AGAGG
ㅗ	GA	ㅐ	AGG	ㅟ	AGG
ㅛ	GGA	ㅒ	GAG	ㅘ	GAAG
ㅜ	AG	ㅘ	GAGA	ㅙ	AAGG

본 논문에서는 각 자소에 따른 획 대표 순서열을 19개 자음과 21개 모음에 대해 각각 표 1, 표 2와 같이 구성한다. 또, 다양한 필기체를 허용하기 위해 추가적 획 순서열을 이용한다. 예를들어 문자가 정자로 입력되었다고 가정하면 '가' 문자의 경우 "A F 2 G 3 A"와 같이 나열될 수 있고, '안' 문자의 경우 "O A 3 A 6 G A"와 같이 나열될 수 있으나, 자소의 배열 및 필자의 특성에 따라 "A G 2 G 3 A"혹은 "A H G F 2 G 3 A" 와 같이 나타날 수 있음을 의미한다.

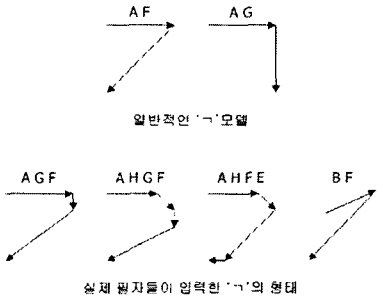

그림 7. 획 순서열의 예

이와 같이 획 순서열 만을 통해 인식을 수행하는 경우에는 복잡한 문자도 비교적 적은 수의 벡터로 표현할 수 있으므로 계산량을 줄이는 장점이 있지만, 그림 5에서 보는 바와 같이 필자의 필기 개성에 따라 한 문자에 대해 다양한 순서열이 존재할 수 있어 이에 따른 오인식이 증가하는 문제점을 가지고 있다. 또한, 문자를 기울어지게 필기하거나 문자의 필기 순서를 일반적인 경우와 다르게 하는 경우에는 전혀 다른 순서열로 출력될 수도 있으며, 서로 다른 문자임에도 불구하고 같은 순서열로 구성될 수도 있다. 예를 들어 'ㅓ'와 'ㅜ'는 서로 다른 문자이지만 동일한 'A G' 획 순서열로 구성되며, 'ㅏ'와 'ㅜ'는 동일한 'A' 획 벡터와 'G' 벡터를 이용하지만 서로 다른 획 순서열로 구성되어 있다.

획 벡터를 추출한 경우 기준 벡터 순서열 외의 벡터가 순서열에 추가된 경우를 '첨가 에러'로 규정하고, 기준 벡터값이 순서열에 나타나지 않은 경우를 '삭제 에러', 그리고 기준 벡터값과 다른 벡터값이 순서열에 나타난 경우를 '대체 에러'로 규정한다.

본 논문에서는 이러한 에러율을 최소화 시키기 위하여, 전처리 과정에서 등간격의 재샘플링을 수행한다. 재샘플링시 샘플링 간격을 작게 하면 보다 조밀한 간격으로 특징점을 추출할 수 있으므로 삭제 에러율은 줄일 수 있으나 첨가 에러율은 늘어날 수 있다. 반대로 샘플링 간격을 크게 하면 첨가 에러율은 줄일 수 있으나 삭제 에러율이 증가하게 된다. 따라서, 이와같은 trade-off 관계를 고려하여 실험을 통하여 최적 간격을 찾을 필요가 있다. 또, 재샘플링에 평활화과정을 추가하여 필기상의 떨림에 의한 방향 벡터의 불필요한 변화를 줄임으로써 첨가 에러율을 더욱 줄이고 인식률을 높일 수 있다.

획 벡터로 추출된 각 획은 그림 6에서와 같이 left-to-right CHMM(Continuous HMM)으로 모델링 한다.

pen-down 상태에 있는 획에 대해서는 획을 이루는 포인트의 수가 적을 경우를 감안하여 2 상태 CHMM을 이용하여 모델링 하고, pen-up 상태에 있는 획에 대해서는 자기 천이 확률이 없는 1상태 HMM으로 모델링 한다.

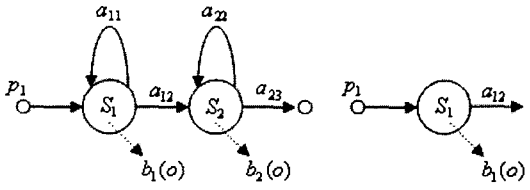

그림 8. 획 기반 연속HMM의 구성
(pen-down모델(왼쪽)과 pen-up모델(오른쪽))

4. 실험 및 고찰

한글 데이터베이스는 PDA와 같은 터치스크린을 이용한 온라인 입력문자 인식에의 적용을 고려하여 PDA상에서 수집한 100명분의 데이터베이스를 이용한다. 이는 470개의 글자로 구성된 완성형 한글 5set을 20명의 필자가 한 set씩 필기한 것이다.

획 기반 HMM의 경우 획 분류 에러 발생시에도 누적 확률값에 의해 자소단위 인식성능은 향상되나, 정확한 획 분류를 통하여 직접적인 인식률 향상을 가져오게 할 수 있다. 따라서 본 연구에서는 최적의 재샘플링 간격을 찾아내기 위하여 간격조정에 따른 획 분류 에러율을 조사하였다.

표 3. 재샘플링 간격에 대한 자음의 획 분류 에러율 (+1: 1 획 첨가 혹은 삭제, +2↑: 2 획 이상 첨가 혹은 삭제)

자음	정확도	첨가에러율		삭제에러율		대체에러율
		+1	+2 ↑	+1	+2↑	
RS sth=4	59.1%	13.1%	21.4%	2.3%	0.0%	2.7%
RS sth=5	69.2%	12.2%	11.0%	2.5%	0.0%	3.6%
RS sth=6	72.8%	9.1%	6.9%	2.7%	0.0%	6.9%
RS sth=7	75.6%	7.3%	3.4%	4.6%	0.2%	7.6%
RS sth=8	75.0%	5.4%	2.6%	6.6%	1.2%	7.7%
SM + RS	76.7%	5.8%	3.1%	4.6%	0.2%	8.1%

표 4. 재샘플링 간격에 대한 모음의 획 분류 에러율

자음	정확도	첨가에러율		삭제에러율		대체에러율
		+1	+2↑	+1	+2↑	
RS sth=4	82.2%	9.2%	2.3%	3.2%	0.5%	1.3%
RS sth=5	90.0%	3.6%	0.6%	3.5%	0.5%	0.9%
RS sth=6	89.2%	4.0%	0.5%	3.0%	0.5%	1.9%
RS sth=7	90.6%	3.5%	0.0%	1.9%	0.5%	2.7%
RS sth=8	89.2%	0.7%	0.0%	3.0%	0.5%	5.8%
SM + RS	86.2%	2.4%	4.0%	3.7%	0.3%	2.7%

표 3과 4는 각각 자음과 모음에 대한 획 분류 실험 결과를 나타내며, 기준 벡터가 순서열에 1개, 2개 이상 추가된 첨가 에러와 벡터가 순서열에 1개, 2개 이상 나타나지 않은 삭제 에러, 그리고 기준 벡터값이 다른 벡터값으로 나타난 대체 에러를 조사하여 나타낸 것이다. 표3, 4로부터 자음보다 모음의 획분류 정확도가 높음을 확인할 수 있으며, 이는 모음의 획 구성이 자음의 획 구성 보다 상대적으로 간단하기 때문으로 생각된다. 7/100 길이의 재샘플링을 수행한 경우 획 분류 정확도가 가장 높음을 확인할 수 있으며, 재샘플링 간격이 작은 경우 추가적인 획이 첨가되는 에러가 높아지며, 재샘플링 간격이 큰 경우 짧은 획이 삭제되는 에러율이 증가함을 확인할 수 있다. 최적 재샘플링 간격값(7/100)에 평활화 과정을 추가적으로 사용한 경우 평활화 과정을 추가하지 않은 기존의 재샘플링 간격값(5/100)의 경우에 비해 정확도가 평균 3.7% 향상함을 알 수 있으며, 특히 첨가 에러율이 감소함을 확인할 수 있다.

5. 결 론

본 논문에서는 자연스러운 온라인 필기체 인식을 위하여 획 기반 HMM(Substroke HMM)을 도입하여 이를 기반으로 한 온라인 필기체 인식 시스템을 구현하였다. 획 기반 HMM은 기존의 Whole-character HMM에 비하여 보다 적은 수의 HMM만으로 복잡한 형태의 문자도 효과적으로 표현할 수 있으며 인식 속도의 향상 및 메모리 감소, 인식 성능의 향상 등 여러 가지 장점을 가지고 있다. 획 기반 HMM의 사용시 획 분류의 정확도 향상은 직접적으로 인식률의 향상을 가져온다. 따라서 재샘플링 간격을 조정하며 획 분류 실험을 수행한 결과 평활화와 7/100 길이의 재샘플링을 수행한 경우가 평활화와 재샘플링을 수행하지 않은 경우에 비해 정확도가 평균 3.7% 향상됨을 알 수 있었으며, 특히 첨가 에러율이 감소함을 확인할 수 있었다.

참고문헌

[1] J.Hu, M.K.Brown and W.Turin, "HMM Based On-Line Handwriting Recognition", *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1996.

[2] Junko TOKUNO, Nobuhito INAMI, Shigeki MATSUDA, Mitsuru NAKAI, Hiroshi SHIMODAIRA, Shigeki SAGAYAMA, "Context-dependent Substroke Model for HMM-based On-line Handwriting Recognition", *Proc.IWFHR'02*, 2002.

[3] Mitsuru NAKAI, Kaoto AKIRA, Hiroshi SHIMODAIRA and Shigeki SAGAYAMA, "Substroke Approach to HMM-based On-line Kanji Handwriting Recognition", IEEE, 2001

[4] 석수영, 정현열, "비트맵' 파라미터를 이용한 온라인 필기체 문자인식", 신호처리 합동학술대회, 2001.

[5] 성운재, "계층적 곡선표현기법을 이용한 온라인 필기 한글 인식", 한국과학기술원 전산학과 석사학위 논문, 1991.