

필기체 한글 문자 인식을 위한 획 추출에 관한 연구

최 영 규[†] · 이 상 범^{††}

요 약

필기체 문자 인식은 온라인 필기체 문자 인식과 오프라인 필기체 문자 인식으로 나누어진다. 온라인 필기체 문자 인식은 태블릿과 같은 펜 기반의 전자식 입력 장치를 이용하여 필기의 순서와 획의 위치와 같은 동적인 필기 정보를 문자의 입력 시 획득할 수 있어 오프라인 필기체 문자 인식에 비해 큰 연구 성과를 이루었다. 그러나 오프라인 필기체 문자 인식은 온라인 필기체 문자 인식에서와 같이 동적인 정보를 입력 받을 수 없고, 다양한 필기와 자소의 겹침이 심하며 획 사이의 잡영을 많이 가지고 있어 인식의 전처리 결과에 따라 인식 성능이 크게 달라진다. 본 논문에서는 오프라인 필기체 한글 문자 인식을 위해 문자의 동적인 정보를 포함하는 획을 효과적으로 추출하는 방법을 제안한다. 제안된 방법은 전처리 과정으로 먼저 Watershed 알고리즘을 이용하여 입력된 필기체 문자 영상의 향상 및 이진화를 수행한다. 이진화된 문자부를 변형된 L_ü와 Wang의 세선화 알고리즘을 사용하여 세선화를 수행한 후 문자에서의 특징점을 추출하여 세그먼트 화소열을 추출하고, 최대 허용 오차법을 이용하여 벡터화 한다. 벡터화의 수행으로 몇 개의 획이 하나의 세그먼트로 묶인 경우, 하나의 세그먼트 화소열은 2 또는 그 이상의 세그먼트 벡터로 분리된다. 추출된 세그먼트 벡터들을 완전한 획으로 재구성하기 위해서 오른손 필기 좌표계 시스템을 이용하여 벡터의 방향적인 성분을 인간의 필기 획의 방향에 알맞게 수정하고, 수정된 세그먼트 벡터의 방향성과 분기 정보를 이용하여 인접한 결합 가능한 세그먼트 벡터를 결합함으로써 문자 인식에 적합한 완전한 획으로 재구성한다. 실험 결과 제안된 방법이 필기체 한글 문자 인식에 적합함을 알 수 있었다.

A Study on Stroke Extraction for Handwritten Korean Character Recognition

Young Kyoo Choi[†] · Sang Burm Rhee^{††}

ABSTRACT

Handwritten character recognition is classified into on-line handwritten character recognition and off-line handwritten character recognition. On-line handwritten character recognition has made a remarkable outcome compared to off-line handwritten character recognition. This method can acquire the dynamic written information such as the writing order and the position of a stroke by means of pen-based electronic input device such as a tablet board. On the contrary, Any dynamic information can not be acquired in off-line handwritten character recognition since there are extreme overlapping between consonants and vowels, and heavily noisy images between strokes, which change the recognition performance with the result of the preprocessing. This paper proposes a method that effectively extracts the stroke including dynamic information of characters for off-line Korean handwritten character recognition. First of all, this method makes improvement and binarization of input handwritten character image as preprocessing procedure using watershed algorithm. The next procedure is extraction of skeleton by using the transformed L_ü and Wang's thinning algorithm, and segment pixel array is extracted by abstracting the feature point of the characters. Then, the vectorization is executed with a maximum permission error method. In the case that a few strokes are bound in a segment, a segment pixel array is divided with two or more segment vectors. In order to reconstruct the extracted segment vector with a complete stroke, the directional component of the vector is modified by using right-hand writing coordinate system. With combination of segment vectors which are adjacent and can be combined, the reconstruction of complete stroke is made out which is suitable for character recognition. As experimentation, it is verified that the proposed method is suitable for handwritten Korean character recognition.

키워드 : 한글 문자 인식(Korean character recognition), 획 추출(stroke extraction), 세선화(thinning), Watershed 알고리즘(watershed algorithm)

1. 서 론

일반적으로 문자 인식 시스템은 전처리(preprocessing) 과정, 특징 추출 과정, 매칭(matching), 후처리(postprocessing) 과정으로 이루어진다. 전처리 과정의 주된 목적은 문

자에서 특징을 추출하기 위해 입력 패턴을 보다 의미있는 패턴으로 재표현 하는 것이다[1]. 입력 장치로부터 얻어진 데이터에서 시간적 또는 공간적인 정보를 잃게 되는 오프라인 필기체 한글 문자 인식에서는 윤곽선(contour)을 추출하거나 골격선(skeleton)을 추출하는 등의 전처리를 수행하여야 한다. 그러다 보니 입력 문자에 포함될 수 있는 많은 잡영과 왜곡으로 다양한 형태의 필기체 한글 문자는 많은 정보의 손실과 왜곡을 동반한 불완전한 전처리를 수행한다

[†] 준 회원 : 단국대학교 대학원 전자공학과
^{††} 종신회원 : 단국대학교 공학부 컴퓨터공학전공 교수
논문접수 : 2002년 2월 22일, 심사완료 : 2002년 4월 25일

는 어려움을 가지고 있다. 따라서 인식의 전처리 과정의 성능과 품질의 질적인 향상에 따라 인식률의 향상이 기대되어진다. 오프라인 필기체 한글 문자 인식에 있어 전처리 성능 및 품질 개선에 대한 많은 연구 논문이 끊임없이 발표되고 있지만 아직도 연구에 있어 다양한 방법의 접근이 필요하다. 필기체 한글 문자에 있어 영상 분할 즉 문자부와 배경부를 분리하는 과정은 중요한 과정으로 이진화를 위한 최적의 문턱치(threshold) 설정에 많은 연구가 있어 왔다. 문턱치 설정은 크게 전역적인 문턱치 설정과 지역적인 문턱치 설정으로 나누어지며, 영상 분할에서 전역적인 문턱치 설정은 처리 속도는 빠르나 영상의 질 문제에 영향을 준다. 그리고 지역적인 문턱치 설정에 의한 방법은 전역적인 문턱치 방법 보다 질적인 문제는 개선되었지만 수행속도는 느린 단점이 있다. 따라서 영상 향상을 통한 개선된 영상을 전역적인 문턱치 방법을 수행하여 배경과 문자 부분을 분리한 영상 결과를 얻어내는 것이 일반적인 방법이다[2].

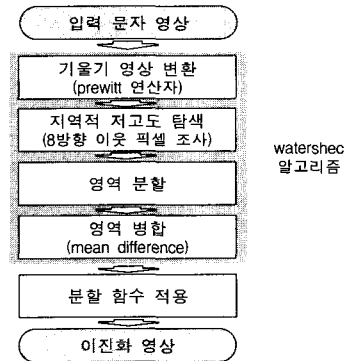
오프라인 필기체 한글 문자 인식의 방법 중 문자의 기하학적인 특징을 사용하는 방법은 필기체 한글 문자의 방대함과 필체의 다양성, 글자간의 유사성이 높아 필기체 한글 문자 처리에 어려움이 있다. 한글 문자를 구성하는 획(stroke)을 추출하고, 획간의 패턴을 분석하여 인식하는 방법은 인간이 직관적으로 문자를 인지하는 방법과 유사하며, 획의 위치, 방향성, 획 들간의 구조적인 관계와 같은 다양한 획의 구조 정보를 이용하기 때문에 기하학적인 특징을 사용하는 방법에 비해 인식 시 더욱 안정된 매칭이 가능하다. Doermann과 Rosenfeld는 래스터 영상에서 시간적인 정보를 복원하기 위해서 지역적, 전역적 단서를 이용하는 방법을 제안했다[3]. 이러한 단서는 입력되는 필기체 문서의 그레이 영상과 필기 펜의 종류에 따라 변화가 심하기 때문에 이러한 접근 방식은 픽셀밀리나 복사기에 의한 입력 영상에 대해서는 좋은 결과를 기대할 수 없다. Privitera와 Plamondon는 필기체 문자를 윤곽선을 이용하여 불완전한 획을 추출하는 방법을 제안했는데, 이 방법은 획이 분기되는 부분과 교차되는 부분과 같은 문자의 특정한 사각 영역을 찾아내는 방법을 사용한다[4]. 그러나 이러한 특정한 영역은 접합되는 획의 굵기와 방향에 따라 가변적이므로 그 분포가 일정하지 않아 사각 영역의 크기를 결정하는 것이 어렵다. Chang과 Yan은 EDT(Euclidean distance transform) 기반의 세션화 알고리즘과 획 분할 방법을 이용한 곡선 추적으로 한자에서의 획을 추출하는 방법을 제안했다[1]. 그러나 이 방법은 한글 문자에 비해 비교적 직선에 가까운 획으로 이루어진 한자를 대상으로 하는 방법으로 한글 문자에의 적용에는 적합하지 않음을 알 수 있다.

본 논문에서는 오프라인 필기체 한글 문자에서 획을 추출하기 위한 새로운 방법을 제안한다. 제안된 방법은 첫째, 전처리 과정으로써 Watershed 알고리즘을 이용하여 입력된 필기체 문자 영상의 향상, 이진화 그리고 특징점 추출에 적합하게 변형된 Lü와 Wang의 세션화로서 골격선 및 문자의

특징점 추출을 수행하고 둘째, 획 추출을 위한 방법으로 추출된 특징점을 이용한 세그먼트(segment) 화소열 추출, 세그먼트 추적에 의한 백터화 그리고 세그먼트 백터의 분리 및 병합을 수행하여 문자 인식에 적합한 완전한 획을 추출한다.

2. 제안된 전처리 과정

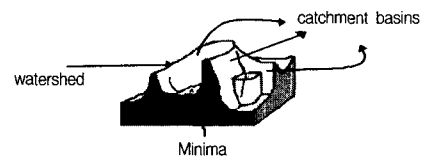
2.1 Watershed 알고리즘을 이용한 이진화



(그림 1) 이진화 구성도

본 논문은 Watershed 알고리즘을 적용하여 문자 영상의 잡영을 효과적으로 흡수하여 영상 향상의 효과를 통해 배경 영역과 문자 영역을 분리하는 이진화를 수행하였다. 수행 시간과 품질을 고려해 Watershed 알고리즘 수행 단계에서 좋은 결과를 보인 방법들로서 알고리즘을 구성하였다. 영상 향상 및 분할의 전체적인 구성도는 (그림 1)에 보여준다. Watershed 영상 변환 과정 중 고도를 가진 경사 영상을 구하기 위해 프리윗(prewitt) 기울기 연산자를 이용하였고, 8방향 이웃 픽셀을 고려한 지역적 저고도 값(local minima)을 추출하였다. 그리고 영역 병합 단계에서는 평균값의 차를 이용한 방법을 이용하였고, 이러한 방법을 통해 Watershed 영상을 분할 함수를 적용하여 배경 영역과 문자 영역을 효과적으로 분리하였다. 분할 함수는 문턱치 처리로 1차 배경 성분으로 분리된 Watershed 영상의 강도(intensity) 분포를 조사하여 일정한 인접 강도 값과의 거리가 문턱치 이상이면 2차 배경 성분으로 분리한다. Watershed 변환을 수행하여 필기체 문자 영상의 끝선의 홀림이나 배경에 포함될 확률이 높은 잡영을 흡수함으로써 영상 향상의 결과를 얻을 수 있고, 적합한 분할 함수 값을 얻어 배경 성분의 분리를 적절히 수행하였다.

2.1.1 Watershed 알고리즘



(그림 2) 위상학적 표현

Watershed 알고리즘은 영상의 명도를 하나의 위상학적 고도로 생각함으로써 이 위상학적 표면에 물을 떨어뜨리면 물은 낮은 고도(minima altitude)에서부터 흘러 들어가 높은 고도(maxima altitude)까지 물이 차 들어가는 원리를 이용한 것이다. (그림 2)와 같이 catchment basin(Watershed)로 구성된다[5-7]. 가장 낮은 높이에서부터 수면 높이를 점차로 증가시키면서, 고도를 넓히는 과정으로 서로 다른 catchment basin이 서로 접하게 될 경우 가상의 댐을 쌓아서 물이 찬 부분을 서로 분리하고, 인접한 영역의 유사성을 조사하여 유사한 영역을 서로 합치는 과정이다.

Watershed 알고리즘은 영상 처리 분야에서 인접한 영역을 효과적으로 근집화 하는 과정을 통해 영상을 분할하는 목적으로 사용되어지는 기법으로 본 논문에서 오프라인 필기체 문자 영상의 이진화를 위해 적용한 Watershed 알고리즘은 4단계로 구성된다. 1단계는 기울기 영상 변환으로 프리윗 연산자를 이용하여 원 영상으로부터 고도를 가진 경사 영상(gradient image)으로 변환한다. 2단계는 지역적 저고도 탐색을 위해 8방향 이웃 픽셀 탐색 조건을 이용하여 경사 영상으로부터 지역적 저고도 값(local minima)을 가진 픽셀들을 추출한다. 3단계는 영역 분할을 위해 지역적 저고도로부터 높은 고도로 범람 과정(flooding)을 수행하여 유사한 영역을 분할한다. 마지막 4단계에서는 는 영역 병합(region merging)을 위해 평균값의 차(mean difference)를 이용하여 유사한 영역에 대한 병합을 수행한다[8].

• Catchment basin

Catchment basin은 지역적인 minima들로 구성된 클래스들의 집합을 의미한다[9]. Catchment basin(CB)은 식 (1)과 같이 모든 영역 D 에서 minima m_i 가 다른 지역적 minima 클래스 m_j 에 집할 때까지의 클래스로 정의한다.

$$CB(m_i) = \{x \in D \mid \forall_j \in I \{i\} : f(m_i) + T_f(x, m_i) < f(m_j) + T_f(x, m_j)\} \quad (1)$$

Catchment basin은 (그림 1)에서와 같이 지역적인 영향권 내의 집합으로써 영상 분할의 기준을 이룬다.

• Watershed

Watershed (f_w)는 Catchment basin 영역이 아닌 부분을 의미한다. Watershed (f_w)는 식 (2)와 같이 영상의 전체 영역 D 에서 Catchment basin에 보수를 취한 값으로 정의한다[9].

$$f_w = D \cap \left(\bigcup_{i \in I} CB(m_i) \right)^c \quad (2)$$

Watershed는 (그림 1)에서와 같이 영상 분할의 기준이 되는 Catchment basin들의 경계면을 이루면서 Catchment basin들의 상호 작용의 기준이 된다.

• 기울기 영상 변환

Watershed 영상 변환의 첫 번째 단계로 오프라인 필기체

한글 문자 데이터에 기울기 연산자를 적용하여 기울기 영상을 얻어낸다. 기울기 연산은 프리윗 연산자를 이용하였다. 기울기 영상 변환을 통해 오프라인 필기체 한글 문자의 배경 영역과 문자 영역을 에지(edge) 부분을 중심으로 부분적인 분리를 수행함과 함께 영상의 고도를 이용한 Watershed 알고리즘을 수행하기 위한 전 단계를 수행한다. 대부분 수평 방향과 수직 방향으로 이루어진 한글 문자 특성을 고려하여 수평 방향과 수직 방향에 민감한 프리윗 연산자의 특징과 결합한다.

• 지역적 저고도 탐색

$$X_{h_{min}} = \{p \in D \mid f(p) = h_{min}\} \\ h_{min} = \min_{r \in N_8(p)} \{f(r) \mid f(r) < f(p)\}; \quad (3)$$

N_8 : 8-neighbor pixel.

기울기 영상에서 8방향 이웃 픽셀을 조사하여 지역적으로 낮은 고도를 가진 픽셀을 탐색한다. 지역적으로 8방향 이웃 픽셀 탐색 조건의 문턱치 설정은 식 (3)과 같이 전체 공간 D 에서 픽셀의 명도 값 함수 $f(p)$ 가 지역적으로 낮은 명도 값 h_{min} 과 같은 값을 가지는 픽셀의 위치와 값을 저고도 값으로 설정하고, 이 값이 8방향 이웃 픽셀 탐색 조건의 문턱치로 설정되어 전체 공간 D 에서 8방향 지역적 저고도 값을 저고도 값과의 거리를 척도로 해서 추출한다.

• 영역 분할

$$X_{h_{min}} = \{p \in D \mid f(p) = h_{min}\} \\ X_{h+1} = MIN_{h+1} \cup IZ_{T_{h+1}}(X_h), h \in [h_{min}, h_{max}] \quad (4)$$

낮은 고도를 가진 픽셀로부터 지역적으로 가장 높은 고도를 가진 픽셀을 하나의 영역으로 분할한다. 식 (4)와 같이 낮은 고도 $X_{h_{min}}$ 에서부터 낮은 고도에 영향을 주고 있는 픽셀들(IZ : influence zone)안에서 낮은 고도 h_{min} 에서 높은 고도 h_{max} 로 영역들은 합한다. 결국 같은 영향권을 가지는 픽셀들로 영역이 분할된다.

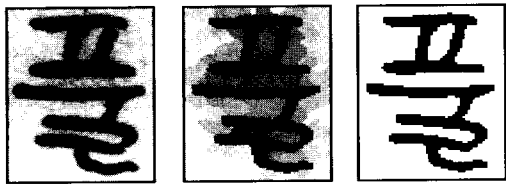
• 영역 병합

$$R_{meandiff} = (R_{SV}(p)/R_S(p)) - (R_{SV}(r)/R_S(r)) \\ R_{SV} = \sum Value, R_S = size \quad (5)$$

분할된 영역들을 인접 영역과의 유사성을 평가하여 유사도가 있는 영역들을 병합하는 과정이다. 인접 영역의 유사도는 서로 다른 인접 영역의 평균값의 차를 이용하여 구하고, 분할된 영역 $R(p)$ 와 인접한 다른 분할된 영역 $R(r)$ 의 평균값의 차 $R_{meandiff}$ 는 식 (5)과 같이 구한다. 얻어진 평균값의 차는 문턱치 비교로서 유사도를 판단한 후 영역을 병합한다. 여기서 평균값은 영상에서의 형태상의 강도를 의미한다. 최종적인 Watershed 알고리즘을 적용하여 변환된 영상은 영상의 평균값 즉 영상에서의 강도 값으로 표현하였다.

2.1.2 분할 함수

Watershed 영상은 영상의 강도로써 표현되었다. 영상의 강도는 배경 성분과 문자 성분을 분류하는 기준으로 사용할 수 있다. Watershed 영상의 강도가 100 이상인 값을 가지면 100 이상의 값은 배경 성분의 값을 의미한다. 따라서 100 이상인 배경성분을 1차로 분리한다. 인접한 강도의 값의 차이가 멀수록 배경 성분일 확률이 높다. 가장 적절한 배경 성분 확률 값을 구하여 2차 배경 성분을 분리하였다. 배경 성분 확률 값을 사용하는 필기체 문자 DB(data base)에서 통계적 확률 수치를 구하여 얻어냈다.



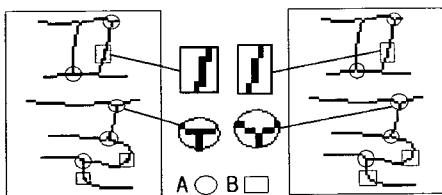
(a) (b) (c)

(그림 3) 문자부와 배경부의 분리

(그림 3)은 Watershed 알고리즘을 이용하여 문자부와 배경부를 분리하여 이진화를 수행하는 과정이다. (그림 3-a)는 필기체 문자 데이터베이스의 래스터 영상이고, (그림 3-b)는 Watershed 알고리즘 적용 결과, (그림 3-c)는 (그림 3-b)의 결과를 분할 함수를 이용하여 이진화한 결과이다.

2.2 골격선 및 특징점 추출

문자에서 획을 추출하기 위한 방법은 문자의 골격선을 추출하여 분기점, 연결점, 끝점, 순환점과 같은 특징점을 추출한 후, 특징점들 사이의 세그먼트 화소열을 벡터화를 수행하여 추출된 세그먼트 벡터를 분리 및 결합하는 과정으로 전개된다. 이 때 문자의 특징점들을 효과적으로 추출하기 위해서는 이진화를 통해 추출된 문자부를 특징점 추출에 적합하게 세선화하는 과정이 필요하다.



(a) Lü와 Wang의 세선화 (b) 변형된 세선화

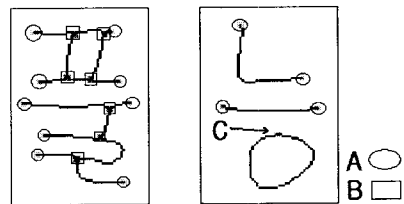
(그림 4) 세선화

본 논문에서는 Lü와 Wang[9]의 세선화 알고리즘의 세선화 결과를 골격선 및 특징점 추출에 적합하게 후처리를 수행하여 사용하였다. Lü와 Wang의 세선화 알고리즘은 (그림 4-a)에서 보는 바와 같이 A 부분의 획의 분기 또는 B 부분의 획의 연결 부분의 특성이 각기 달라 특징점을 추

출하기에는 부적합함을 알 수 있다. 따라서 Lü와 Wang의 세선화 결과를 수정, 보완하는 것이 필요하다. 분기점의 경우는 (그림 4)의 A에서 보는 바와 같이 분기점으로서의 특정 화소(pixel) 주변의 연결된 획의 기수가 3개 또는 4개인 경우이므로, 분기되는 부분 주위에 분기점으로 예상되는 특정 한 화소를 제외한 다른 화소는 3개 또는 4개 보다 많은 이웃 화소를 가지지 못하게끔 변환한다. 연결점의 경우는 연결점으로서의 특정 화소 주위에 연결되는 화소는 2개 보다 많은 수를 가지지 않으므로 (그림 4)의 B에서 보는 바와 같이 분기점으로서의 오 분류를 막기 위해 특정 화소 주변의 이웃 화소가 3개 이상의 이웃 화소를 가지지 못하게끔 변환한다[11]. (그림 4-b)는 변형된 Lü와 Wang의 세선화 방법으로 특징점 추출에 적합하게 골격선을 추출한 결과이다.

<표 1> 특징점의 특성

	특징	주변 화소개수
끝점	문자 획의 끝 부분으로 다른 획과의 접촉이 없는 부분	1
분기점	획과 획 사이의 연결 또는 교차 부분	3 또는 4
순환점	'o' 또는 '口'와 같이 자기 순환되는 영역의 획을 추출하기 위해 결정되는 부분	2
연결점	하나의 획을 구성하는 연결 부분	2



(그림 5) 특징점

<표 1>은 추출된 골격선에서 획을 추출하기 위한 특징점의 특성을 보이며 (그림 5)는 특징점을 추출한 결과를 보여준다. 끝점의 경우는 (그림 5)의 A 부분에서와 같이 문자 영역에서 하나의 획의 끝 부분으로 다른 획과의 접촉 및 교차가 없는 부분으로써, 주변 화소는 1개만을 가지게 된다. 분기점은 (그림 5)의 B 부분으로 획과 획 사이의 연결 또는 교차되는 부분으로써 주변 화소는 분기점에서 분기되는 획의 개수에 따라 변하는데, 하나의 분기점에서 3개 또는 4개의 획으로 분기되는 경우는 주변 화소는 각각 3개, 4개를 가진다. 순환점은 'o' 또는 '口'와 같이 자기 순환되는 영역의 획을 추출하기 위해 결정되는 부분으로 주변 화소는 2개를 가진다. 연결점은 하나의 획을 구성하는 연결 부분으로 주변 화소는 2개를 가지며 끝점, 분기점, 순환점을 제외한 모든 점들로 구성된다.

3. 제안된 획 추출 방법

특징점을 이용하여 세그먼트 화소열을 추적하여 추출된 세그먼트 화소열 각각을 최대 허용 오차법을 이용하여 백

터화 함으로써 세그먼트 벡터를 추출한다. 벡터화 시 하나의 세그먼트 화소열은 각각의 특성에 따라 둘 또는 그 이상의 세그먼트 벡터로의 분리가 이루어지며, 추출된 세그먼트 벡터 각각의 방향, 위치, 분기 여부 정보를 이용하여 둘 또는 그 이상의 세그먼트 벡터를 결합함으로써 세그먼트 벡터를 인식에 적합한 완전한 획으로 재구성한다.

3.1 세그먼트 추출

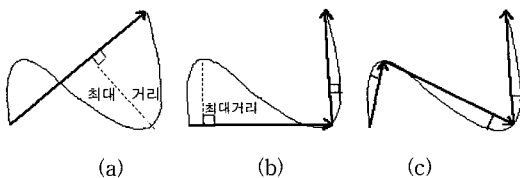
3.1.1 세그먼트 화소열 추적

추출된 끝점, 분기점, 순환점, 연결점을 이용하여 각각의 세그먼트 화소열을 추출한다. 먼저 끝점이나 분기점을 세그먼트의 시작점으로 하여 이웃 연결 화소를 반복적으로 추적하게 되는데 연결 화소가 끝점 또는 분기점일 경우 세그먼트의 끝점으로 결정하고 하나의 세그먼트 화소열을 저장한다. 이 때 추적 및 시작된 화소로서 끝점은 한번 사용될 수 있고, 3개의 획으로 분기되는 3분기점의 경우는 3번, 4분기점의 경우는 4번 사용되어질 수 있다. 이러한 방식으로 끝점, 분기점이 모두 사용되었을 경우는 추적되지 않은 연결점이 존재하는 경우는 순환점을 찾아 추적을 하여 순환되는 세그먼트를 저장한 후 종료된다.

3.1.2 벡터화

저장된 각각의 세그먼트 화소열을 최대 허용 오차범(maximum permission error method)을 이용하여 벡터화를 수행한다. 각각의 세그먼트 데이터의 점들을 추적하면서 세그먼트 벡터 데이터로의 변환을 수행한다. 벡터화를 위해 본 논문에서는 최대 허용 오차범[10]을 사용하였으며 그 과정은 다음과 같다.

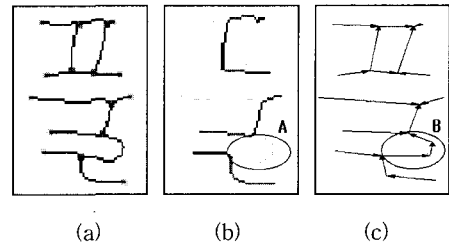
- 추출된 세그먼트 화소열의 시작점과 끝점을 연결하는 벡터를 생성한다.
- 세그먼트 화소열의 화소들 중 이 벡터와 가장 거리가 먼 화소를 선정하여, 그 거리가 임계값 V_{thres} 보다 크면, 그 지점에서 벡터를 두 개로 분리하고, 작거나 같으면 벡터를 근사화 결과로 간주한다.
- 생성된 두 벡터에 대해 과정 2)를 반복적으로 수행하여, 임계값보다 큰 거리가 존재하지 않을 때까지 계속한다.



(그림 6) 최대 허용 오차법의 과정

최대 허용 오차법의 과정을 (그림 6)에 보였다. 추출된 세그먼트 화소열을 벡터화를 적용함으로써 하나의 세그먼트로 구분된 획을 분리할 수 있다. 즉 'ㄱ'이나 'ㄴ'과 같은 자소는 하나의 세그먼트로 분류되게 되는데 벡터화를

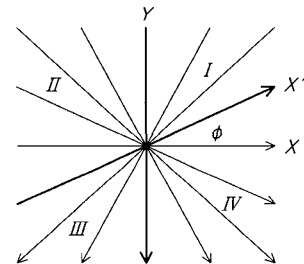
수행함으로써 하나의 세그먼트 화소열을 각각의 획에 적합한 세그먼트 벡터들로 분리 할 수 있다.



(그림 7) 세그먼트 추출 및 벡터화

(그림 7-a)는 추출된 특징점을 나타내며, (그림 7-b)는 골격선에서 특징점을 이용하여 각각의 세그먼트 화소열을 추적한 결과이며, (그림 7-c)는 각각의 세그먼트 화소열 데이터를 최대 허용 오차법을 이용하여 벡터화하여 세그먼트 벡터를 추출한 결과이다. (그림 7-b)의 A와 (그림 7-c)의 B 부분에서 보는바와 같이 하나의 세그먼트 화소열이 3개의 세그먼트 벡터로 분리가 적절히 이루어졌음을 알 수 있다.

3.2 세그먼트 벡터의 결합

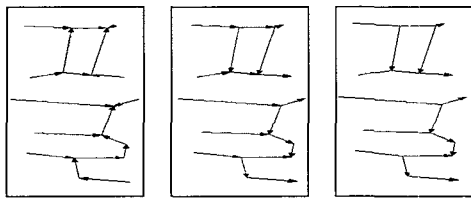


(그림 8) 오른손 필기 좌표계 시스템

필기체 한글 문자는 필기체 한자와 마찬가지로 사용되는 필기 좌표계 시스템은 일반적인 각도 시스템보다는 인간의 필기 방향을 고려하여 기울어진 각도 시스템을 사용하는 것이 더 적합함을 알 수 있다[1]. 오른손 필기 좌표계 시스템은 왼손 필기 방향 시스템과 다르다. 일반적으로 왼손 필기보다 오른손 필기가 월등히 많기 때문에 본 논문에서는 (그림 8)에서 보이는 바와 같이 X축에서 각도 ϕ 만큼 휘어진 X'축을 사용한 Chang과 Yan[1]의 오른손 필기 X'-Y 좌표계 시스템을 사용한다. 실험적으로 필기체 한글 문자에서의 각도 ϕ 는 $15^\circ \sim 25^\circ$ 정도가 좋은 결과를 보였다. 일반적으로 한자나 한글 문자의 경우 획의 방향은 위에서 아래로의 방향 또는 왼쪽에서 오른쪽으로의 방향으로 필기된다. 이러한 특성에 따라 (그림 8)에서의 I 또는 II사분면에서 획이 시작되고 III 또는 IV사분면에서 획이 종료되는 획의 필기 방향을 직관적으로 규정 할 수 있다. 벡터화된 세그먼트를 오른손 필기 좌표계 시스템에 따라 세그먼트 벡터의 방향 성분을 수정 보완하여 실제 필기시의 세그먼트의 필기 방향을 획득 할 수 있다. 또한 벡터화로 획득된 세그먼트 벡터들은 분기점으로 인해

하나의 획이 2개 또는 여러개의 세그먼트 벡터로 나뉘어진 경우가 다수 존재하기 때문에 획득된 세그먼트 벡터의 방향 정보, 위치 정보, 분기 여부를 이용하여 방향성이 같고, 같은 분기점에서 분기된 세그먼트 벡터들을 하나의 획으로 결합한다. 이 때 결합될 세그먼트 벡터의 특징은 다음과 같다. 1) 결합될 세그먼트 벡터의 방향성이 같아야한다. 2) 오차 범위는 두가지로 나뉘는데 먼저 첫 번째로 결합될 세그먼트 벡터들의 양 끝단이 특징점이 끝점일 경우는 25° , 분기점일 경우는 50° 로 설정하여 실험을 수행한 결과 좋은 결과를 보였다. 여러개의 획이 하나의 세그먼트로 오분류 된 경우는 이미 벡터화 과정에서 분리가 이루어졌다.

(그림 9)는 획 추출의 과정으로 (그림 9-a)의 추출된 세그먼트 벡터는 인간의 필기 방향과 무관하게 세그먼트의 추출 시의 방향에 따라 아무런 방향 정보가 없는 벡터이다. (그림 9-b)의 결과는 오른손 필기 좌표계 시스템에 의해 세그먼트 벡터의 방향 정보를 수정, 보완한 결과로써 총 18개의 세그먼트 벡터 가운데 10개의 세그먼트 벡터의 방향 정보가 수정되었다. 또한 (그림 9-c)의 결과는 추출된 세그먼트 벡터의 방향성과 분기 여부 정보를 이용하여 연결 가능한 2개 이상의 세그먼트 벡터를 하나의 완전한 획으로 재결합한 결과로서 총 18개의 세그먼트 벡터를 이용하여 최종적으로 문자 인식을 위한 11개의 모든 획이 추출된 결과이다.



(a) (b) (c)

(그림 9) 획 추출

4. 실험 결과 및 검토

실험에 사용된 샘플은 KAIST에서 제작한 Korea National Language Information Base(대한민국 국어 정보 베이스) CD-ROM에 수록된 한국어 필기체 데이터 베이스로써, 데이터 수집 용지는 갱지이며 필기 도구는 사인펜인 정서체의 양호한 품질의 오른손 필기 데이터를 사용하여 제안



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

(그림 10) 획 추출의 전과정

된 획 추출 알고리즘을 적용하였다.

<표 2> 획 추출 과정

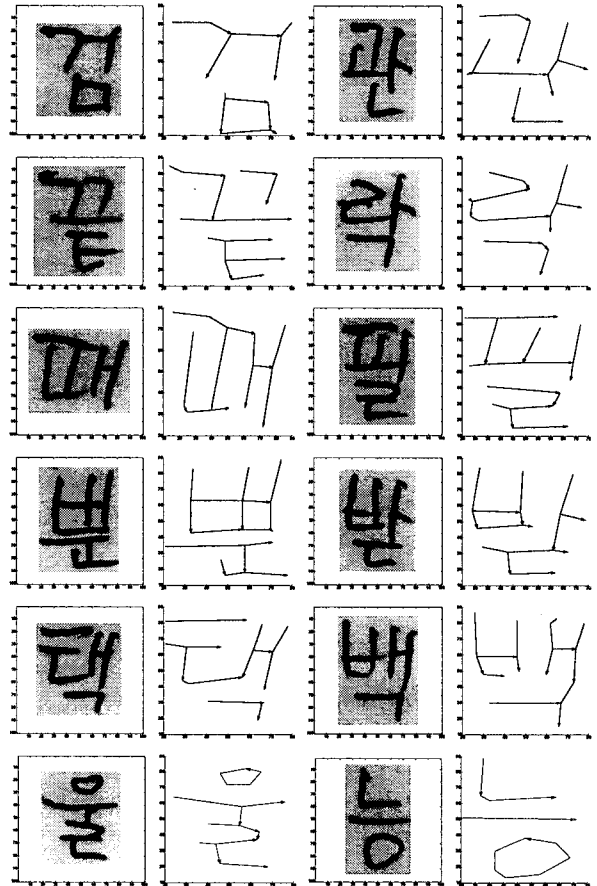
	설 명
a	문자 래스터 영상
b	Watershed변환 결과
c	문자부 분리 결과
d	Lü와 Wang의 세선화 결과
e	변형된 세선화 결과
f	특징점 추출 결과
g	세그먼트 화소열 추적 결과
h	벡터화 결과
i	벡터의 방향성 수정 결과
j	연결 가능한 벡터 결합 결과

(그림 10)에 필기체 문자 영상(KNLIB Kle10001의 44번째 문자 '철'과 '약')을 본 논문에서 제안한 획 추출 방법으로 획을 추출한 결과를 나열하였으며, <표 2>는 각각의 과정을 간략히 나타내었다. 획 추출 알고리즘의 요약은 다음과 같다. 1) (그림 10-a)의 필기체 문자 래스터 영상을 Watershed 알고리즘을 적절히 사용하여 Watershed 영상(그림 10-b)을 추출하고, 분할 함수를 적용하여 필기체 문자 인식에 적합하게 이진화를 통해 문자부(그림 10-c)를 분리한다. 2) 분리된 문자 영역을 Lü와 Wang의 세선화 방법을 적용하여 골격화(그림 10-d)를 수행하고 특징점 추출에 적합하게 변형(그림 10-e)한다. 추출된 골격선의 특징을 이용하여 끝점, 분기점, 연결점, 순환점을 각각 추출(그림 10-f)한다. 추출된 특징점을 이용하여 세그먼트 화소열을 추적(그림 10-g)한다. 추출된 세그먼트 화소열들을 각각 최대 허용 오차법을 이용하여 벡터화(그림 10-h)를 수행하고 추출된 세그먼트 벡터의 방향 성분을 오른손 좌표 시스템을 이용하여 수정, 보완(그림 10-i)한다. 또한 세그먼트 벡터들의 위치, 분기 여부의 정보를 이용하여 연결 가능한 세그먼트 벡터들을 완전한 하나의 획으로 결합(그림 10-j)함으로써 필기체 문자에서 인식을 위한 동적인 정보를 포함하는 획을 추출한다. 또 다른 필기체 한글 문자의 획 추출 결과를 (그림 11)에 보였다.

추출된 획은 문자 입력 시 오류 등으로 인해 문자에서의 자음부와 모음부의 접합 및 교차로 완전한 획을 추출하기 부적절한 자유로운 필기체의 경우 획의 추출 시 다소 오류가 있음을 알 수 있었다. 그러나 이러한 오류는 인식을 위한 자음부와 모음부의 분리 시에 추출된 획의 정보 뿐 아니라 하나의 획으로 병합되기 전의 세그먼트 벡터 정보 또한 같이 이용한다면 이러한 오류는 해결할 수 있을 것으로 기대된다.

본 논문의 효용성을 높이기 위해서 향후 연구 과제로는 효과적인 문자 인식을 위해 추출된 획의 방향 성분, 분기 여부, 위치 등의 여러 가지 정보를 이용하여 한글 문자에서

의 자음부와 모음부의 분리에 대한 연구, 분리된 자음부와 모음부의 인식 그리고 인식된 문자의 오인식된 결과의 보정을 위한 자연어 처리 측면의 후처리에 대한 연구가 수행되어야 할 것이다.



(그림 11) 획 추출 결과

5. 결 론

오프라인 필기체 한글 문자 인식의 경우 문자의 필기시의 정보로서 온라인 필기체 문자 인식에서와 같은 동적인 정보를 적절히 추출하기 위해서 문자에 있어서의 획을 추출하는 방법에 대한 연구가 필요하다. 본 연구에서는 온라인 필기체 문자 인식 시 입력되는 여러 가지 동적인 정보 가운데 인식 시 중요한 정보인 획과 획의 방향성을 오프라인 필기체 한글 문자에서 추출하기 위해 Watershed 알고리즘을 이용하여 그레이 영상을 이진화하여 문자부와 배경간의 분리를 수행함으로써 필기시의 용지 및 필기 도구의 특성에 적합하게 분리를 수행하였고, 획 추출에 적합한 변형된 세선화 방법을 통해 문자에서의 특징점을 추출하여 세그먼트를 추적함으로써 문자의 분기 및 교차 특성에 적합하게 획을 구성하는 세그먼트 화소열을 추출할 수 있었다. 추출된 세그먼트 화소열들을 최대 허용 오차법을 이용하여 벡터화를 수행함으로써 여러개의 획이 하나의 세그먼트 화

소열로 병합된 부분의 적절한 분리가 가능하였으며, 추출된 벡터의 위치, 오른손 필기 좌표계 시스템을 이용한 인간의 필기 방향성을 이용한 수정된 방향 성분을 적절히 이용하여 교차점이 분기점으로 오분류된 경우로 인한 분리된 획을 하나의 완전한 획으로 병합함으로써 필기체 한글 문자 인식에 적합한 획을 추출하였다.

참 고 문 헌

[1] H. Chang and H. Yan, "Analysis of Stroke Structure of Handwritten Chinese Character," IEEE Transactions on Systems, Man, and Cybernetics, Part B : Cybernetics, Vol. 29, No.1, pp.47-61, Feb., 1999.

[2] 최경주, 변혜란, 이일병, "효과적인 이진화를 위한 영상 개선 기법의 정의 및 구현", 정보과학회논문지, Vol.26, 제2호, pp.284-296, 1999.

[3] D. S. Doermann, and A. Rosenfeld, "Recovery of Temporal Information from Static Images of Handwriting," Proc. CVPR'92, pp.162-168, 1992.

[4] C. M. Provitera, and R. Plamondon, "A System for Scanning and Segmenting Cursively Handwritten Words into Basic Strokes," Proc. 3rd ICDAR'95, pp.1047-1050, 1995.

[5] J. Serra and L. Vincent, "Lecture Notes in Mathematical Morphology," Ecole Nationale Supérieure des Mines de Paris, France. 1989.

[6] P. Soille and L. Vincent, "Determining Watershed in Digital Pictures via Flooding Simulations," In Visual Communications and Image Processing'90, Vol.SPIE-1360, 1990.

[7] Michel Couprie and Gilles Bertrand, "Topological Grayscale Watershed Transformation," In SPIE Vision Geometry V Proceedings, Vol.3168, pp.136-146, 1997.

[8] Jos B. T. M Roerdink and Arnold Meijster, "The Watershed Transform : Definitions, Algorithms and Parallelization Strategies," Institut for Mathematics and Computing Science, Report IWI 99-9-06, 1999.

[9] H. E. Lü and P. S. P. Wang, "A Comment on 'A Fast Parallel Algorithm for Thinning Digital Patterns'," Commun. ACM Vol.29, No.3, pp.239-242, 1986.

[10] K. Wall, and P. E. Danielsson, "A Fast Sequential Method for Polygonal Approximation of Digitized Curves," CVGIP 28, pp.220-227, 1984.

[11] Y. K. Choi and S. B. Rhee, "Robust Stroke Extraction Method for Handwritten Korean Characters," Proceedings of ITC-CSCC, Vol.2, pp.819-822, 2000.

[12] R. C. Gonzalez, and R. E. Woods, "Digital Image Processing," Addison-Wesley, 1992.



최 영 규

e-mail : young@dankook.ac.kr

1994년 단국대학교 전자공학과 졸업
(공학사)

1997년 단국대학교 대학원 전자공학과
졸업(공학석사)

2001년~현재 단국대학교 대학원 전자공
학과(공학박사수료)

2000년~2001년 (주)페스씨인

관심분야 : 패턴인식, 인공지능, 멀티미디어응용



이 상 범

e-mail : sbrhee@dankook.ac.kr

1974년 연세대학교 전자공학과(공학사)

1978년 서울대학교 대학원 전자공학과
(공학석사)

1986년 연세대학교 대학원 전자공학과
(공학박사)

1984년 미국 IOWA대학교 컴퓨터공학과 객원교수

1979년~1999년 단국대학교 전자·컴퓨터공학과 교수

1997년~1999년 단국대학교 교무·연구처장

1997년~현재 단국대학교 멀티미디어산업기술연구소장

2000년~현재 단국대학교 공학부 컴퓨터공학전공 교수

관심분야 : 컴퓨터구조, 패턴인식, 디지털 신호처리