손으로 설계한 서식 문서의 주요점 검출 및 서식 구조 벡터화

김병용*
*상지영서대학 전자계산과
e-mail : bykim@cs.youngseo.ac.kr

Main Points Extraction and Layout Vectorization of Hand-designed Forms

Byeong-Yong Kim*
*Dept. of Computer Science, Sangji Youngseo College

요 약

본 논문은 손으로 자유롭게 그린 서식 문서의 주요점을 검출하여 서식의 구조를 벡터화하는 방법을 제안한다. 선 성분의 주요점을 검출하여 그 구조를 벡터화하는 방법은 주로 인쇄 서식 문서의 구조 분석에 적용하기 좋은 방법이다. 이에 반해 손으로 설계한 서식 문서는 주요점 부분이 왜곡되어 있기 때문에 주요점의 검출이 손쉽게 이루어지기 곤란하다. 이 논문에서는 이러한 문제를 해결하기 위해 손으로 설계한 서식 문서를 세선화한 다음 여유 성분을 갖는 마스크를 적용하고 후처리를 통해 주요점 부분의 심한 왜곡을 보상하는 방법을 제안하여 손으로 설계한 서식 문서에서도 주요점의 검출이 가능하도록 하였다. 제안한 방법의 유효성을 확인하기 위한 실험 결과 손으로 설계한 서식의 경우 91.9%, 인쇄 서식의 경우 100%의 벡터화 성공률을 보여주어 제안한 방법이 손으로 설계한 서식 구조의 벡터화에 유효함을 확인하였다.

1. 서론

문자 인식 및 문서 처리에 대한 연구는 크게 온라인 필기체 인식과 오프라인 인쇄체 인식의 두 가지 분야로 나누어져 수행되어 왔다[1]. 이 두 가지 분야의 연구 성과는 개인 휴대용 단말기(PDA)의 입력 수단과 상용 문자 인식 패키지에 적용되고 있으며 문자 인식 자체에 대한 연구는 상당한 수준까지 진척되어 있다. 그러나 필기체 오프라인 인식에 관한 연구는 과거 수 십 년간 연구되어 왔음에도 불구하고 그 문제의 복잡성으로 인하여 아직 상용화되지 못하고 있다. 더욱이 전자 문서가 보편화 되었음에도 불구하고 인쇄 문서의 사용이 날로 증가하고 있는 현시점에서 온라인 필기체 및 오프라인 필기체의 인식에 대한 연구의 필요성이 대두되고 있다.

오프라인 문자 인식 분야의 연구 동향은 인쇄체 인식 단계를 벗어나 현재는 오프라인 필기체 인식에 관한 연구가 주를 이루고 있다. 오프라인 인쇄체 인식의 발전 단계가 먼저 문자 인식에 초점을 맞추어 연구하다가 문자 인식의 성과가 어느 정도 달성된 시점에서는 문서 처리에 대한 연구가 활성화 되는 단계로 발전해 왔다. 이에 따라 오프라인 필기체 인식도 현재는 필기 숫자 및 문자 인식에 그 연구의 초점이 맞추어져 있지만 그와 더불어 인식 결과를 실용화하기 위해서는 필기 문서의 처리에 대한 연구도 병행해야 한다.

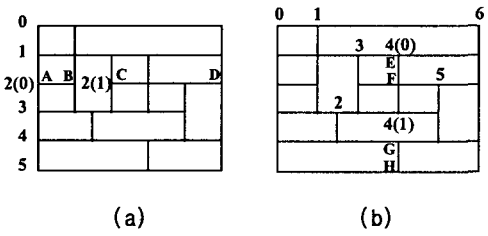
문서 처리에 대한 연구는 일반적인 문서 영상의 경우 처리해야 할 데이터량이 많고 종류도 다양하며 형태가 수시로 변하기 때문에 문서의 구조를 자동으로 분석할 수 있는 시스템을 만드는 것이 매우 어렵다는 것을 깨달아 인쇄체 문서의 경우도 주로 문서의 형태가 일정한 서식을 갖추고 있는 문서, 즉 일정한 형식의 논문, 우편물, 수표나 전표 등과 같은 서식 문서 영상의 구조 분석에 관한 연구가 활발히 진행되어 왔었다[2][3].

* 이 논문은 2000 년도 상지영서대학 학술연구비 지원에 의하여 연구된 것임.

더욱이 손으로 작성한 문서의 경우는 인쇄된 문서보다도 더욱 더 형태가 다양하기 때문에 모든 문서의 구조를 자동으로 분석하는 것은 매우 난해할 수 밖에 없다. 따라서 본 논문에서는 각종 서식의 설계와 웹 페이지의 레이아웃 설계, 각종 하드 카피의 구조 설계 및 워드 프로세서 등과의 결합을 통해 수험표, 이름표, 상품에 붙이는 레이블 등 각종 표를 보다 손쉽게 설계할 수 있도록 하기 위해 주요점 검출을 통해 손으로 설계한 서식 문서의 문자 분리 및 서식 구조의 벡터화 기법을 제안하고자 한다.

2. 선 성분 주요점 이용법의 개요

선 성분 주요점 이용법은 인쇄 서식 문서의 구조 분석에 자주 사용되는 방법이다. J. LIU 등은 서식의 구조를 정의하기 위해 서식의 구조를 이루는 선 성분을 수평 그룹(H-group)과 수직 그룹(V-group)으로 나눈 다음 각 그룹의 선 성분에 일정한 순서를 부여함으로써 서식의 구조를 나타내는 Frame Template 을 정의하였다[4]. 그림 1 은 Frame Template 의 예로서 그림 1(a)와 그림 1(b)는 각각 Frame Template 을 구성하는 수평 그룹과 수직 그룹의 선 성분 순서를 나타낸 것이다. 그림 1(a)는 최상단 수평선을 기준으로 각 선의 상대적인 순서를 나타낸 것이며, 그림 1(b)는 최좌측 수직선을 기준으로 각 수직선의 상대적인 순서를 나타낸 것이다. 그림 1(a)의 선 성분 A-B, C-D 와 그림 1(b)의 선 성분 E-F, G-H 는 각각 같은 순서를 갖는 선 성분으로 이들은 각 순서 내에서 부그룹(subgroup)을 이루고 각 부그룹 내에서의 순서는 괄호 안에 표시하였다.



(a) 수평 그룹 순서, (b) 수직 그룹 순서
그림 1. Frame Template

서식의 구조를 알아내기 위해 선 성분의 주요점을 이용하는 방법은 주로 선 성분의 끝점, 모서리점, 교차점을 이용하게 된다. 그러나 사각형 모양의 외곽선을 갖는 서식의 경우는 독립된 끝점이 존재하지 않기 때문에 모서리점과 교차점을 검출함으로써 서식의 구조를 알아낼 수 있게 된다. 이경우 모서리점과 교차점은 그림 2 와 같이 모두 9 가지가 존재하게 된다. 이들 중 서식의 외곽선을 알아내기 위해서는 TL, TR, BL, BR 로 표시된 4 개의 모서리점이 필요하게 되며, 수평선과 수직선을 알아내기 위해서는 각각 L, R, T, B 로 표시된 4 개의 교차점이 필요하게 된다. 따라서 총 8 개의 주요점을 서식의 구조 분석에 사용하게 된다.

일반적으로 서식내의 선 성분은 충분히 길게 작성되는 것이 보통이므로 인쇄 서식의 경우 투영을 통해 선 성분의 위치를 추정해 내는 방법이 사용되기도 한다. 그러나 투영 결과는 서식 영상의 기울어짐에 민감하게 반응하게 되며, 손으로 설계한 서식 문서 영상과 같이 선 성분이 일정한 방향으로 작성되지 않은 영상에는 적용하기 곤란한 단점이 있다. 따라서 인쇄 서식 문서 영상과 손으로 설계한 서식 문서 영상에 모두 적용하기 위한 방법으로는, 영상을 일단 세선화한 후 미리 정의된 마스크(mask) 를 적용하여 각 선 성분의 주요점을 찾아내는 것이다. 본 논문에서는 마스크로 25 x 25 픽셀의 행렬을 사용하였다.

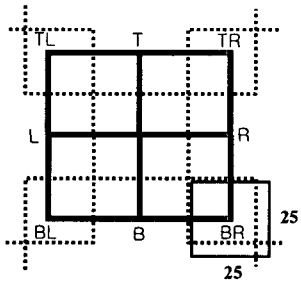
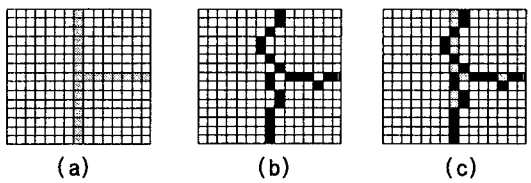


그림 2. 서식의 구조 분석을 위한 주요점

3. 마스크 적용의 문제

세선화한 형태로 작성한 마스크를 세선화한 영상에 적용하면 세선화 과정에서의 영상 왜곡으로 인하여 부작용이 발생하게 된다. 따라서 마스크 작성시 일정한 정도의 여유를 주어야 하는데 이러한 여유는 영상의 한 부분에서 동일한 주요점이 여러 개 검출되는 부작용을 초래한다.

그림 3 은 세선화한 영상에 선 성분의 폭이 1 인 마스크를 적용할 때의 부작용을 보여준다. 그림 3(a)의 마스크를 그림 3(b)와 같이 왜곡된 교차점에 적용하는 경우 결과는 그림 3(c)와 같아진다. 그림 3(c)에서 마스크와 일치하는 흑화소의 개수는 13 개가 된다. 그러나 같은 마스크를 왜곡되지 않은 완전한 직선 성분에 적용하는 경우 일치하는 흑화소의 개수가 15 개가 되어 오히려 직선 부분에서 더 큰 일치성을 보이게 된다. 따라서 이러한 마스크로는 원하는 교차점을 찾을 수 없게 된다.

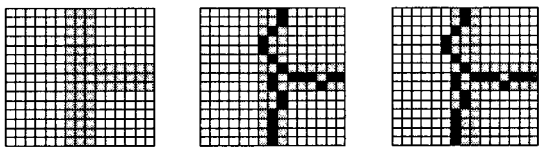


(a) 마스크, (b) 세선화 한 부분 영상, (c) 마스크 적용 결과
그림 3. 마스크 적용의 부작용

그림 3 과 같은 부작용을 해결하는 한 방법으로 마스크에 여유 성분을 부여하는 것이 있다. 이러한 여유 성분은 마스크 내에 존재하는 선 성분의 폭을 일정하게 넓힘으로써 부여될 수 있는데 그 예는 그림 4(a)와 같다. 그림 4(a)와 같이 선 성분의 폭이 1 이 아닌 마스크를 사용하여 각 교차점(LRTB)를 검출하는 방법은 아래의 식(1)과 같다.

$$LRTB = \{A_i \mid A_i = (\sum B_{fj} \times B_{mj} \geq \alpha_1) \text{ AND } (A_i \text{ 내 BLACK 화소수} \leq \alpha_2)\} \quad (1)$$

단, A_i: 마스크가 적용된 부분 영상,
B_{fj}: 부분 영상의 각 화소,
B_{mj}: 마스크의 각 화소,
α_1, α_2: 임계값, i = 1, 2, ..., n, j = 1, 2, ..., m



(a) (b) (c)

(a) 여유 성분 마스크, (b) 마스크 적용 결과 1, (c) 마스크 적용 결과 2

그림 4. 여유 성분을 갖는 마스크의 적용

식(1)의 $\alpha 1$ 은 마스크와 부분 영상의 일치 여부를 결정하는 임계값이며, $\alpha 2$ 는 수직선과 수평선이 교차한 + 모양의 교차점을 LRTB 의 집합에서 제외하기 위한 임계값이다. 영상이 완벽하다고 가정하는 경우 이 두 임계값이 동일한 값을 갖게 되나 실제 적용시에는 잡영을 고려하여 다른 값을 사용하게 된다. 25 x 25 픽셀 마스크의 경우 $\alpha 1$, $\alpha 2$ 의 이상 적인 값은 37 이 되나 실제 적용시의 $\alpha 1$ 값은 이보다 약간 작은 값을 사용하게 되며 $\alpha 2$ 는 더 큰 값을 사용하게 된다. 본 실험에서는 경험적인 방법으로 각각 30 과 44 를 사용하였다.

4. 처리 절차

그림 4(a)는 15 x 15 픽셀 마스크로 이상적인 영상의 α_1, α_2 값은 22 가 된다. 그런데 영상의 주요점을 찾기 위해 마스크를 영상의 상하, 좌우 순서로 스캔하며 적용하는 경우, 동일한 부분 영상에 대해 그림 4(b) 및 그림 4(c)와 같은 경우가 연속해서 발생하게 된다. 이 경우 그림 4(b)의 흑화소 일치값은 21 이며, 그림 4(c)의 경우는 20 이 된다. 따라서 실제 적용에 있어 α_1 은 22 보다 작은 값을 사용하게 되고, α_2 는 좀더 큰 값을 사용하기 때문에 이들 두 경우가 모두 교차점 L 로 인식된다. 이것은 마스크에 여유 성분을 부여했기 때문인데 선폭이 3 인 마스크를 사용하고 영상을 순차적으로 주사하는 경우, 한 교차점을 최대 9 개의 교차점으로 인식할 수 있다. 이를 확인하기 위하여 실제 실험해본 결과 한 개의 교차점을 대부분 2-3 개의 교차점으로 인식하였다.

이러한 문제점을 해결하는 방법으로는 가장 먼저 검출된 위치를 리스트에 보관하고 교차점이 검출될 때마다 리스트를 검색하여 이미 검출된 교차점의 좌우 3 픽셀 이내에서 검출된 동일한 교차점은 기각하는 방법을 사용할 수 있다. 그러나 이 경우 매번 교차점이 검출될 때마다 리스트를 검색해야 하는 부담이 증가하게 된다. 따라서 본 실험에서는 일단 모든 교차점을 검출하고 후처리를 통해 중복되는 검출 결과를 삭제하는 방법을 사용하였다.

본 논문에서 사용한 선 성분 주요점 검출 및 서식 구조 벡터화의 전체 절차는 다음과 같다.

단계 1, 영상에 연결 요소 분석법을 적용하여 문자 영역과 선 성분을 분리한다. 인쇄 서식의 경우 연결 요소 분석법에 의해 문자 영역의 완전한 분리가 가능하며, 손으로 설계한 서식의 경우도 대부분의 문자 영역이 이 단계에서 분리되는 것을 전제로 한다.

단계 2, 단계 1에서 분리된 선 성분 영상의 네 모서리를 시작점으로 하여 안쪽 방향으로 스캔하면서 서식 외각 사각형의 네 모서리점 TL, TR, BL, BR을 찾는다.

단계 3, 영상의 좌측 상단을 시작점으로 좌우-상하 방향으로 스캔하면서 영상 내에 존재하는 모든 L, R, T, B를 찾아낸다. 이때 각 교차점의 종류와 좌표를 저장하는 리스트 LRTB를 만들고 교차점이 찾아질 때마다 해당 노드를 리스트 LRTB에 추가한다.

단계 4, LRTB 리스트에서 동일한 교차점이 중복 검출된 것을 찾아내어 삭제한다. 그러면 모든 교차점이 LRTB 리스트에서 유일하게 된다. 그런 다음 LRTB 리스트를 좌표에 따라 정렬한다. 이때 각 노드가 정렬되는 순서는 다음과 같다.

(가) 1차로 수평 선 성분(교차점 L 및 R)과 수직 선 성분(교차점 T 및 B)을 구분한다.

(나) 수평 선 성분 내의 교차점을 그 위치에 따라 상-하, 좌-우의 순서로 정렬한다.

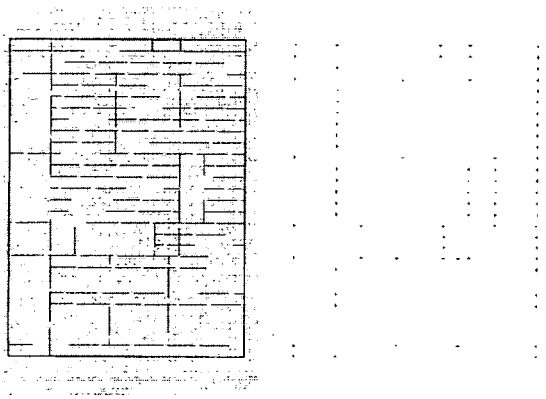
(다) 수직 선 성분 내의 교차점을 그 위치에 따라 좌-우, 상-하의 순서로 정렬한다.

단계 5, 정렬된 LRTB 리스트를 처음부터 검색하면서 L 과 R 및 T 와 B 의 쌍을 서로 연결하여 각 선 성분의 벡터를 계산하고 선 성분 리스트를 만든 다. 그리고 단계 2 에서 찾아진 네 개의 모서리점을 참조하여 서식의 외곽 상자를 만든다.

단계 6, 서식의 외곽 상자를 포함하여 LRTB 리스트로부터 만들어진 선 성분을 위치에 따라 정렬한다. 정렬 순서는 우선 수평선 그룹과 수직선 그룹의 두 그룹으로 나누고, 각 그룹 내의 선 성분을 서식의 좌측 상단으로부터의 거리에 따라 정렬한다.

단계 7, 외곽 상자를 중심으로 정렬된 순서에 따라 각 선을 확장하거나 축소하면서 서식의 형태를 만들어 간다. 이때 각 선 성분의 이웃을 참조하여 각 선의 정확한 길이를 추정해 낸다. 각 선의 정확한 길이가 찾아졌으면 이들의 벡터를 파일에 저장한다.

그림 5 는 실험에 사용한 서식 중에서 문자 영역을 분리하고 세선화한 서식과 거기서 찾아낸 선 성분의 주요점을 나타낸 것이다. 그림 5 에서 보는 것과 같이 TL, TR, BL, BR 및 L, R, T, B 의 모든 교차점이 검출되었음을 알 수 있다.



(a) (b)

(a) 세선화한 서식, (b) 선 성분의 주요점
그림 5. 세선화한 서식과 선 성분의 주요점

5. 실험 및 결과

제안 방법의 유효성을 확인하기 위한 실험은 A4 용지 크기의 서식 2 개를 대상으로 실험하였다. 각각의 서식에 대해 한 개는 인쇄된 서식을 대상으로 하고, 다른 하나는 같은 형태의 서식을 손으로 설계한 다음 실험하였다.

실험 환경의 하드웨어는 펜티움 III 700MHz, 256MB 의 IBM PC 이고, 운영체제는 Windows98 이며 프로그래밍 언어는 Microsoft 사의 Visual C++ 6.0 을 사용하여 구현하였다. 서식 입력을 위한 스캐너는 Sharp 사의 JX-330P 를 사용하여 모든 서식을 그레이스케일 (grayscale) 300DPI 의 해상도로 입력하였다.

서식 영상이 그레이스케일로 입력되었기 때문에 주요점 검출 전에 해당 영상을 이진화 및 세선화[5] 한 다음 제안 방법을 적용하였다.

[표 1] 벡터화 실험 결과표

실험 구분	서식작성		영상 크기	벡터화 결과				수행 시간 (초)	비고
				수평 선 성분		수직 선 성분			
	인쇄 여부	손설계 여부		서식내 벡터수	검출 벡터수	서식내 벡터수	검출 벡터수		
실험 1	o	x	2340x2976	20	20	24	24	11.30	서식 1
실험 2	x	o	2172x3096	20	18	24	21	10.83	
실험 3	o	x	1944x2820	30	30	12	12	6.87	서식 2
실험 4	x	o	2004x2784	30	29	12	11	8.79	

[표 1]의 실험 결과표를 보면 인쇄 서식 중 서식 1 은 44 개, 서식 2 의 42 개의 벡터를 모두 검출하여 100%의 벡터화 비율을 나타냈으며, 손으로 설계한 서식의 경우는 서식 1 은 44 개의 벡터 중 39 개, 서식 2 는 42 개의 벡터 중 40 개의 벡터를 검출하여 도합 86 개의 벡터 중 79 개를 검출하여 91.9%의 벡터화 비율을 보였다. 처리 속도는 영상의 크기가 약 7MB 정도가 되나 평균 처리 시간이 9.45 초로 비교적 빠를 처리 속도를 보여 주었다.

6. 결론

손으로 설계한 서식을 자동으로 처리하기 위해 선 성분의 주요점을 검출하여 그 구조를 벡터화하는 방법을 제안하였다. 제안 방법은 손으로 설계하였기 때문에 왜곡된 서식의 주요점을 검출하기 위해 여유 성분을 가진 마스크를 적용하여 모든 후보점을 검출한 다음 후처리를 통해 주요점의 왜곡을 보상하고 해당 벡터를 획득하는 방법을 사용하였다.

제안 방법의 유효성을 확인하기 위한 실험을 실시한 결과 손으로 설계한 서식에서는 91.9%, 인쇄 서식에서는 100%의 벡터화 성공률을 보여 제안한 방법이 손으로 설계한 서식 구조의 벡터화에 유효함을 확인하였다. 향후 연구 과제로는 서식의 전처리를 통해 왜곡 성분을 보상함으로써 인쇄 서식과 비슷한 벡터화 성공률을 획득하는 방법을 구안하는 일이다.

참고문헌

[1] 이성환, 문자인식: 이론과 실제 I, II 권, 홍릉과학출판사, 1994.

[2] 김병용, 권오석, 손으로 설계한 서식 문서의 문자 영역 분리 및 서식 벡터화, 한국정보처리학회 논문지, 제 7 권, 제 10 호, 2000.

[3] 김병용, 손으로 설계한 서식 문서의 문자 영역 분리 및 서식 구조 벡터화, 박사학위논문, 충남대학교, 2001.

[4] J. Liu, X. Ding, and Y. Wu, "Description and Recognition of Form and Automated Data Entry", Proceedings of the 3rd International Conference on Document Analysis and Recognition, pp. 579-582, Montreal, Canada, 1995.

[5] F. F. Chang, Y. P. Cheng, T. Pavlidis, and T. Y. Shuai, "A Line Sweep Thinning Algorithm", Proceedings of 3rd International Conference on Document Analysis and Recognition, pp. 227-230, Montreal, Canada, 1995.