형식문서에서 지정된 셀내의 문자추출 및 복원

심상옥, 유진용, 김민기, 권영빈
중앙대학교 컴퓨터공학과

Character Extraction and Restoration in the Specified Cell of Form Document

Sang-Ok Sim, Jin-Yong Yoo, Min-Ki Kim, Young-Bin Kwon
Dept. of Computer Science and Engineering, Chungang University

요 약

세금계산서나 영수증등의 형식문서를 처리하기 위해서는 일반문서와는 달리 형식문서에서 인식의 대상이 되는 특정 셀에 대한 추출이 필요하다. 본 논문에서는 정형화된 형식문서에서 원하는 특정 셀의 내용만을 추출하는 방법을 제시하고자 한다. 제안된 방법은 지정된 셀을 이루고 있는 라인을 제거하는 것과, 라인제거시 손상된 문자를 복원하는 과정으로 나뉜다. 우선 라인들의 평균적인 두께를 구한 후 라인을 트레이스(trace)하면서 이 두께 범위내에 있는 라인은 지운다. 트레이스하는 과정에서 두께보다 큰 라인은 문자와 접촉된 것으로 판단하여 이 접촉된 좌표를 저장한 후 미리 정의된 접촉유형을 이용하여 문자의 복원 작업을 수행한다.

1. 서론

일정한 양식을 갖는 형식문서에 사람의 손으로 쓰여진 필기체 문자의 전산처리를 위해서는 컴퓨터를 이용한 문자인식의 과정이 필요하게 된다. 이러한 인식의 전단계로 형식문서에서 인식을 하고자 원하는 위치에 있는 셀의 내용만 추출하는 과정이 필요하게 된다. 셀의 내용을 추출 할 때는 라인이외의 문자들만을 분리해내야만 정확한 문자인식의 수행이 가능하므로[1], 문자이외의 셀을 구성하는 라인들은 제거되어야 한다. 즉, 그 셀을 구성하는 4개의 라인과 그 라인들에 연결된 다른 셀을 이루는 라인들의 제거가 필수적이다.

그러나 이러한 형식문서상에 기록된 오프라인 필기체 문자들은 형식문서의 라인들과 겹쳐질 수가 있는데, 이런 문자들은 라인의 단순한 제거만 수행한다면 같이 지워지므로, 이런 끊어진 문자들의 복원과정이 필요하게 된다[2,3,4]. 그러므로 본 논문에서는 아직까지 분석이 미진한 한글 양식문서에 대하여 라인과 겹쳐진 문자에 대한 형태의 분석과 분류를 바탕으로 라인의 제거와 이로 인해 끊어지는 문자의 복원을 위한 알고리즘[5, 6]을 제시하고 구현된 결과를 보여 향후 문서복원을 통한 필기체 문자인식에 응용될 수 있도록 한다.

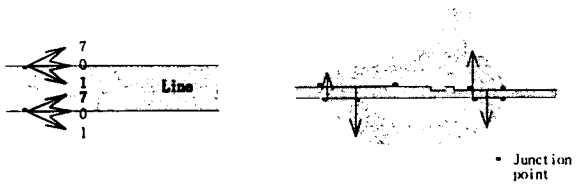
또한 입력문서들은 자동급지 같은 방법을 사용하여 입력이 행해지므로 약간의 기울어짐이 발생할 수 있다. 이 문제는 기울어진 문서를 보정(skew correction)한 후 처리하므로써 해결할 수 있다[7]. 그러나 기울어진 문서를 복원하는 것은 영상의 왜곡을 발생시키며 또한 많은 처리 시간을 요한다. 그러므로 기울어짐이 미세할 경우 기울어진 문서를 곧바로 처리할 수 있는 방법이 효과적이다.

본 논문의 구성은 크게 셀을 구성하고 있는 라인을 제거하는 단계, 라인제거시 끊어진 문자의 복원 단계, 실험과 결론으로 구성되어 있다.

2. 라인의 제거

2.1 셀을 이루는 주요 4개 라인의 제거

필드 추출을 위해서는 원하는 셀을 표현하는 4개의 모서리 좌표(셀의 모양은 모두 직사각형)가 주어져야 한다. 4개의 좌표가 주어지게 되면 이 좌표를 이용해 셀을 이루는 4개의 주요라인을 트레이스(trace)할 수가 있다. 라인의 트레이스는 자동급지시 생길수 있는 기울어진 양식도 처리를 가능하게 하기 위해서 4개의 라인간의 접합점(junction point)을 찾아 이를 이용해 직선의 방정식을 구해서 수행했다. 보통 라인은 실제로는 두께가 수 픽셀 이상이 되기 때문에 한 라인에 대해 윗면과 아랫면으로 나누어서 트레이스를 하게 했다. 트레이스하는 중 윗면의 경우에는 윗 방향으로의 런길이(run length)를 조사하면서[8] 라인의 두께에 비교해 어떤 임계치 값 이상이 되면 그 점을 저장하고 라인에 해당되는 부분은 지운다. 여기에서 저장되는 점이 문자와 라인이 겹쳐진 경우 복원에 이용할 접합점이 된다. 아래 <그림 1>에 접합점의 예를 보이고 있다.

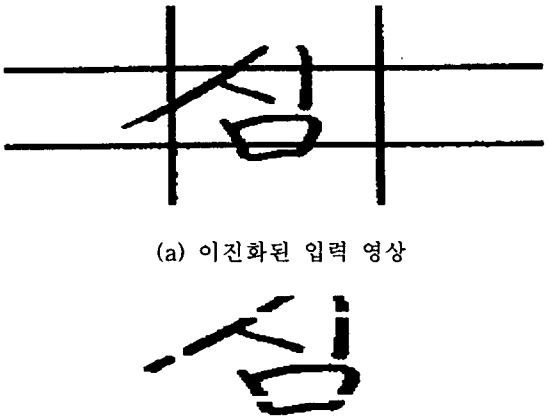


<그림 1> 라인의 트레이스와 접합점 결정

위와 반대로 아랫면을 트레이스하는 경우에는 아랫방향으로의 런길이를 계속 조사해가면서 임계치 이상의 값이 되면 위와 마찬가지로 접합점을 저장하고 라인은 지운다. 이렇게 해서 4개의 라인이 지워진 결과가 <그림 2>에 나타나있다.

2.2 외부 라인의 제거

형식문서는 여러 가지 복잡한 형태의 셀 구성을 가질 수 있다. 따라서 셀에 연결된 라인들의 모양도 여러 가지가 가능하게 된다. 하나의 셀 안에는 라인이 존재할 수 없다고 가정했기 때문에 셀 내에 있는 라인은 제거가 불가능하지만, 셀 외부에 연결돼 있는 라인들은 제거가 가능하다.



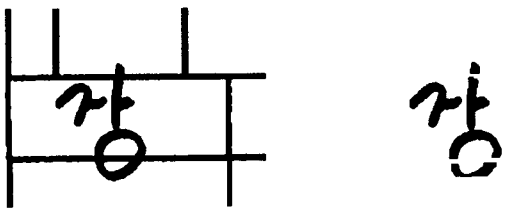
(a) 이진화된 입력 영상

(b) 입력영상에서 라인이 제거된 결과

<그림 2> 4개의 주요라인의 제거

셀을 이루는 주요 4개의 라인중엔 수평라인이 두 개, 수직라인이 두 개가 존재한다. 먼저 수평라인 중 윗 수평라인에 연결돼 있는 라인은 윗 방향으로의 런길이가 라인을 판별하는 임계치값 보다 길면 라인으로 인식해서 제거하고, 아래 수평라인에서는 아랫방향으로의 런길이를 조사해서 제거작업을 수행한다. 수직라인도 위와 마찬가지로 왼쪽에 있는 수직라인의 경우에는 왼쪽 방향으로의 런길이를 구해서 제거하고, 오른쪽 수직라인의 경우에는 오른쪽 방향으로의 런길이를 구해서 라인을 제거한다.

여기서 라인을 판별하는 기준이 되는 값은, 라인인지 혹은 문자의 긴 획중의 일부인지를 잘 구별할 수 있는 가장 적당한 값을 찾아야 한다. <그림 3>에 위의 작업을 수행한 결과 그림이 있다.



(a) 입력영상 (b) 라인제거 후

<그림 3> 다양한 셀의 라인 제거

3. 손상된 문자의 복원

일정한 양식을 갖는 문서상에 기록된 오프라인 필기체 문자들은 형식문서의 라인들과 겹쳐질 수 있다. 앞에서 일단 형식문서를 이루는 라인들을 제거했었다. 그러나 라인의 단순한 제거는 라인에 겹쳐진 문자들을 손상시키므로 라인의 제거 후에 문자부만을 복원하는 과정이 추가되어야 한다. 그렇기 때문에 이 단계에서 인쇄양식위에 기록한 필기문서의 라인제거 및 문자복원 알고리즘을 적용하여 라인의 제거와 함께 제거된 라인으로 인하여 손상되는 문자의 복원을 수행한다.

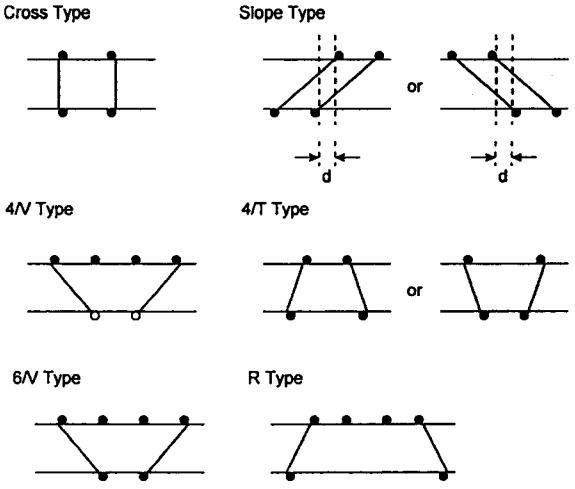
라인의 제거 및 문자의 복원을 위해서는 다량의 자료를 수집하여 내용을 분석한 후 발생할 수 있는 경우를 구분하고 이를 처리할 수 있는 알고리즘을 구현해야 한다. 이를 위하여 본 연구에서는 형식문서에 겹쳐지는 가능성을 확인하기 위하여 중앙대학교에 재학중인 학생들과 그 학생들의 친척들을 대상으로 인쇄된 자료형식을 배포하고 수거한 후, 접합된 형태에 대한 형태적인 분석을 수행하였다. 수집된 자료는 총 616매였으며 이들 자료를 대상으로 하여 겹쳐진 형태에 대한 분석을 시도하였다. 라인에 접촉되는 문자의 접합점 수에 따라 먼저 자료를 대분류한 후, 이를 다시 라인과 문자가 접촉된 모양에 따라서 접합형태가 유사한 타입별로 소분류를 수행하였다. 이때, 소분류되는 타입의 이름은 그 모양과 유사한 영문 대문자 알파벳을 사용하였다.

<표 1> 접합점의 형태분석에 따른 클래스 분류

접합점 수	Type	Case
2		
4	C	
4	T	
4	S	
4	V	
6	R	
6	V	

분석의 결과로 나온 문자의 접합형태에 따른 클래스 분류는 <표 1>과 같이 접합점이 두 개인 경우와 4/C, 4/T, 4/X, 4/V, 6/R, 6/V Type 등 6가지 클래스로 나뉜다. <표 1>의 case에서 흰색은 라인이고 검은 색은 문자의 일부분을 나타낸다.

라인의 제거는 앞에서 밝힌 바와 마찬가지로 라인을 트레이스할 때 동시에 이루어진다. 이때, 겹쳐지는 문자는 그 접합점을 저장하고 문자 복원 알고리즘에 의하여 손상된 문자를 복원한다. 문자와 라인의 접합형태에 따른 각 타입별 접합형태와 복원 알고리즘은 다음 <그림 4>과 같다.



<그림 4> 타입별 문자 복원 방법

접합점이 2개인 경우는 라인의 제거로 인하여 접촉된 문자부위가 끊어지지 않으므로 문자의 인식에 커다란 지장이 없으므로 단순히 라인의 제거만을 수행한다. 위 그림에서 *Cross type*은 접합점이 4개로 라인과 문자가 교차되는 형태를 말한다. 대응되는 접합점을 연결하므로써 문자의 복원을 수행한다. *Slope type*은 접합점이 4개로 문자가 라인에 비스듬히 기울어진 형태이다. 이 경우도 각각의 대응되는 접합점을 연결한다. *4/V type*은 라인의 한쪽면에만 접합점이 4개이고 V자 형태로 접합되는데 이때는 대응되는 라인의 반대면에 임의의 접합점을 설정하고 4개의 접합점중 최외곽 두 개와 임의의 접합점을 연결한다. *4/T type*은 접합점이 4개로 T형태로 접합되는 것이다. 대응되는 접합점을 연결해도 손실되거나 추가되는 부분이 극히 적고 문자의 인식에도 별 영향을 미치지 않으므로 그대로 연결한다. *6/V type*과 *R type*은 접합점이 4개로 접합되는

형태가 사다리꼴 모양이다. 복원은 대응되는 최외곽 접합점들을 연결한다.

<그림 5> 복원한 결과

<그림 5>는 <그림 2>의 문자를 접합점을 이용해 복원한 결과를 보인 것이다. 먼저 문자의 끊어진 테두리 부분을 위의 복원 방법대로 두 접합점을 연결하는 직선을 그은 후, 그 두 직선 사이의 내부를 채운 것이다.

4. 필드의 추출

일반적으로 일정한 양식을 갖는 문서상에 기록된 오프라인 필기체 문자들은 형식문서의 라인들과 겹쳐지거나 그 내용이 기록되어야 할 형식문서의 셀을 벗어날 수 있기 때문에 본 논문에서는 필드추출 결과를 저장할 메모리의 크기를 실제 추출하려는 셀의 크기보다 상, 하, 좌, 우로 15 픽셀 씩 확장을 했다. 객관적으로 사람이 형식문서의 어떤 셀에 내용을 기록할 때는 되도록 그 해당 셀안에 내용을 기록하려는 경향이 있기 때문에, 이 정도의 확장이라면 충분하다고 여겨진다.

하지만 15 픽셀 씩 범위를 확장을 하다 보면 아래의 그림처럼 인접한 셀의 내용중에서 15픽셀 이내에 들어있는 내용도 추출된 필드결과에 포함이 되는 문제가 생긴다. 따라서 본 논문에서는 그 문제를 해결하기 위해 <그림 6>과 같이 셀을 구성하는 주요 4개의 라인과 접촉된 문자의 경우에는 셀의 밖에 있더라도 살려두고, 그렇지 않고 라인과 접촉되지 않고 셀의 범위를 벗어난 문자는 원하는 셀내의 문자가 아니라고 여기고 제거한다.

(a) 입력 영상 (b) 결과 영상

<그림 6> 셀 외부 데이터 삭제

5. 실험 및 결과분석

실험은 AGFA ARCUS 스캐너에서 300dpi로 스캔된 그레이 영상에 대하여 수행하였고 프로그램은 PC상에서 Visual C++로 작성되었다. 실험은 추출될 셀내의 문자가 라인에 접촉된 다양한 형태들과 라인에 접촉된 문자가 없는 형태, 그리고 기울어진 형태를 갖는 데이터를 대상으로 수행하였다.

실험에서는 셀을 구성하는 라인과 문자의 접촉이 일어나지 않는 경우에는 에러가 발생할 확률이 없으므로, 둘 사이의 접촉이 일어날 수 있는 여러 가지 경우의 수를 고려하여 실험을 수행했다. 실험결과 라인과 접촉된 대부분의 경우에 라인의 제거와 이로 인해 발생한 끊어진 문자의 복원이 제대로 수행되었다. 단, 문자 복원시 원형 그대로는 복원되지 않고 약간의 훼손이 있었지만 이는 인식에 아무런 영향을 미치지 않을 정도의 왜곡이다. 또한 기울어진 문서에 대한 실험에서도 수평선을 기준으로 해서 ±3° 정도까지의 기울어짐에 대해서는 수행이 제대로 되었다.

6. 결론

본 논문에서는 형식문서에서 인식을 원하는 특정 위치에 있는 셀의 내용을 추출하기 위해 라인을 제거하고 이로 인해 끊어진 문자를 복원하는 방법을 제시하였다. 그 결과 어느정도의 기울어짐을 가진 문서에서도, 발생가능하다고 생각된 모든 경우의 접촉형태에 대해 라인의 제거와 문자의 복원이 가능하였다.

그러나 추출할 필드의 결과를 저장할 메모리를 실제 셀의 크기보다 15 픽셀로 고정된 값으로 확장을 시켜서, 문자가 15 픽셀 밖으로 벗어난 경우에는 그부분은 손실되는 문제점이 생기고, 약간의 기울어진 형식문서에 대해서는 처리가 가능했으나 그 기울기가 더 커지는 경우에는 제대로 작업을 수행하지 못하는 문제점 등은 앞으로 더 연구가 필요하다.

참고문헌

[1] Shunji Mori, Ching Y. Suen, and Kazuhiko yamamoto, "Historical Review of OCR Research and Development", Proc. of the IEEE, Vol. 80, No. 7, pp. 1029-1058, July 1992.

[2] Ying Liu, Richard French, Sargur N. Srihari, "An Object Attribute Thresholding Algorithm for Document Image Binarization", International Conference on Documents Analysis and Recognition, pp. 278-281, 1993.

[3] Dacheng Wang and Sargur N. Srihari, "Analysis of Form Images", International Conference on Documents Analysis and Recognition, pp. 181-191, 1991.

[4] Didier Guillevic, Ching Y. suen, "Cursive Script Recognition: A fast reader scheme", International Conference on Documents Analysis and Recognition, pp. 311-314, 1993.

[5] 유진용, 권영빈, "인쇄양식위에 기록한 필기문서의 라인제거 및 문자복원", 한국정보과학회 봄 학술발표논문집 Vol. 23, No. 1, pp. 289-292, 1996.

[6] 유진용, "접촉된 문자를 갖는 기울어진 형식문서에서의 정보추출", 중앙대학교 석사학위 논문, 1997.

[7] Henry S. Baird, "The Skew Angle of Printed Documents", Proc. Conf. of the Society of Photographic Scientists and Engineers, pp. 14-21, 1987.

[8] H. Freeman, J. M. Glass, "Computer processing of line drawing images", ACM Computing Surveys, Vol. 6, No. 1, pp. 57-97, 1974.