복잡한 컬러 문서에 대한 문자인식

경북대학교 컴퓨터 공학과 인공지능 연구실

양철용, 김갑기 김진욱, 김항준

A Character Recognition on Complex Color Documents

A.I Lab. KyungPook National University

E-mail : caz@chollian.net

ChulYong, Yang KabKi, Kim JInWook, Kim HangJoon, KIm

요약-최근 수많은 인쇄된 문서들이 HTML과 같은 디지털 문서로 바뀌고 있으며 이를 자동으로 변환해 주는 문자인식 기술에 대한 관심이 증가하고 있다.

본 논문에서는 그림과 글자가 공존하는 문서에서 자동으로 문자영역을 추출해서 문자를 인식하는 방법을 제안한다. 우선 입력문서는 유사한 칼라로 이루어진 영역들로 나누어진 뒤 휴리스틱 룰에 의해 문자후보 영역과 비 문자 영역으로 나누어진다. 그 다음 이들 문자후보영역들은 문자인식기를 이용하여 문자 혹은 문자의 일부분으로 인식된다. 제안된 방법으로 여러 문서들에 대하여 실험한 결과를 보이며 그 성능을 평가한다.

1. 서론

인터넷의 발달로 수많은 인쇄된 문서들이 HTML과 같은 디지털 문서로 바뀌고 있으며 이를 자동으로 변환해 주는 문자인식 기술에 대한 관심이 증가하고 있다. 과거 문자인식은 주로 문서구조가 간단하며 그림과 글자가 분명히 구분되는 문서에 대한 처리가 주된 관심사였다. 하지만 기업의 팜플렛과 같은 문서들은 다양한 칼라의 글자와 그림들이 공존하기 때문에, 일반적인 문자인식 방법에서 접근하던 방법으로 문자영역을 추출하고 낱자를 추출하는 방법으로는 더 이상 효과적인 문서인식을 수행할 수 없게 되었다.

이와 같은 문제점을 해결하기 위해 문자영역의 특성을 고려하여 문자영역을 찾는 시도가 있었다[1]. 또한 신경망을 이용하여 영상에서 문자영역을 추출하는 방법도 제안되었으며[2], 문자영역 추출 후 인식까지 수행한 연구도 있었다[3].

이에 본 논문에서는 기업의 팜플렛과 같은 다양한 크기와 다양한 칼라의 글자와 그림이 공존하는 문서에 대해 효과적으로 문자영역들을 추출하고 추출된 영역 내에 존재하는 문자들을 인식하는 방법을 제시한다.

본 논문에서 제시하는 방법은 크게 두 부분으로 나누어진다.

첫 번째 단계는 입력영상을 균일한 칼라영역들끼리 묶는 세그멘테이션 과정이다. 이때 레이블링을 효과적으로 수행하기 위해서 입력영상을 그레이스케일 영상으로 변환한 뒤 세그멘테이션을 수행한다. 동일한 칼라영역으로 레이블링된 각각의 세그먼트들의 특성(칼라, 크기)과 주변 세그먼트들간의 상관관계(위치, 인접여부, 포함관계)를 이용하여 이들 세그먼트들 중 비 문자영역 세그먼트들을 제외시킨다.

두 번째 단계는 첫 번째 단계에서 추출된 문자후보 세그먼트들에 대해서 문자 인식기를 이용하여 인식을 하는 과정이다. 하지만, 첫 번째 단계에서 추출된 세그먼트들은 문자영역 세그먼트들과 비 문자 영역세그먼트들이 함께 존재한다. 그리고, 한글의 경우 일반적으로 하나의 세그먼트로 낱자가 구성되는 경우는 아주 드물다. 따라서 인식을 수행하기 위해서는 각각의 세그먼트들에 대해 인접한 세그먼트들끼리 같은 그룹으로 묶어서 이들에 대해서 인식을 수행해야 한다. 서로 일정거리 이내의 세그먼트들을 각각의 그룹으로 묶는 데에는 여러 가지 경우가 존재한다. 본 논문에서는 릴렉세이션(Relaxation) 방법을 이용하여 이들을 한데 묶은 뒤 문자인식기를 이용하여 인식을 수행한다. 문자인식기는 원형정합방법을 이용하여 인식을 수행한다. 입력 문자를 가로 세로 4×5의 크기로 메쉬(Mesh)한 후 이를 템플릿과 비교하는 방법을 이용한다. 각각의 세그먼트 조합에 대해서 문자인식기는 인식결과를 확률 값으로 나타낸다. 이때 너무 낮은 인식결과 값을 보이는 세그먼트의 조합이 존재하는 경우는 인식 대상에서

제외시킨다. 가능한 모든 경우에 대해서 인식을 수행한 뒤 가장 높은 인식결과를 보이는 세그먼트 조합에 대한 인식코드를 인식 결과로 출력한다.

2장에서는 전처리에 대하여 설명하며, 3장에서는 영역분리작업 수행 후 비 문자 영역과 문자후보 영역을 나누는 방법에 관하여 설명하며 4장에서는 추출된 문자후보 영역에서 문자인식을 수행하는 방법에 관하여 설명한다. 그리고 실험결과에서는 본 논문에서 제시한 방법을 이용하여 다양한 문서들에 대해서 인식을 수행한 결과를 보이며, 제시하는 방법이 복잡한 그림과 글자가 공존하는 문서들에 대해서 효과적으로 인식을 수행하는 것을 보인다. 또한 결론에서는 본 방법이 앞으로 해결해야 할 문제점도 함께 제시한다.

2. 전처리

본 논문에서 제안하는 방법은 영역분리작업의 결과에 크게 영향을 받는다. 영역분리 작업을 효과적으로 수행하기 위해서 입력문서에 대해 몇 가지 전처리를 수행한다.

우선 입력문서에 옅은 색의 배경이 있을 경우 이를 제거하기 위해 입력 영상을 그레이스케일 영상으로 변환한 후, 그레이 영역에서 농도가 임계치 이하인 점들을 지운다. 옅은 색의 배경은 대부분 그림인 경우이며 이 배경을 그대로 둘 경우 영역분할결과가 문자인식을 수행하기에 충분한 결과를 낼 수 없다. 입력영상의 각 픽셀의 그레이 값 P[i,j]는 0에서 255 사이의 값을 가진다고 할 때, 본 방법에서는 농도 값 210 이상인 점들을 모두 제거한다.

3. 문자영역 탐지

입력문서에 대해서 문자인식을 수행하려면 우선 문자영역의 위치를 찾는 작업이 선행되어야 한다. 그림과 글자가 함께 공존하는 문서의 경우 문자영역을 찾기가 쉽지 않은데, 본 논문에서는 입력영상의 칼라정보와 기하학적 정보를 이용해서 문자후보영역들의 위치를 찾는다.

3.1 영역 레이블링

레이블링은 크게 두 단계로 이루어진다. 첫 번째 단계는 각 픽셀에 대해서 유사한 농도 값을 가지는 픽셀들을 하나의 블록으로 묶는 과정이며 두 번째 단계에서는 첫 번째 단계에서 생성된 각 블록들에 대해서 크기가 작은 블록들을 유사한 그레이 농도 값을 가지는 이웃블록들로 합병하는 과정이다.

이웃하는 두 픽셀 P[i,j]와 P[k,l]의 농도 값 차이가 Θ이하이면, 즉 $|[i,j[- P[k,l]|< \Theta$이면 두 픽셀은 같은 레이블을 가진다.

본 논문에서는 이웃 화소들 중 임계치 값 20 이하의 거리를 가지는 점들에 대해 같은 레이블을 할당한다.

또한 생성된 영역들 중 화소의 개수가 40개 이하의 영역들은 이웃 영역들 중 그레이 농도가 가장 가까운 영역으로 합병한다.

3.2 문자후보 영역 추출

문자후보 영역 추출은 각 레이블링된 블록들의 크기와 칼라특성 그리고 주변 블록들간의 관계들을 이용해서 구해진다. 본 논문에서 사용하는 휴리스틱 룰은 다음과 같다.

① 4가지 이상의 다른 칼라의 블록을 포함하는 블록은 문자후보 영역에서 제외시킨다.
② 다른 블록을 포함하는 블록을 포함하는 블록은 문자후보 영역에서 제외시킨다.
③ 블록의 높이나 폭이 입력문서의 높이나 폭의 1/2을 초과하는 경우 문자 후보 영역에서 제외시킨다.
④ 블록의 높이와 넓이의 비가 1 : 2 혹은 2 : 1 이 이내이면서 픽셀의 개수가 임계치 이상인 블록은 문자 후보 영역에서 제외시킨다.
⑤ 블록의 폭 혹은 높이가 임계치 이하인 블록은 문자 후보 영역에서 제외시킨다.
⑥ 블록을 둘러싸는 최소 사각형 넓이 대 픽셀의 개수비가 임계치 이하면 문자 후보 영역에서 제외시킨다.

4. 문자인식

본 방법에서 사용하는 문자인식기는 원형정합방법을 이용하여 인식을 수행한다. 그레이 스케일의 입력 문자를 가로 세로 4×5의 크기로 메쉬(Mesh)한 후 이를 그레이 스케일 템플릿들과의

거리를 계산하여 가장 가까운 거리를 가지는 템플릿을 인식결과로 출력하는 방법을 이용한다. $T[i,j]$를 템플릿의 그레이 값이라고 할 때 입력문자 K와 템플릿 L의 거리는 다음과 같이 구한다.

$$D(K,L)=1-\sum_{i=1}^{4}\sum_{j=1}^{5}\frac{|T[i,j]-P[i,j]|}{255}$$

각각의 세그먼트 조합에 대해서 문자인식기는 인식결과를 확률 값으로 표현되며, 이때 너무 낮은 인식결과 값을 출력할 경우 미 인식으로 간주한다.

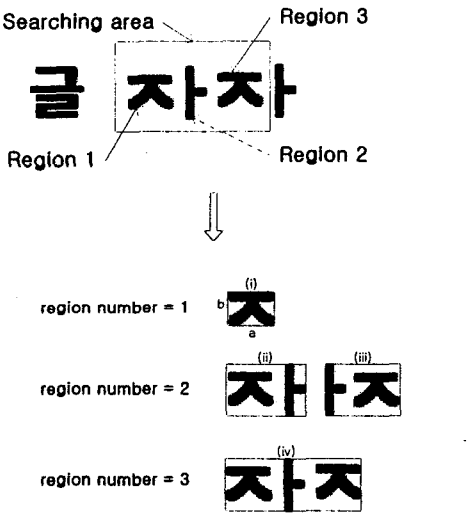

그림 1. 문자인식을 위한 문자후보 영역 그룹핑

한글의 경우 글자는 보통 두 개 이상의 자 소들이 결합하여 한 글자를 이룬다. 영어 알파벳의 경우 대부분 한 블록으로 구성되는 경우가 많기에 문자후보영역 탐지단계에서 나온 후보영역들을 바로 인식기로 인식하여 결과를 추출할 수 있지만 한글을 인식하기 위해서는 두 세 개의 블록들을 하나의 문자인식단위로 간주해야 하는 문제점이 있다.

그림 1은 문자인식을 위해서 문자후보영역들을 그룹핑하는 방법을 보여준다. 서로 이웃하는 유클리디안 거리 ϕ이내의 영역들 중 같은 칼라 값을 가지는 영역들을 하나의 입력패턴으로 만들어 문자인식기를 적용시킨다. 이와 같은 방법으로 모든 문자후보영역에 대해서 가능한 모든 조합으로 인식을 수행한다.

5. 실험결과

본 논문에서 제안하는 방법을 테스트하기 위해서 모두 20장의 문서를 가지고 실험을 하였다. 실험에 사용한 시스템은 펜티엄III 550 MHz이며 HP ScanJet 3300C 스캐너를 이용하였고 스캔영상의 해상도는 200 DPI이다. 그림 2는 몇몇 입력영상에 대해서 인식을 수행한 결과를 보여준다.

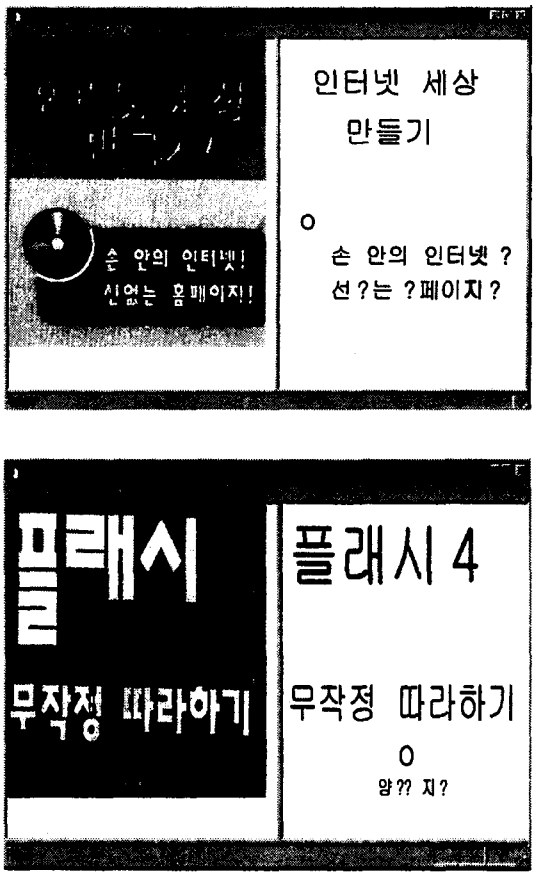

그림 2. 인식결과

표1.은 제안한 방법의 인식률을 보여준다. 인식기에서 인식 값을 임계치 이하로 출력할 경우 이를 미 인식 문자로 간주하였다.

표 1. 인식률

미 인식	오 인식	총 인식률
10%	5%	85%

표 2는 제안한 방법의 수행 시간 분석을 보여준다. 표 2에서 보듯이 대부분의 시간은 영역분할을 할 때 소요된다. 향후 이 부분에 대한 고속의 알고리즘 개발이 필요하다고 볼 수 있다.

표 2. 수행시간

전처리	120ms
영역분할	6s
문자후보영역 추출	30ms
문자인식	860ms

제안한 방법은 이웃하는 글자끼리 자소가 붙어 있는 경우 올바른 인식을 수행할 수가 없다. 문자영역을 미 인식 한 경우나 오 인식 한 경우를 살펴보면 크게 영역분할 시 오류, 접촉문자 오류 문자인식기 오류로 나누어 볼 수 있다. 표 3에 그 비율을 나타내었다.

표 3. 잘못 인식한 원인

영역분할 오류	5%
접촉문자 오류	85%
문자인식기 오류	10%

본 논문에서 제안하는 방법은 영역분할 단계의 결과에 크게 영향을 받는다. 특히 입력문서에 색의 변화가 완만한 영역이 존재할 경우 이의 영역분할이 제대로 되지 않으며 뒤 문자인식단계에도 영향을 주게 된다. 특히 영역분할에서 구한 블록들에 대해서 메쉬 방법으로 문자 인식을 수행하기에 영역분할이 안정적으로 되지 않을 경우 인식기에서 오 인식이나 미 인식을 내는 경우가 많았다.

제안한 방법은 회사 홍보문서와 같은 깨끗하고 그림과 글자가 공존하는 복잡한 문서에 대해서 잘 동작한다. 하지만 신문과 같이 이웃 글자들의 획이 서로 붙어있는 경우가 많은 문서에 대해서는 아직 보완해야 할 점들이 많이 있다.

6. 결론

현재 시중에서 시판중인 대부분의 문자인식기들은 그림과 글자가 복잡하게 배치되어 있는 문서에 대해서 문자인식을 제대로 수행하지 못한다. 그 이유는 입력문서에서 글자부분과 그림부분을 효과적으로 분류하지 못하기 때문이다.

본 논문에서는 이렇게 글자와 그림이 복잡하게 배치되어 있는 문서에 대해 문자영역을 자동으로 찾아서 이를 인식하는 방법을 제안하였다.

제안하는 방법은 문자후보영역 추출과 문자인식 두 부분으로 구성되었다. 문자후보영역 추출은 우선 영역분할후 각 영역의 칼라정보와 크기정보 그리고 위치정보들을 이용하여 문자후보영역과 비 문자 영역을 구분하였으며, 문자 후보영역들에 대해서 가까운 영역들끼리 한데 묶어 입력패턴을 만든 뒤 인식을 수행하였는데 릴렉세이션 방법으로 모든 가능한 조합에 대해 다 적용하였다.

제안한 방법은 개개의 문자들이 이웃 문자들과 서로 떨어져 있는 영상에 대해서는 우수한 인식률을 보였다. 하지만 이웃 글자간에 자소가 서로 접촉하는 경우 효과적으로 처리하지 못하는 문제점이 있다. 또한 영역분할에서 많은 시간이 소요되기 때문에 대량의 문서를 처리해야 할 경우 커다란 문제가 된다.

향후 연구에서는 이런 속도 문제와 자소 간의 접촉문제를 해결해야 하며 또한 글자영역의 다른 영상적 특성, 예를 들면, 글자부분과 주변배경 부분의 칼라 값이 급격하게 변하는 특성도 함께 고려하는 방법도 찾아볼 것이다..

참고문헌

1. Ki-Young Jeong, Keechul Jung, and Hang Joon Kim, "Neural Network-based Text Location for News Video," ACM Multimedia, 1999.

2. Anil K. Jain, and Bin Yu, "Automatic Text Location in Images and Video Frames," Pattern Recognition, Vol. 31, NO. 12, pp. 2055-2076, 1998

3. Jun Ohya, Akio Shio, and Shigeru Akamatsu, "Recognizing Characters in Scene Images", IEEE PAMI Vol. 16, No 2, February 1994