연결요소와 색상정보를 이용한 실제적 문서영상 분할

김 병 기†

요 약

문서영상의 분할은 문서인식의 전 과정 중에서 인식률에 큰 영향을 미치는 단계이지만 국내에서는 양적·질적으로 이에 대한 실제적인 연구가 부족한 것이 사실이다. 그 결과, 구조가 복잡하거나 칼라로 인쇄된 문서의 분할은 여전히 개선할 점이 많다. 본 논문에서는 불규칙한 다단, 점선, 그래픽, 사진 등의 다양하고 복잡한 요소로 구성된 문서의 실제적인 분할문제들을 살펴보고, 연결요소와 색상정보를 이용하여 이들을 효율적으로 분할하는 실제적 문서영상 분할 기법을 제안한다. 윤곽선 추출을 이용하여 다양한 형태의 모든 연결요소를 추출하고, 추출된 연결요소별 유형판정 및 연결요소 병합기준을 이용함으로써 정확한 문서영상 분할이 가능하다. 또한 색상문서의 경우, 정확한 문서분할과 처리시간 개선을 위하여 먼저 이진화된 문서에 대하여 문자와 비문자 영역으로 분할한 후, 필요에 따라 비문자 영역에 대하여 별도의 칼라별 영역분할을 수행한다. 제안된 방법의 성능을 확인하기 위하여 다양한 구조와 내용을 갖는 180장의 문서영상을 대상으로 문서분할 실험을 수행하였다. 아울러, 6가지 국내외 상용 문서인식 소프트웨어의 문서영상 분할 결과와 비교함으로써 제안한 방법이 복잡한 문서영상의 실제적 분할에 우수한 성능을 보임을 확인하였다.

Practical Page Segmentation using Connected Components and Color Information

Pyeoung-Kee Kim†

ABSTRACT

While page segmentation is an important step in document recognition, there haven't been many researches on it. More improvement is still needed on the segmentation of document elements in complicated or color documents. In this paper, I present a new page segmentation method which can segment pages with multiple columns, dotted lines, graphics, and photographs. I extract all connected components using contour following and combine them depending on the size and positional information of them. Separate text location is done for non-text color regions to extract possible text lines. To see the performance of the proposed method, experiments are done for 180 documents. Four commercial OCR programs are also tested and the proposed method showed the best result.

1. 서 론

정보화와 더불어 전자문서 사용의 증가에도 불구하고 인쇄매체가 갖는 고유한 장점 때문에 여전히 많은 정보가 문서형태로 제작 및 사용되고 있다. 따라서, 문서로 표현된 정보의 자동입력을 위한 첫 단계로서 효율적인 문서영상 분할기법의 필요성은 정보화와 더불어 오히려 증대되어 왔다.

문서처리는 문서의 영역을 기하학적으로 분리하고 분리된 영역의 속성을 지정하는 문서분석(Document Analysis)과 문서요소의 논리적 상관관계를 추출하는 문서이해(Document Analysis)로 나눌 수 있다[1]. 문서영

† 정 회 원 : 신라대학교 컴퓨터정보공학부 교수
논문접수 : 1999년 9월 29일, 심사완료 : 1999년 11월 25일

상 분할(Page Segmentation)이라 함은 문서분석의 한 단계로서 스캐너 등의 입력장치를 통하여 그림 형태로 입력된 문서영상으로부터 문서 작성자가 의도한 문서 요소별로 영역을 분리해 내는 것을 말하고, 문서영상 분할을 통하여 얻어진 문서요소에 대하여 유형을 지정하는 것을 문서영상 분류(Page Classification)라고 구분한다. 분류의 대상이 되는 대표적 문서요소로는 사진, 표/글상자, 선, 문자블록, Line Drawing, 큰글자, 도형/순서도 등이 있다.

문서영상 분할이 문서인식시스템에 있어서의 첫 단계이고, 이 단계의 처리결과가 전체 문서인식 시스템의 성능에 직접적으로 영향을 끼침에도 불구하고 현재까지의 문서인식시스템의 문서영상 분할 루틴은 많은 개선할 점을 갖고 있다[1, 2]. 현재까지 개발된 국내외 상용시스템의 경우, 특히 다음과 같은 문제점들이 시스템의 영상분할 성능 저하에 결정적인 요인이 된다. 첫째, 사진이나 그림과 같은 비문자 영역(Non-character region)을 둘러싸는 직사각형의 가상영역 내에 "그림 제목"과 같은 문자블록이 인접하게 위치한 경우에 문자블록이 그림의 일부로 포함되어 추출이 누락되는 경우가 많다. 둘째, 점선(Dotted Line)과 같은 다양한 형태의 선 추출 오류 및 이로 인한 다른 블록의 오분류가 많이 발생한다. 셋째, 다단(Multiple Columns) 문서의 경우, 두 개 이상의 단 영역에 가로로 걸쳐 불규칙하게 배치된 그림, 사진, 문자블록 등의 추출 오류가 많다. 넷째, 단의 개수가 많은 다단문서에서의 단 처리 오류가 많다. 이러한 문제점들은, 일반적인 문서가 포함하는 요소가 다양하고 이들의 배치가 복잡하다는 이유 이외에도 문자인식에 비하여 상대적으로 복잡한 구조를 갖는 문서에서의 영상 분할에 대한 연구가 부족하였다는 데에도 원인을 찾을 수 있다.

좋은 문서영상 분할 기법은 다음과 같은 성질을 가져야 한다. 첫째, 문서처리 시스템의 활용범위가 제한되지 않도록 처리가능한 문서 종류에 대한 제약이 가급적 적거나 없어야 한다. 즉, 복잡하게 배치된 다단을 갖는 잡지 문서나 점선, 표, 복잡한 모양의 그림을 갖는 임의의 문서에 대한 처리가 가능하여야 한다. 뿐만 아니라, 다양한 색상(Color)을 갖고 복잡하게 인쇄된 문서의 처리도 가능하여야 한다. 둘째, 영상 입력과정에서 어느 정도 기울어진 상태로 입력되는 문서의 처리가 가능해야 한다. 기울어진 문서의 처리는 영상분할 이전 단계에서 별도의 전처리 과정으로 수행하는 경우도 있고, 어떤 영상분할 알고리즘은 별도의 기울어짐 처리가 필요 없는 경우도 있다[3, 4]. 셋째, 전체 문서처리 속도의 향상을 위하여 영상 분할에 소요되는 처리시간이 짧을수록 좋다. 넷째, 문서영상 분할과 동시에 영상 분류 정보를 추출할 수 있으면 영상 분류 단계의 복잡도를 줄일 수 있다.

본 논문에서는 연결요소와 색상정보를 이용하여 전술한 성질을 만족하는 실제적 문서영상 분할 기법을 제안하고, 각 문서요소별 판정기법을 소개한다. 윤곽선 추적을 이용한 연결요소 추출로 처리 가능한 문서종류에 대한 제약을 없애고, 비문자 영역에 국한된 선별적 색상문자 추출로 색상문서에 대한 처리를 가능하게 함과 아울러 속도를 개선한다. 문자영역 추출과 아울러 각 블록의 유형분류를 병행함으로써 문자 영역분할의 정확도를 높이고 영상분류 단계의 부담을 덜어준다. 잡지와 같이 문서구조가 복잡한 일반문서에 대한 국내외 상용 문서인식시스템과의 비교실험을 통하여 기존 시스템들의 성능과 문제점을 살펴보고 제안한 문서영상 분할 기법의 성능을 확인한다.

본 논문의 구성은 다음과 같다. 2장에서는 문서영상 분할과 관련한 국내외의 연구들을 살펴보고, 3장에서는 제안한 기법을 처리 단계별로 소개한다. 제안한 기법의 성능을 확인하기 위하여, 다양한 잡지문서에 대한 실험 및 국내외 상용 문서인식시스템과의 성능비교를 4장에 나타내고 결론은 5장에 보인다.

2. 관련 연구

문서영상의 영역분할은 영역 추출의 단위에 따라 크게 상향식 분할법과 하향식 분할법으로 나눌 수 있다[2]. 상향식 분할법은 연결화소와 같은 개념을 이용하여 문자, 문자열, 문단, 문서(쪽)와 같은 순서로 기본 추출단위(Basic Object)로부터 큰 복합 추출단위(Complex Object)로 병합해 나가는 방식이다. 하향식 분할법은 투영과 같은 방법을 이용하여 상향식과 반대로 큰 추출단위로부터 작은 추출단위로 분리해 나가면서 영역을 분할하는 방식이다.

국내의 관련 연구를 크게 초기의 투영을 이용한 하향식 방법과 연결요소를 이용한 상향식 방법으로 구별할 수 있다. 그러나, 지금까지의 문서영상 분할에 대한 국내의 연구는, 다소 제한적인 문서에 대한 연구나 문자열 내에서의 글자나 자소 분할에 관한 연구가 주류

를 이루었고 테스트 문서의 수가 충분하지 못하였다. 초기의 문서분할 관련 연구로는 참고문헌 [5]와 [6]을 들 수 있다. [7]에서는 연결화소를 이용하여 문서영상을 단어 단위의 문자영역과 비문자 영역으로 분할하고, 자소 인식을 병행한 문자열 추출 방법을 제안하였다.

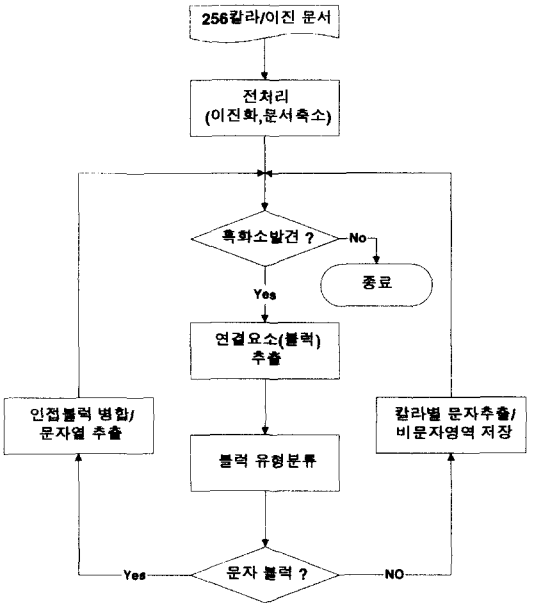
[8]에서는 문서영상의 에지(Edge) 정보를 이용하여 블록분할 및 유형분류를 수행한다. 에지 화소들의 세기, 방향별 분포, 에지 화소의 수 등을 특징으로 하여 배경과 밝기변화에 둔감한 유형분류를 시도하였다. [9]에서는 논문 초록부분을 대상으로 국내 상용 문자인식기(O.C.R)의 인식 성능을 테스트하였다. 문서의 구조가 매우 간단한 논문 초록에 대한 실험이었지만 국내 상용 문자인식기의 성능이 DB구축과 같은 분야에 실제로 활용하기에는 부족한 것으로 드러났다. [10]에서는 색상 클러스터링과 연결요소를 이용하여 색상문서에서의 개별 문자추출을 수행하였다. [11]에서는 복잡한 색상문서에서 지역별 양자화를 이용한 문자추출을 수행하였다. 이는, 문자의 색상이 다양하고 배경이 복잡한 경우에 전역적 색상 양자화가 갖는 단점을 개선한 것이다.

국내의 경우와는 달리, 국외의 경우에는 다양한 문서영상 분할 연구가 진행되어 왔다[12-24]. 특히, 문서의 검은 화소 대신에 배경의 흰 화소를 대상으로 'White Tile'을 구한 후 이들의 기하학적 위치관계를 그래프로 표현한 후 영역분할을 시도하여 좋은 결과가 보고된 경우도 있다[13, 14]. 투영의 단위를 화소가 아닌 연결요소를 에워싸는 영역(Bounding Box)을 사용한 경우도 있다[15]. 아울러, 문서영상 분할 알고리즘의 성능평가를 위하여 벤치마킹(Benchmarking)하려는 시도도 있다[23, 24]. 국외 문서영상 분할 관련 연구들이 주로 문자와 비문자 영역의 기본적 단위 추출을 위주로 한 반면, 국내 연구들은 추출된 영역의 유형분류나 문자 인식까지 병행한 경우들도 많았다. 이는 한글이 갖는 조합문자(Combined Character)의 특징으로 인하여 추출된 문자영역의 확인이 상대적으로 서구의 비조합 문자의 경우보다 다소 복잡한 점에 기인한다고 볼 수 있다.

3. 연결요소를 이용한 문서영상 분할

3.1 제안한 방법의 개관

본 논문에서 제안한 문서영상 분할 방법은 크게 전처리, 연결요소(블록) 추출, 블록의 유형분류 및 병합, 문자열 추출 및 비문자영역 저장으로 나뉜다. (그림 1)에 제안한 문서영상 분할기법의 흐름을 보였다.



(그림 1) 제안한 시스템의 개관

첫째, 전처리 단계에서는 입력문서가 색상문서일 경우에의 이진화와 흑백문서에 대한 문서크기 축소 전처리를 수행한다. 본 연구에서는 입력문서의 색상정보 포함여부를 미리 아는 경우와 모르는 임의의 경우에 선택적으로 사용이 가능하도록 두개의 문서 스캐닝 메뉴를 갖도록 하였다. 즉, 처리할 대상이 흑백 문서임을 미리 아는 경우에는 입력할 때 흑백모드(B/W Mode)로 스캔하고, 문서의 종류를 모르는 일반적인 임의문서의 경우에는 256 색상모드(Color Mode)로 스캔한다. 문서가 색상모드로 입력된 경우에 한하여 처리속도를 높이기 위하여 이진화 전처리를 수행한다. 이 때, 이후의 단계에서 가능한 비문자 영역에 대한 색상별 문자영역 추출을 위하여, 입력된 색상영상을 별도의 배열에 보관한다. 영상분할에서는 문자인식이 필요하지 않고 인쇄된 영역이 축소된 상태에서도 영상분할이 가능하므로, 처리속도 개선과 스미어링(Smearing) 효과를 갖는 문서축소 전처리를 수행한다. 문서의 축소는 입력문서 영상을 가로 및 세로 방향으로 각각 1/3 크기로 차례로 축소시키는 간단한 방법을 사용한다.

둘째, 연결요소 추출 단계에서는 이진화된 영상에 대

하여 좌상단(0,0)에서부터 시작하여 위쪽에서 아래쪽으로, 왼쪽에서 오른쪽으로 주사하면서 검은 화소를 찾아 연결요소를 구한다. 본 논문에서 연결요소를 사용한 것은 다단문서와 같은 복잡한 문서나 색상문서 등에서 불규칙하게 배치된 문서요소들을 정확하게 추출하고, 어느 정도 기울어진 문서에서도 영역분할이 가능하도록 하기 위함이다. 연결요소의 추출방법으로는 윤곽선 추적을 사용하였다.

셋째, 발견된 연결요소(블록)에 대하여 이후 과정에서 더 이상의 블록분류가 필요한지를 판별하기 위하여 문자블록인지를 판정하는 유형분류를 수행하고 인접블록을 병합한다. 문자영역으로 판정된 블록의 경우, 상·하·좌·우에 인접한 블록과의 블록높이, 시작위치, 상대적 크기 등을 고려하여 같은 문자열에 속할 경우 인접 블록들을 병합한다. 비문자 블록에는 (1) 그림이나 사진과 같이 실제로 비문자 영역인 경우, (2) 여러 색상들이 사용된 문자영역이지만 이진화 과정에서의 정보손실로 비문자로 판정되는 경우, (3) 비문자 영역 내에 문자가 있는 경우 등으로 나눌 수 있다. 이러한 세 가지의 경우들 중에서, 두 번째와 세 번째의 경우에 정확한 영역분할을 위하여, 여러 색상값을 갖는 비문자 영역에 대해서는 각 색상별로 이진영상에서 사용하는 분할 기법을 반복 사용한다.

넷째, 최종적인 문자블록들을 추출하고, 비문자 영역으로 판정된 블록은 그림형태로 저장한다.

3.2 전처리

대부분의 문서영상 분할 방법에서는 문서영상의 획득 후에 문서 기울기의 보정, 글자의 진행방향 결정, 잡음제거, 스미어링(Smearing)과 같은 전처리 과정을 거친다[5]. 본 연구에서는 가로쓰기 형태의 문서를 영상분할의 대상으로 하며, 투영과 같이 기울기에 민감한 하향식 기법 대신에 윤곽선 추적을 사용하므로 이진화와 문서크기 축소 이외에 별도의 기울기 보정이나 글자의 진행방향 결정과 같은 전처리는 수행하지 않는다.

입력문서가 흑백문서임을 사전에 알지 못하는 임의의 문서인 경우에는, 비문자 블록에 대하여 있을 수 있는 칼라별 영상분할 과정을 위하여 256 칼라모드로 스캐닝 한다. 전체 영역을 색상 처리 대상으로 하는 대신, 문서영상 분할 과정에서 비문자로 판정된 영역에 대하여만 색상의 사용여부를 판정하는 루틴을 사용하여 선별적으로 처리한다. 즉, 문서영역 전체에 대하여 칼라 문서인지를 판정 및 처리하지 않고 일단 흑백상태에서의 문자/비문자 영역 분류과정을 거친 후에 비문자 영역에 대하여만 칼라여부를 판정하므로 색상정보 사용 여부의 판정 및 영역 분류에 소요되는 처리시간을 줄인다.

잡음이란 인쇄시 의도하지 아니한 문자나 비문자 정보를 의미하는데, 임의 블록의 잡음 여부는 블록의 크기가 과다하게 작은지 여부, 인접 블록과의 거리가 자신의 블록크기에 비하여 멀리 떨어져 있는지 여부, 이웃 블록과의 위치 상관관계 등에 의하여 판정될 수 있다. 블록 유형분류 단계에서 잡음으로 판정되는 블록은 영상분할 대상에서 제외된다.

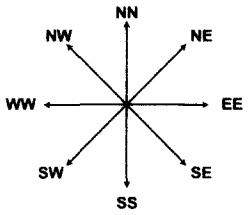
스미어링이란 연결되어야 바람직한 요소임에도 불구하고 영상입력 과정에서 연결되지 못하는 경우가 생기지 않도록 검은 화소 사이에 존재하는 길이가 작은 여백을 검은 화소로 바꾸는 과정으로, 스미어링 임계치를 잘 정해 주어야 하고 수행속도를 느리게 할 소지가 있다. 본 논문에서는 스미어링 효과도 가지면서 전처리 이후의 처리속도를 향상시킬 수 있도록 문서크기 축소를 수행하므로 별도의 스미어링 과정은 사용하지 않는다. 문서 크기의 축소는 9 화소(3×3) 크기의 윈도우(Window)를 문서 영상 위에 겹치지 않게 움직여 가면서 윈도우상의 9 화소들 중에서 한 화소라도 흑화소이면 축소후의 화소값을 흑화소로 하는 논리 OR연산을 수행한다. 따라서, 문서 축소를 통하여 문서크기 축소 전의 경우와 비교하여 처리시간을 대략 1/9 정도로 줄이는 효과를 갖는다. 이 때, 윈도우를 더 크게 하면 처리시간을 줄일 수는 있지만 축소 전 영상과의 왜곡정도가 심해질 수 있으므로 9화소 정도의 크기가 적절하다.

3.3 연결요소 및 블록 정보의 추출

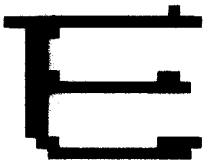
3.3.1 연결요소의 추출

문자열 발견을 위하여 전처리 과정에서 축소된 문서영상에 대하여 연결요소를 찾는다. 문서영상의 좌상단에서 시작하여 우측·아래 방향으로 줄 단위로 주사하면서 연결요소를 찾는다. 본 논문에서는 윤곽선 추적을 위하여 (그림 2(a))에 나타낸 바와 같은 8방향을 사용한다.

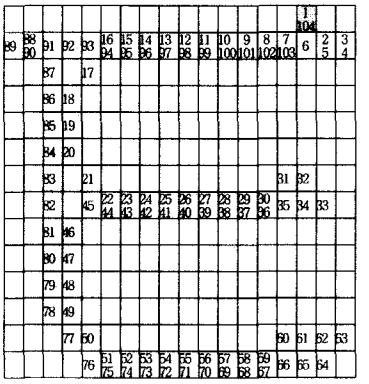
(그림 2(b))와 (c)는 각각 추적할 문자영상과 추적결과의 예를 나타내고, (c)에서의 숫자는 각 화소가 방문된 순서를 나타낸다. 먼저, 처음 만나는 흑화소를 시작점으로 하여 시계방향으로 연결화소를 추적한다. 획의



(a) 8방향



(b) 문자 영상



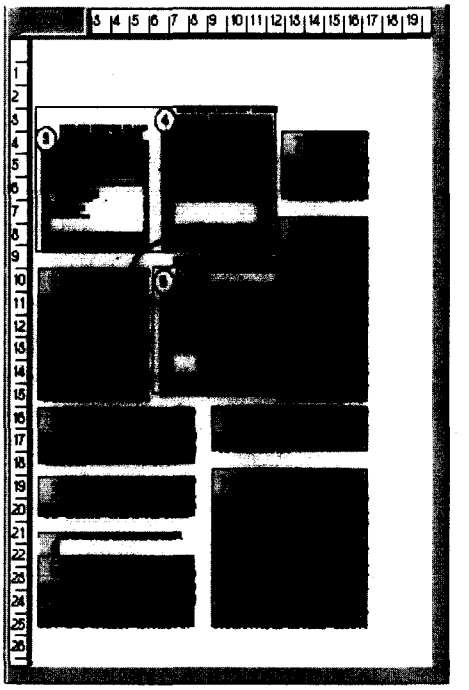
(c) 윤곽선 추적의 예

(그림 2) 윤곽선추적

굵기가 1인 경우 연결요소의 추적이 중간에서 중단되는 것을 막기 위하여 한 화소당 최대 2번 방문할 수 있도록 하였고, 시작점으로 되돌아오면 추적을 종료하게 된다. 이때, 윤곽 정보만 추출해도 되고, 불필요한 화소추적을 방지하도록 4-이웃화소(Four Neighbors) 모두가 흑화소인 내부 화소(Inner Pixel)들은 윤곽선 추적 대상에서 제외시킨다.

연결요소를 추출할 때, 이미 추출된 영역이 이후의 윤곽선 추출 과정에서 다시 블록 추출대상에 포함되지 않도록 하기 위하여, 추출된 연결요소내의 흑화소들은 지워야 한다. 이 때, 기존 연구와 다수의 상용 시스템들에서 연결요소를 둘러싸는 최접직사각형(MBR) 내의 모든 흑화소를 지움으로써 지우지 않아야 할 다른 블록의 흑화소도 지우는 경우가 많다.

(그림 3)에 상용 OCR 프로그램 "아르미"에 의한 오류를 예로 보였다. 문자블록은 추출된 블록의 왼쪽 상단에 일련번호로 표시되어 있는데, 그림에서 타원으로 굵은 실선으로 표시된 부분에서 문자열 블록 6번의 문자열 일부(아래 부분)가 바로 아래쪽에 위치한 그림영역에 포함되어 그림과 함께 비문자 영역으로 잘못 분할된 경우를 보여준다. 본 논문에서는, 연결요소 추출시에 각 행(row)별로 최좌측·최우측 열 정보를 기억하여 MBR내의 흑화소 삭제시에 두 값의 내부 영역에 속하는 흑화소만 지우게 함으로써 의도하지 않는 화소의 제거와 이로 인한 블록추출 오류를 막는다.



(그림 3) 그림 주위 문자열 일부가 비문자 영역으로 잘못 분할된 예

3.3.2 블록 정보의 추출

윤곽선 추적을 통하여 블록을 발견하면, 그 블록의 상·하·좌·우를 둘러싸는 최접사각형(MBR)을 구한 후 그 블록의 정보(속성)를 구한다. 본 논문에서 추출하는 블록 정보로는 블록의 가로 및 세로의 크기, 블록 내 흑화소의 비율, 연결요소 방문시의 8방향에 따른 8가지 화소 진행방향 분포와 같은 것들이 사용된다[3, 4].

3.4 블록의 유형분류와 다양한 문서요소의 처리

블록의 유형분류는 해당 블록이 문자, 그림 또는 사진, 그래프, 선 등의 어떠한 내용인지를 판정하는 과정이다. 문서영상 분할 단계에서는 해당 블록이 문자영역인지 아니면 비문자 영역인지의 구별만 가능하면 되므로 유형분류가 반드시 필요한 것은 아니다. 그러나,

유형분류 정보는 문자블록인지의 판단뿐만 아니라 해당 블록이 다수 유형의 블록이 결합되어 있어서 추가 분류가 필요한 경우인지 판명하기 위하여도 필요하다. 이전 단계에서 추출한 블록의 정보를 이용하여 다음과 같은 문서내의 다양한 문서요소를 판별한다. 각 문서요소별 판정 및 처리는 다음과 같다.

3.4.1 문자블록의 판정

문자블록을 제외한 모든 문서요소를 비문자 블록이라 하고, 추출된 블록에 대하여 우선 문자블록인지 확인한다. 추출된 블록이 문자블록인지의 구분은 블록의 크기(높이, 가로길이), 세로/가로의 비율, 화소 진행방향 분포, 흑화소 비율 등을 이용하여 구현한다. 본 논문에서는 추출된 블록의 높이가 전체 문서 높이의 0.1배 이하이고 블록 내 흑화소의 비율이 33% 이상 85% 이하이면 문자블록으로 판정한다. 블록높이가 문서 높이의 0.1배 이상인 경우는, 문자의 높이가 A4 문서를 기준으로 할 때, 3cm 이상임을 의미하고, 이보다 큰 문자는 영상분할 단계에서는 일단 비문자로 처리한다는 뜻이다.

문자블록의 흑화소 비율을 33% 이상 85% 이하로 한 것은 실험적 수치이다. 즉, 흑화소 비율이 33% 보다 작은 블록은 그래프나 Line Drawing 등인 경우이고, 85% 이상인 블록은 사진이나 실선 등이 된다. 물론, 비문자블록이면서 흑화소 비율이 33%~85% 사이인 경우도 가능하지만, 이러한 경우는 문자블록의 높이 제한을 적용하여 문자/비문자 블록의 구분은 정확히 이루어진다.

3.4.2 표 및 그래프의 판정

표나 그래프의 경우에는 문자 블록에서와는 달리, 연결요소 발견과정에서 화소 추적의 진행방향 중 대각방향보다는 수평 및 수직 방향이 절대 다수를 차지한다. 이러한 점을 이용하여 블록내의 흑화소 비율이 33% 미만이고 화소의 수직/수평 진행방향이 전체 이동방향의 90% 이상인 경우에는 표 또는 그래프로 판정한다.

3.4.3 사진, 그림, 큰 글자의 판정

사진 및 그림의 경우는 문자블록이 아니고, 표/그래프의 경우에도 해당하지 않으면서 블록 내 흑화소의 비율이 85%초과인지 확인함으로써 판정한다. 이는, 대부분의 그래프에는 배경영역이 많이 포함된 반면, 사진 및 그림에는 이진화과정에서 전경영역이 많이 포함되기 때문이다. 큰 글자의 경우에는 영상 분할 과정에서는 일단 사진이나 그림으로 판정되고, 이 후 문자인식 단계에서 글자로 인식할 수 있다.

3.4.4 실선 및 점선의 판정

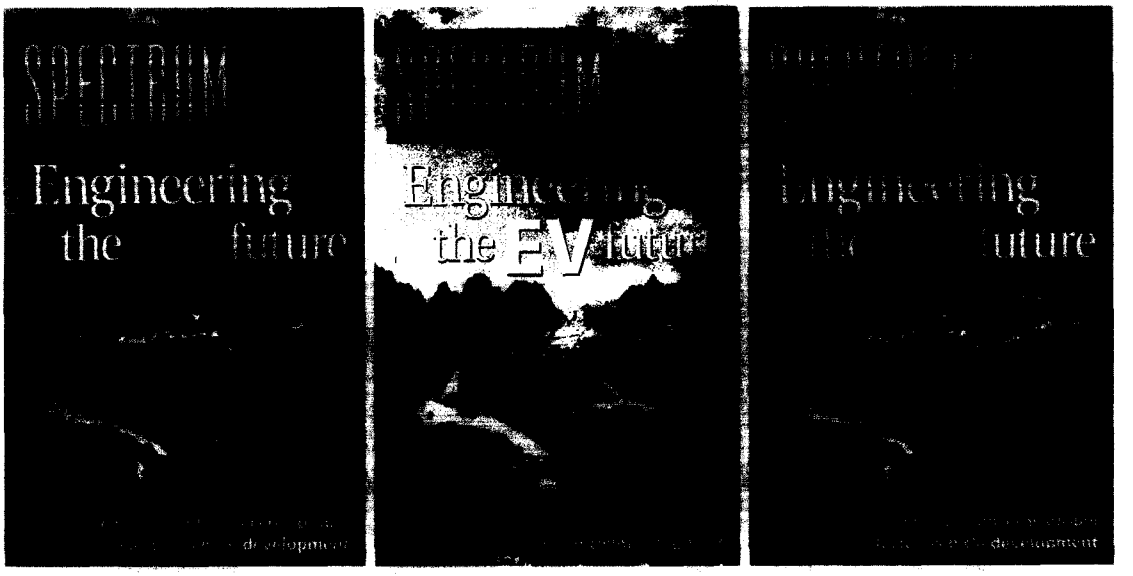
문서영상 분할에서 사용되는 선은 실선과 점선으로 구분되고, 일반적으로 점선의 처리가 실선의 처리보다 까다롭다. 실선의 경우에는 8가지 화소 진행방향 중 서로 반대 방향의 두 가지 진행방향의 빈도수가 전체 진행방향 빈도수의 65% 이상을 차지하고 흑화소의 비율이 90%이상이 넘는지 확인함으로써 판정할 수 있다. 가로 및 세로 점선의 경우에는, 인접블록의 병합과정에서 흑화소 비율이 90%이상이고 블록간 높이의 차이가 일정 화소 수 이하인 블록들이 일정한 방향으로 인접해 있는지 확인함으로써 판정한다. 실선의 경우에 흑화소 비율을 90%로 한 것은, 실선이 약간 기울어진 경우에도 추출 가능하도록 하기 위하여 여유를 둔 것이다.

3.5 색상 비문자 블록의 분할

실선이나 점선을 제외한 비문자 영역으로 판정된 블록에 대하여, 해당 블록이 3개 이상의 빈도수가 큰 색상수를 가질 경우 별도의 색상문자 추출과정을 거친다. 즉, 다수의 색상들이 사용된 문자영역이지만 이진화 과정에서의 정보 손실로 비문자로 취급되는 경우나, 비문자 영역 내에 비문자 영역과 밝기값이 유사하지만 색상이 다른 문자가 섞여 있는 경우인지 판정하는 단계를 거친다. 이와 같은 경우의 가장 대표적인 예로는 사진, 그림, 도표의 제목들이 비문자 영역과 상하나 좌우로 가까이 위치해 있어서 문자영역이 해당 비문자 영역에 포함되는 경우를 들 수 있다.

비문자영역의 색상포함 여부는 색상들 간의 거리(Distance)와 색상의 사용빈도(Frequency) 정보를 사용한다. 원본 색상영상에 대하여, 영역 내에서 색상의 사용빈도가 일정 비율 이상인 색상들을 구하고, 이 중 가장 빈도가 높은 색상을 기준으로 시작하여, 색상간 유클리디안 거리(Euclidean Distance)가 임계치 이상인 색상들에 대하여 색상별로 영상분할을 실시한다. 입력문서가 흑백문서임을 미리 알고 있는 경우에는 흑백문서 전용 메뉴를 사용하여 스캐닝부터 흑백으로 처리한다.

(그림 4)에, 색상문서의 이진화 결과와 색상 문자추



(a) 입력 문서 영상 (b) 이진화된 문서영상 (c) 색상 블록 분할 결과

(그림 4) 색상으로 인쇄된 문자열의 영상 분할

출의 경우를 보였다. (a)의 입력 영상에 대한 (b)의 이진화된 영상을 보면 문자들이 배경(그림)과 함께 섞여서 문자추출이 불가능함을 볼 수 있다. 반면에 (c)의 색상별 블록 분할의 경우에는 문자의 명암과 색상에 관계없이 문자열이 제대로 추출됨을 볼 수 있다[11]. 이렇게 함으로써 적은 추가 처리 시간을 들여 일차적으로 비문자로 판정된 영역에서 가능한 모든 문자열을 추출하는 효과를 갖게 된다.

3.6 인접 문자블록의 병합

문서영역 전체에 대하여 블록을 추출한 후, 블록의 속성이 문자블록으로 판정된 블록들에 대하여 더 큰 문자블록(문자열, 문단)으로 병합한다. 본 논문에서는, 인접해 있는 두 개의 문자블록이 다음의 조건을 만족하면 동일한 문자블록으로 병합한다.

(1) 두 문자블록의 영역이 겹치거나 한 블록이 다른 블록을 포함하는 경우 : 이 경우는, 한글과 같은 조합문자에서 자소나 문자의 일부가 보다 큰 영역을 갖는 문자나 문자열 부분에 위치상 겹쳐져 있는 경우에 이들을 하나의 문자나 문자열 블록으로 병합하기 위한 조건이다.

(2) 두 블록이 좌우 위치관계에 있고, 왼쪽 블록의 오른쪽 경계와 오른쪽 블록의 왼쪽 경계간의 좌우 간격이 각 문자블록 평균 높이의 140% 이하인 경우 : 좌우 방향으로 인접한 단어블록이나 이미 병합된 문자열 블록들을 병합하여 보다 큰 문자열 블록으로 병합하기 위한 조건이다.

(3) 두 블록이 상하 위치관계에 있고, 위쪽 블록의 아래쪽 경계와 아래쪽 블록의 위쪽 경계간의 상하 간격이 각 문자블록 높이의 50% 이하이고, 두 블록의 가로 길이와 위치의 차이가 각 블록 높이의 50%를 넘지 않는 경우 : 두 개의 상하 위치관계를 갖는 문자열 블록을 합쳐서 보다 큰 높이를 갖는 하나의 문자블록으로 병합하는 조건으로서, 이 병합과정을 통하여 하나의 문자 블록을 형성할 수 있다.

4. 실험 및 결과 분석

본 논문에서 제안한 문서영상 분할기법의 성능을 확인하기 위하여 문서영상 분할 실험을 수행하였다. 실험환경으로는, 입력장치로 엡손 GT-9500 스캐너를 사용하였고 문서영상 분할 알고리즘은 펜티엄 II PC상에서 C++ 언어로 구현하였다. 문서영상은 300 Dpi(OmniPage의 경우에는 150 Dpi) 해상도로 입력하였다.

(그림 5) 실험에 사용된 문서영상의 예

실험대상 문서는 다양하고 복잡한 문서구조에 대한 실험을 가능하게 하기 위하여 "월간 마이크로소프트"를 비롯한 10개의 잡지로부터 180장의 문서영상을 스캔하여 사용하였다. 이 중, 100장은 문서구조의 복잡성에 관계없이 임의 페이지를 스캔하여 구한 것이고, 나머지 80장은 다소 복잡해 보이는 문서를 스캔하여 구한 것이다. 테스트 데이터를 두 가지로 나눈 것은, 일반적인 복잡도에서의 문서영상 분할 성능과 복잡한 문서에서의 분할 성능을 비교함으로써 시스템 성능의 안정성을 확인하기 위함이다. "복잡한 문서"의 기준은 단(Column)의 개수, 문서요소들의 배열 복잡성, 비문자 영역의 복잡도 등을 참고로 하여 주관적으로 정하였다. (그림 5)에 나타낸 예와 같이, 잡지와 같은 일반문서의 구조는 논문지와같이 정형화된 문서보다 상당히 복잡함을 알 수 있다.

국내 상용 시스템으로는 "아르미 4.0", "글눈 98", "슈퍼리더 97"을 사용하였고, 국외 상용 시스템으로는 미국 PC Magazine 잡지의 벤치마킹 결과에서 우수한 성능을 보인 "OmniPage 8.0", "Presto! OCR 3.0", "Recognita Plus 4.0"을 사용하였다. 각 시스템의 성능을 평가하기 위하여 10가지의 대표적 문서요소별로 오류 경향을 구하였다. 임의 테스트 문서영상 100장에 대한 각 상용 문서인식 시스템과 제안한 문서영상분할 방법의 실험결과를 <표 1>에 나타내었다. <표 1>에서 '오류(#)'와 '비율(%)'은 각각 문서분할 과정에서 발생한 오류의 개수와 전체 오류에 대하여 상대적으로 차지하는 비율을 나타낸다. 본 실험에서는 해당 문서영상에서 한 영역의 오류라도 있을 경우에 이를 그 문서 전체의 오류로 간주하였기 때문에 오류율이 다소 높게 나타났다. 이렇게 간주한 이유는, 문서영상 분할 단계가 잘못 처리될 경우에 이후의 문자분할이나 문자인식 단계에 치명적인 오류를 주게 되어 문서인식률을 크게 저하시키기 때문이다. 각 시스템이 가장 많은 오류를 발생한 문서요소에 대한 비율을 표 내에서 음영으로 표시하였다.

제안한 문서영상 분할 기법은 상용 시스템에 비하여 안정적으로 적은 오류를 보였고, 특히 상용 시스템이 취약한 성능을 보이는 문자블록과 표의 추출 성능이 우수하였다. 임의로 선택한 100장의 문서와 다소 복잡한 문서 80장에 대하여 각 시스템별로 정확하게 분할한 문서 수와 분할율을 <표 2>에 나타내었다.

"아르미"와 "슈퍼리더"의 경우에는 문서 복잡도에

〈표 1〉 시스템의 문서요소별 오류 경향(100장당)

시스템 / 문서요소	아르미		글눈		슈퍼리더		OmniPage Pro 8.0		Presto! OCR 3.0		Recognita Plus 4.0		제안한 방법	
	오류(#)	비율(%)	오류(#)	비율(%)	오류(#)	비율(%)	오류(#)	비율(%)	오류(#)	비율(%)	오류(#)	비율(%)	오류(#)	비율(%)
사진/그림	3	6.7	2	5	6	12.5	8	16.7	5	20.8	2	10.5	1	14.3
표	3	6.7	1	2.5	3	6.3	7	14.5	0	0	2	10.5	0	0
선/점선	1	2.2	7	17.5	11	22.9	4	8.3	0	0	2	10.5	2	28.6
다단	3	6.7	13	32.5	6	12.5	12	25.0	4	16.7	4	21.1	1	14.3
문자블록	17	37.8	11	27.5	14	29.2	4	8.3	8	33.3	6	31.6	0	0
LineDrawing	2	4.4	0	0	3	6.3	4	8.3	0	0	0	0	1	14.3
글상자	4	8.9	0	0	0	0	3	6.3	0	0	0	0	0	0
어긋난 문단	1	2.2	0	0	0	0	0	0	0	0	0	0	1	14.3
큰글자	7	15.6	5	12.5	4	8.3	3	6.3	6	25.0	1	5.3	1	14.3
도형/순서도	4	8.9	1	2.5	1	2.1	3	6.3	1	4.2	2	10.5	0	0
계	45	100	40	100	48	100	48	100	24	100	19	100	7	100

〈표 2〉 시스템별 문서 영상 정분할 결과

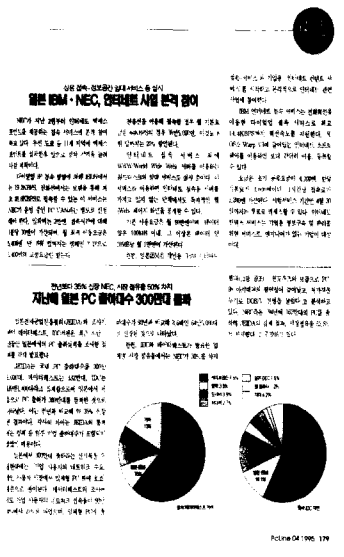
문서 \ 시스템		아르미	글눈	슈퍼리더	OmniPage Pro 8.0	Presto! OCR 3.0	Recognita Plus 4.0	제안한 방법
임의 문서 (100장)	문서수	60	71	59	59	82	85	95
	분할률	60.0%	71.0%	59.0%	59.0%	82.0%	85.0%	95.0%
복잡한문서 (80장)	문서수	39	18	44	20	50	54	71
	분할률	48.9%	22.5%	55.0%	25.0%	62.5%	67.5%	88.8%
계		55.0%	49.4%	57.2%	43.9%	73.30%	77.2%	92.2%

대하여 큰 차이를 보이지 않는 반면, 글눈을 비롯한 나머지 국내외 상용 시스템의 경우에는 구조가 복잡한 문서에 대하여 분할 성능이 상당히 저하됨을 볼 수 있었다. 단순한 구조를 갖는 문서의 경우에 대부분의 상용 시스템이 60~85%의 문서영상 분할성능을 보인 반면, 제안한 방법은 문서복잡도에 거의 관계없이 92% 이상의 안정적인 분할율을 보임으로써 가장 높은 문서영상 분할 성능을 보였다.

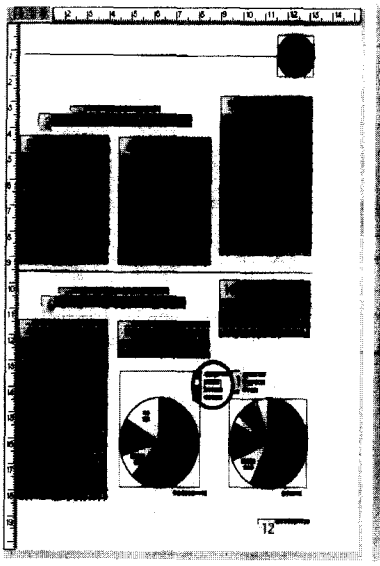
각 시스템별로 다단 문서영상 분할의 예를 (그림 6)에 나타내었다. (a)에는 3단으로 구성된 입력 문서 영상을 보였다. 제목에 해당하는 문자열이 두 개의 단에 가로방향으로 걸쳐져서 인쇄되어 있고, 아래쪽의 그림에 대한 범례가 문단간의 여백 위치에 가로로 걸쳐져 있는 다소 복잡한 문서이다. (b)는 (a)의 영상에 대한 상용시스템 "아르미"의 영상분할 결과를 보여준다. 그림에서 범례 부분의 일부(실선 타원으로 표시된 부분)가 추출대상에서 누락된 것을 볼 수 있다. (c)는 "슈퍼리더"에 의한 처리 결과를 나타내는데, 우측의 세 번째 단에서 가로방향 실선을 기준으로 두 개의 문자블록이 상하로 분리되어야 하는데 하나의 블록으로 잘못 병합·추출 된 것을 볼 수 있다.

(d)는 "Presto! OCR"에 의한 결과를 나타내는데, 추출된 각 블록의 좌상단에 표시된 숫자는 블록번호를 나타낸다. 5~7번 블록 부분은 입력 영상에서 문자의 크기가 다른 두 줄의 문자열이 상하로 구성되어 있는데, 분할 결과는 좌우로 세 개의 문자블록으로 잘못 분리되었음을 보여준다. 또한, 8번 블록은 두 개의 단이 하나의 단으로 잘못 병합된 경우이다. (e)는 "Recognita Plus"에 의한 분할결과를 보여주는데, 문자블록의 순서와 영역분할이 잘 된 것을 볼 수 있다. 제안된 방법에 의한 결과는 (f)에 보였는데, 각 블록이 정상적으로 분할되었음을 볼 수 있다. 특히, "Recognita Plus"를 제외한 상용 시스템들이 다단간의 구분 성능이 좋지 못한 반면, 제안한 방법에서는 문자블록간의 속성과 아울러 들여쓰기 정보를 이용하여 정확한 문단 구분이 가능하였다.

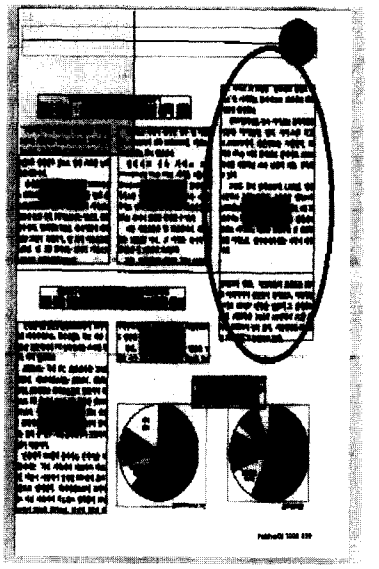
(그림 7)에 표 내부에 다단을 갖고 있는 문서에 대한 영상분할 결과를 보였다. (a)에는 3개의 가로 실선과 2개의 다단으로 구성된 표를 포함하는 입력 문서영상을 보여준다. (b)에 나타난 "아르미"에 의한 결과는,



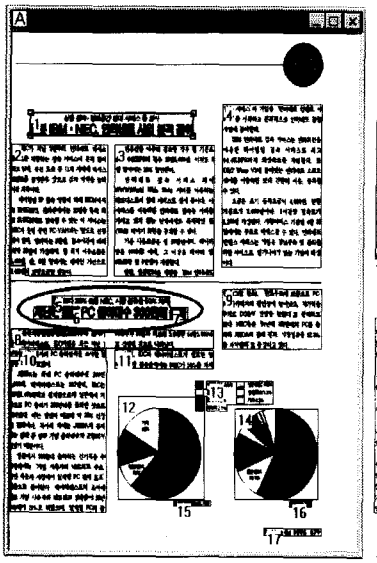
(a) 입력 문서 영상



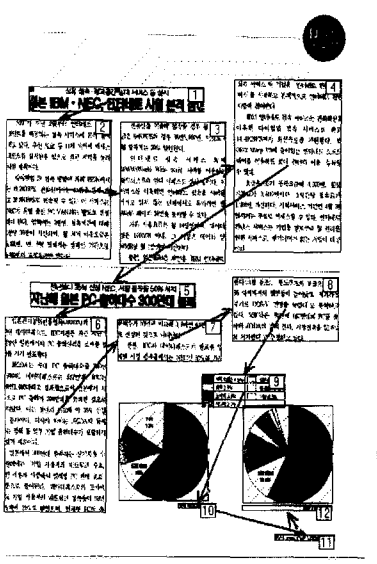
(b) 영상분할 결과(아르미)



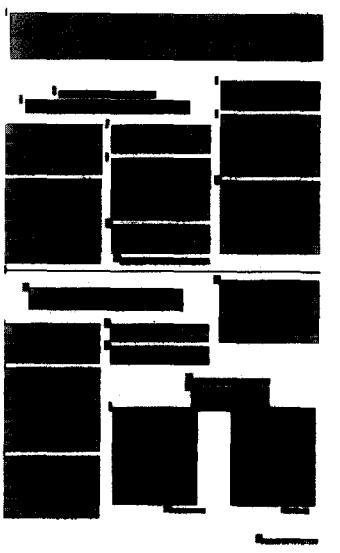
(c) 영상분할 결과(슈퍼리더)



(d) 영상분할 결과(Presto! OCR)



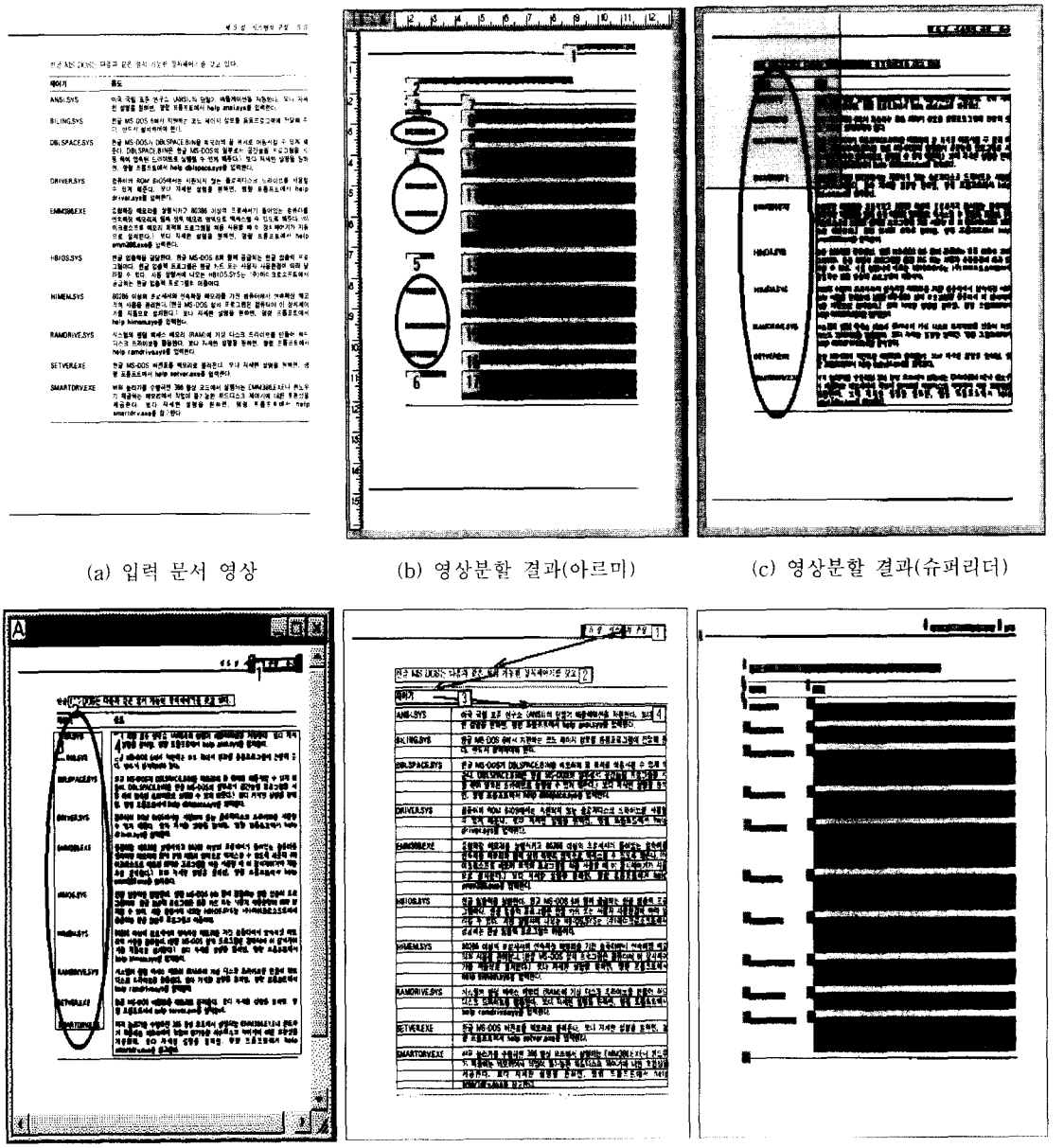
(e) 영상분할 결과(Recognita)



(f) 영상분할 결과(제안된 방법)

(그림 6) 각 시스템별 다단 문서영상 분할의 예

오른쪽 단 부분에 대한 분류는 제대로 되었지만 왼쪽 단에 대한 블록 추출에 일관성이 결여되어 다수의 문자열 블록이 추출 대상에서 누락된 것을 볼 수 있다. (c)에는 "슈퍼리더"에 의한 결과를 나타내었는데, 왼쪽 단의 문자열 모두가 추출 대상에서 제외됨을 알 수 있다. "Presto! OCR"에 의한 (d)의 결과는 표 내에서 다단의 분할은 제대로 이루어졌지만 단 내에서 문자 블록 사이의 분할이 이루어지지 않아서 큰 하나의 문자 블록으로 잘못 병합되었다. "Recognita Plus"에 의한 결과를 보여주는 (e)로부터, 표 내의 단 구분은 제대로



(a) 입력 문서 영상 (b) 영상분할 결과(아르미) (c) 영상분할 결과(슈퍼리더)

(d) 영상분할 결과(Presto! OCR) (e) 영상분할 결과(Recognita) (f) 영상분할 결과(제안한 방법)

(그림 7) 각 시스템별 표 내부에 다단을 갖는 문서영상 분할의 예

되었지만 문자열 상하간의 블록구분이 제대로 되지 않아 각각의 문자열을 표의 한 셀(Cell)로 처리한 오류를 볼 수 있다. 이 예에서 본 것과 같이, 대부분의 상용 시스템에서는 표 내의 다단문서 처리 및 문자블록의 추출에 상당한 문제점이 있는 것을 알 수 있다. (f)는 제안한 방법의 분할 결과를 나타내는데, 표 내부의 다단과 문자블록 병합 및 추출이 제대로 처리된 것을 보여준다.

제안된 방법의 성능을 실행속도 면에서 측정하기 위하여 300 Mhz의 클럭 속도와 64 MBytes의 주기억장치

〈표 3〉 각 시스템별 문서영상분할 처리시간

시스템	아르미	글눈	슈퍼리더	OmniPage Pro 8.0	Presto! OCR 3.0	Recognita Plus 4.0	제안한 방법
A4 문서당 평균 처리 속도(초)	1~2	1~2	1~3	1~2	1~1.5	1~1.5	1.31

용량을 갖는 펜티엄II PC를 사용하였다. A4 크기의 문서영상에 대한 각 시스템별 문서영상 분할에 사용되는 처리시간은 <표 3>에 나타내었다.

<표 3>에 나타난 상용시스템에 대한 처리 시간은, 프로그램 소스가 없으므로 프로그램을 이용한 직접적인 처리시간의 측정이 불가능하여 수작업으로 구한 개략적인 수치이다. 최근에 발표된 국외 상용 시스템들이 국내의 그것들보다 처리시간 면에서 다소 우수하였다. 본 논문에서 제안된 방법은 프로그램 상에서 직접 구한 결과, 평균 1.31초의 처리시간이 걸렸고 문서영상에 따라 최소 0.83초에서 최대 1.87초가 걸렸다. 따라서 제안된 방법이 처리속도 면에서 기존의 방법과 대등한 성능을 보임을 확인 하였다.

5. 결 론

본 논문에서는 연결요소와 색상정보를 이용하여 불규칙한 형태로 배열된 다단, 점선, 도형, 사진 등의 복잡한 문서요소들을 효율적으로 판정하고 분할하는 상향식 문서영상 분할 기법을 제안하였다. 먼저, 전처리를 거친 이진영상의 검은 화소들로부터 윤곽선 추적을 이용하여 연결요소와 그 속성을 구한다. 추출된 연결요소의 속성과 이웃 블록과의 상대적 위치관계 등을 고려하여 문자/비문자 블록을 추출하고, 두 문자블록의 크기와 이차원적인 배치를 고려하여 문자열이나 문단과 같은 보다 큰 문자블록으로 병합한다. 실선과 점선을 제외하고 비문자로 판정된 블록 중에서 3가지 이상의 색상을 갖는 경우에 정확한 문자추출을 위하여 선별적으로 색상별 문자열 추출을 수행한다.

제안된 방법의 성능을 확인하기 위하여 다양한 형식과 내용을 갖는 180장의 문서영상을 대상으로 6가지의 국내외 상용 문서인식 소프트웨어와 비교실험을 수행한 결과, 92%의 문서영상 분할 결과를 보임으로써 제안한 방법이 문서영상의 분할에 있어서 성능이 우수함을 확인하였다.

제안한 문서영상 분할 기법은 다음과 같은 점에서 의의를 찾을 수 있다.

첫째, 스미어링과 같이 시간이 많이 소요되는 전처리 대신에 축소된 이진 영상에 대하여 연결요소를 추출함으로써 처리시간이 단축되어 우수한 추출성능과 아울러 빠른 처리가 가능하다.

둘째, 색상문서에 대하여 먼저 이진화한 상태로 문자/비문자 블록을 추출하고, 선별적으로 비문자 영역에 대하여 색상 문자열 추출을 수행한다. 따라서, 문서 전체 영역에 대하여 색상별로 처리할 경우보다 처리 시간이 단축되고, 여러 색상이 혼재된 경우에도 정확한 문자 추출이 가능하다.

셋째, 추출된 각 블록에 대하여 실제적인 유형분류 기준을 설정하고 이를 문서분할에 사용함으로써 정확한 문자열 추출이 가능하다.

넷째, 여섯개의 국내외 상용시스템과의 비교 실험을 통하여 제안한 방법이 기존 시스템보다 영상분할 성능이 보다 우수함을 확인하였다.

향후 연구과제로, 제안된 영상분할 기법을 보다 다양하고 많은 실세계 문서를 대상으로 최적화 하고, 이를 기반으로 하여 인식성능이 우수한 문서인식 시스템을 구축할 예정이다.

참 고 문 헌

[1] Y. Y. Tang, *et al*, "Document analysis and understanding : A brief survey," Proceedings ICDAR, pp.17-31, 1991.

[2] 김두식 외 2인, "한글문서 분석 및 인식기술의 최근 연구동향", 전자공학회지, 제24권, 제9호, pp. 1058-1070, 1997.

[3] A. Antonacopoulos and R. T. Ritchings, "Flexible page segmentation using the background," Proceedings ICPR, pp.339-344, 1994.

[4] A. Antonacopoulos and R. T. Ritchings, "Representation and classification of complex-shaped printed regions using white tiles," Proceedings

ICDAR, pp.1132-1135, 1995.

[5] 남궁재찬 외 2인, "한국어 문서로부터 문자분리 및 도형추출에 관한 연구", 전자공학회논문지, 제25권, 제9호, pp.73-83, 1988.

[6] 이인동 외 2인, "블록영상의 추출 알고리즘", 한국정보과학회논문지, 제18권, 제2호, pp.218-226. 1991.

[7] 장명욱외 2인, "연결화소를 이용한 문서영상의 분할 및 인식", 한국정보과학회논문지, 제20권, 제12호, pp.1741-1751, 1993.

[8] 박창준 외 2인, "문서영상의 에지정보를 이용한 효과적인 블록분할 및 유형분류", 전자공학회논문지, 제33권 B편, 제10호, pp.120-129, 1996.

[9] 한선화 외 3인, "문자인식 기술을 이용한 데이터베이스 구축", 한국정보처리학회 논문지, 제6권, 제7호, pp.1713-1723, 1999.

[10] 김의정 외 2인, "칼라문서에서의 개별문자 추출", 한국정보처리학회 추계학술발표논문집, 제4권, 제2호, pp.595-598, 1997

[11] P. K. Kim, "Automatic text location in complex color images using local color quantization," Proceedings IEEE TENCON '99, Vol.1, pp.629-632, 1999.

[12] A. K. Jain and B. Yu, "Document representation and its application to page decomposition," IEEE PAMI, Vol.20, No.3, pp.294-308, 1998

[13] M. Ozaki, "Column segmentation by white space pattern matching," Proceeedings ICDAR, pp.134-137, 1995.

[14] T. Pavlidis and J. Zhou, "Page segmentation by white stream," Proceedings ICDAR, pp.944-953, 1991.

[15] J. Ha, *et al*, "Document page decomposition by the bounding-box projection techniques," Proceedings ICDAR, pp.1119-1122, 1995.

[16] E. Trupin and Y. Lecourtier, "A modified contour following algorithm applied to document segmentation," Proceedings ICPR, pp.525-528, 1992.

[17] T. Saitoh, *et al*, "Document image segmentation and text area ordering," Proceedings ICDAR, pp.323-329, 1993.

[18] D. Drivas and A Amin, "Page segmentation and classification utilising bottom-up approach," Proceedings ICDAR, pp.610-614, 1995.

[19] N. Normand and C. Viard-Gaudin, "A background based adaptive page segmentation algorithm, Proceedings ICDAR, pp.138-141, 1995.

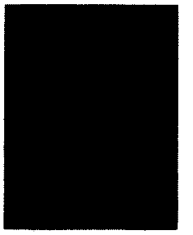
[20] D. Sylwester and S. Seth, "A trainable, single-pass algorithm for column segmentation," Proceedings ICDAR, pp.615-618, 1995.

[21] T. Saitoh and T. Pavlidis, "Page segmentation without rectangle assumption," Proceedings ICPR, pp.277-280, 1992.

[22] J. Sauvola, M. Pietikainen, "Page segmentation and classification using fast feature extraction and connectivity analusis," Proceedings ICDAR, pp.1127-1131, 1995.

[23] B. A. Yanikoglu and L. Vincert, "Ground-truthing and benchmarking document page segmentation," Proceedings ICDAR, pp.601-604, 1995.

[24] S. Randriamasy and L. Vincert, "Benchmarking page segmentation algorithms, Proceedings CVPR, pp.411-416, 1994.



김 병 기

e-mail : pkkim@lotus.silla.ac.kr

1988년 경북대학교 전자공학과 졸업 (학사)

1990년 경북대학교 대학원 전자계산기공학과(공학석사)

1995년 경북대학교 대학원 컴퓨터공학과(공학박사)

1995년~1996년 부산여자대학교 컴퓨터과학과 전임강사

1997년~현재 신라대학교 컴퓨터정보공학부 조교수

관심분야 : 패턴인식, 영상처리, 멀티미디어