문자 인식 기술을 이용한 데이터베이스 구축

한 선 화[†] · 이 충 식[††] · 이 준 호[†††] · 김 진 형[††††]

요 약

문자 인식 기술은 인쇄된 형태로 존재하는 수많은 정보를 데이터베이스화 할 수 있는 가장 유용한 대안이다. 본 논문에서는 문자 인식 기술을 사용한 데이터베이스 구축의 타당성을 조사하기 위하여, 문자인식기를 사용한 데이터베이스를 시범적으로 구축하였다. 우선 데이터베이스를 구축할 때 문자 인식기의 선택 시 고려하여야 할 사항들을 살펴보고, 이를 기준으로 4가지의 상용 문자 인식기에 대한 인식 실험을 거친 후 그 중 인식 성능이 가장 좋은 것을 선택하였다. 대상 문서로는 다양한 인쇄 품질 및 특성을 갖는 실제 논문집의 초록을 대상으로 삼았으며, 대량 데이터에 대한 인식률 계산을 위해 수작업된 데이터베이스가 있는 KT 테스트 컬렉션[1]을 선택하였다. 실험은 실제 대용량 데이터베이스 구축과 유사한 환경을 만들기 위해, 문서별 학습이나 기울기 보정 등의 사전 작업을 생략하였다. 실험 결과 970편의 논문 요약문에 대해 평균 문자 인식률 90.5%를 보여, 한글 문자 인식 기술이 아직 데이터베이스 구축에 활용되기에는 이르다는 것을 보였다. 문자 인식에 의한 인식 오류에서는 수작업한 문서에서 발견되는 오류와는 상이한 유형이 많이 발견된다. 본 논문에서는 추후의 연구를 위하여 문자 인식 텍스트에서 나타나는 오류의 유형을 분류하였다.

Building Database using Character Recognition Technology

Sun-Hwa Hahn[†] · Chung-Sik Lee[††] · Joon-Ho Lee[†††] · Jin-Hyung Kim[††††]

ABSTRACT

Optical character recognition(OCR) might be the most plausible method in building database out of printed matters. This paper describes the points to be considered when one selects an OCR system in order to build database. Based on the considerations, we evaluated four commercial OCR systems, and chose one which shows the best recognition rate to build OCR-text database. The subject text, the KT-test collection, is a set of abstracts from proceedings of different printing quality, fonts, and formats. KT-test collection is also provided with typed text database. Recognition rate was calculated by comparing the recognition result with the typed text. No preprocessing such as learning and slant correction was applied to the recognition process in order to simulate a practical environment. The result shows 90.5% of character recognition rate over 970 abstracts. This recognition rate is still insufficient for practical use. The errors in OCR texts are different from those of manually typed texts. In this paper, we classify the errors in OCR texts for the further research.

1. 서 론

고대 이집트에서 파피루스가 발명된 이래 인류의 문화는 종이라는 인쇄매체에 기록되어 전달, 계승되어 왔다. 그러나 지난 40여년 동안에 걸친 컴퓨터 분야의 급속한 발전은 방대한 양의 정보가 컴퓨터와 통신망에 의해 유통되는 정보화 사회를 탄생시켰으며, 정보화 사회에서 종이라는 전통적인 인쇄 매체는 그 역할이 감소되고 있다. 즉, 스무 권에 달하는 백과사전의 방대한 정보가 손바닥만한 CD-ROM에 저장되고, 시간과

† 정 회 원 : 연구개발정보센터(KORDIC) 선임연구원
†† 비 회 원 : 한국과학기술원(KAIST) 전산학과
††† 정 회 원 : 숭실대학교 컴퓨터학부 교수
†††† 종신회원 : 한국과학기술원 전산학과 교수

논문접수 : 1998년 9월 4일, 심사완료 : 1999년 5월 24일

장소에 구애됨이 없이 컴퓨터 네트워크를 통하여 유통되고 있다.

정보화 사회 성공의 관건들 중의 하나는 수 백년의 역사를 통해 생성되어 왔고, 현재도 활발히 생성되고 있는 활자화된 정보를 얼마나 효율적으로 전산화하여 사람들에게 제공하는가에 있다. 현재까지는 수작업을 통하여 이러한 문서들의 전산화 작업을 수행하고 있으나, 이러한 방식은 너무도 비효율적이고 처리할 수 있는 정보의 양에도 한계가 있다. 따라서 문서 형태의 정보들을 효율적으로 입력할 수 있는 획기적인 방법이 요구된다.

문자 인식은 인공지능 기법을 사용하여 활자화되어 있거나 손으로 작성된 문서를 인식하여 컴퓨터에서 사용할 수 있는 내부 표현 방식으로 변환시켜 주는 분야이다. 문자 인식 기술은 스캐너를 이용하여 입력된 문서 영상에 포함된 문자들을 자동으로 인식하는 수단을 제공함으로써, 수작업에 의존하던 자료 입력 방식을 자동화시켜 준다.

문자 인식 기법을 이용하여 활자화되어 있는 방대한 자료로부터 데이터베이스를 구축하려면, 우선 현재 사용 가능한 문자 인식기가 어느 정도의 성능을 지니고 있으며, 여러 유형의 문서를 인식할 때 어떤 특성을 보여주는지에 대한 조사가 선행되어야 한다. 또한 문자 인식된 텍스트에서 발견되는 오류의 유형에 대한 조사는 추후 데이터베이스의 품질을 높이는데 유용하게 사용될 수 있다. 본 논문에서는 위에서 언급된 두 가지 조사에 대하여 실제 문서를 문자 인식을 통해 입력하는 과정을 수행함으로써 향후 문서 인식을 이용한 데이터베이스 구축에 필요한 지침을 제공하고자 한다.

2. 문자 인식 기술의 개요 및 기술 수준

2.1 문자 인식 기술

문자 인식 기술이란 스캐너나 펜 입력 기기 등을 통해 입력된 영상으로부터 각종 문자 정보를 분리, 인식하여 전산화된 화일의 형태로 재창출하는 전산학의 한 분야이다. 문자 인식은 문자가 입력되는 방법에 따라 온라인 문자 인식과 오프라인 문자 인식으로 나누어진다. 온라인 문자 인식은 펜 입력기기를 통해 실시간으로 입력되는 문자를 인식하며, Hand-held 컴퓨터의 입력기기나 일반 컴퓨터의 보조 입력기기를 위한 기반 기술로 활용된다.

오프라인 문자 인식은 스캐너를 통해 입력된 문서 영상으로부터 문자를 인식하며, 활자화되어 있는 방대한 분량의 문서들을 자동으로 전산화하는데 유용하게 활용될 수 있다. 오프라인 문자 인식 기술은 다음과 같이 크게 문서 구조 분석 및 이해, 전처리, 특징 추출, 분류, 후처리단계로 구분될 수 있다[2].

- 문서 구조 분석 및 이해 단계에서는 스캔된 문서 영상으로부터 잡영을 제거하고, 문서의 구조를 이해하여 문서 영상으로부터 이미지와 문자 부분을 분리해 내고 문자 부분간의 순위 관계를 설정한다.
- 전처리 단계에서는 기울어진 문자열을 보정하고 문자열로부터 문자 영상을 분리한다.
- 특징 추출 단계에서는 분리된 문자 영상의 골격을 추출한 후 문자의 구성 원리에 입각한 특성 및 이들의 상관관계를 도출한다.
- 분류 단계에서는 추출된 문자의 특성에 따라 해당 문자 영상이 어떠한 유형의 문자에 속하며, 이 영상에 해당하는 문자 후보에는 어떤 것들이 있는지를 알아낸다.
- 후처리 과정에서는 인간이 문장을 이해하여 나가는 문맥적 정보를 이용하여 오류를 교정하거나, 후보 문자로부터 문자 영상에 해당하는 문자를 선택한다. 후처리에 해당하는 정보로는 전후의 언어적인 문맥에 의한 정보, 문자 단위 인식 정보 등이 있다. 후처리는 주소록과 같이 제한된 영역에 대한 문서를 인식할 때 매우 효과적으로 사용될 수 있다.

2.2 상용 문자 인식 시스템의 기술 수준 비교

40여년 전부터 연구 기관과 학술 기관에서 문자 인식에 관한 연구를 꾸준히 진행시켜 온 미국에서는 이미 100% 가까운 인식률을 보이는 문자 인식 시스템의 개발을 완료, 상용화하였으며, 이러한 문자 인식 시스템은 사무 자동화 및 데이터베이스 구축에 일익을 담당하고 있다.

1999년 1월 20일자 PC Magazine은 미국내 상용 문자 인식 제품에 대한 성능 실험을 수행한 결과를 기재하였다[3]. 이 실험에서는 단어 단위 인식률을 측정하

였으며, 다양한 인쇄품질을 가진 서적, 팩스문서, 사용자지침서, 업무용 서신, 신문 및 잡지 기사 등을 대상으로 40페이지 분량의 테스트 문서 집합을 구성하였다. 각 문서 영상은 다양한 글자 크기(4~72 폰트), 기울어진 영상, 밝거나 진한 영상등을 골고루 포함하도록 하였다. 또한 표, 이미지, 다준 컬럼 등 다양한 문서 구조를 포함하도록 하였다. 300 dpi로 스캔된 문서 흑백의 TIFF 파일로 저장하였다. 실험은 단어 인식률 뿐 아니라 페이지 당 평균 인식 속도도 함께 측정하였다. 7개의 상용 문자 인식 시스템을 대상으로 실험하였으며, 최대 인식률은 98.8%의 단어 인식률을 보였다. 인식률은 사람이 일일이 오인식 단어를 계수하여 계산하였다. 이 실험은 다양한 문서 유형에 대한 실험을 수행하기는 하였으나 대상으로 하는 문서의 장수가 40쪽에 그치고 있다는 단점이 있다. 본 논문에서는 실제 데이터베이스 구축에 사용될 논문지의 초록만을 대상으로 실험을 수행하였으며, 문서 영상의 특성을 분류하여 각 특성별 인식률을 조사함으로써 어떠한 문자 인식 시스템이 어떤 특성을 가진 문서 영상에 특히 강점/약점을 보이는 가를 알 수 있다. 또한 인식률 계산도 UNIX의 ***diff*** 명령을 이용하여 자동 계산함으로써 많은 수의 문서를 대상으로 인식률을 계산할 수 있다.

일본의 경우, 정부 주도하에 PIPS라 불리는 프로젝트를 수립하여, 산학 협동으로 효과적인 연구를 수행하고 있다. 일본의 문자 인식 연구 역사도 미국과 비슷하나, 문자 자체의 복잡도가 영어에 비해 높고, 인식하여야 할 대상이 히라가나, 가타가나에 한자까지 포함되기 때문에 난이도는 영어보다 훨씬 높다고 볼 수 있다. 그러나 도시바의 ExpressReader 70J의 경우 업체의 자체 발표에 의하면 약 4천자의 한자를 인식할 수 있을 뿐만 아니라, 문자의 크기도 6에서 40폰트까지 인식할 수 있으면서도 99.5%의 높은 인식률을 보이는 등 우리 나라에 비해 앞선 인식 기술을 보이고 있다[4].

한국에서의 문자 인식 기술은 1980년대 말부터 일부 학교를 중심으로 시작되었으며, 1993년 3월 삼홍시스템이 최초의 PC용 문자 인식 시스템인 NeuroOCR을 출시하였다. 같은 해 5월에 한국인식기술의 글눈이 발표되었으며, 뒤를 이어 여러 상용 제품들이 출시되어 1997년까지 출시된 한글 문서 인식 시스템은 총 6종이다[5]. 상용 문자 인식 시스템에 대한 비교 실험은 1994년 컴퓨터 매거진에 의해 최초로 실시되었다[6]. 대상 문자 인식 시스템은 하이아트 글눈, NeuroOCR, 르네상스OCR의 3종이었으며, 인식 실험과 인식 속도를 측정하였다. 대상 문서로는 소설, 논문, 프린터 출력문, 그림이 혼용된 잡지, 도표가 혼용된 잡지, 신문 가로쓰기의 일곱 가지를 각 1매씩 선정하였다. 이 실험은 초기의 문자 인식 시스템의 성능 비교라는 의의는 있으나, 인식률을 논하기에는 대상 문서의 수가 지나치게 작다. 동 잡지는 1996년 5월호에 다시 문자 인식 시스템에 대한 비교 실험 기사를 기고하였다[7]. 이 실험은 아르미, 글눈, 뉴로OCR 3종의 문자 인식 시스템을 비교 대상으로 수행되었으며, 레이저프린터 출력물, 2단 편집 잡지, 일반 도서, 한자가 혼합된 출력물, 영문, 2단 편집 신문, 도표 등 7가지 문서에 대해 각각 1매씩의 문서를 대상으로 하였다. 이 역시 일반적인 인식률을 논할 수 없는 실험이었으며, 단순히 각 문서에 대한 각 시스템의 인식 결과를 보여주는 형식을 취하고 있다. 인식 결과를 살펴 볼 때 1994년도의 실험에 비해 인식 능력이 한결 우수해 진 것을 알 수 있다.

1997년 HelloPC지는 한국의 대표적 OCR 4종에 대한 미니 벤치마킹을 시도하였다[5]. 대상 시스템은 스피드리더 1.2, 뉴로OCR, 글눈 97, 아르미 3.0이며, 대상 문서는 단락(한글/영문), 장문(한글/영문), 국/영 혼용, 해상도(150/300/600dpi), 변형문서, 필기체 폰트, 한글/한자 혼용 문서의 7종으로 각 유형 1건의 문서를 대상으로 실험하였다. 이 실험 역시 각 유형마다 하나의 문서를 실험 대상으로 삼았기 때문에 전체적인 인식 성능을 말할 수 없다.

이상에서 살펴본 바와 같이 문자 인식 시스템에 대한 나름대로의 실험은 꾸준히 행해져 왔지만, 대용량의 문서에 대한 실험은 한 번도 수행된 적이 없다. 그 이유는 우선 대용량의 문서를 스캔하여 파일로 저장하는 일에 많은 노력이 들 뿐 아니라, 각 시스템 별로 실험하는 일도 많은 시간이 소요되기 때문이다. 또한 인식률을 측정하기 위하여는 원래의 문서와 인식된 문서를 비교하여야 하는데, 지금까지 모든 실험의 경우 이를 사람이 직접 비교하여 인식률을 산출하거나, 혹은 인식률 산출을 포기한 채 사용자에게 인식 결과만을 보여주는 형식을 취하고 있다. 대상으로 하는 문서

의 수가 많아질 경우 사람이 직접 문서를 비교하여 인식률을 산출해 내는 작업은 거의 불가능하다.

본 논문에서는 보다 실제적인 환경에서 객관성 있는 인식 성능을 테스트하고자 각 특성 별로 선별된194편의 실제 문서를 사용하여 상용 문자 인식 시스템의 성능을 측정하였다. 또한 가장 높은 성능을 보인 문자 인식 시스템을 사용하여 약 1,000편의 논문 초록으로 구성된 KT 테스트 컬렉션을 대상으로 인식 실험을 수행함으로써, 문자 인식에 의해 대용량의 데이터베이스를 구축할 때의 예상 인식률을 추측하였다.

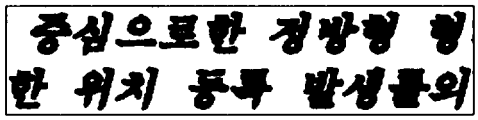
3. 문자 인식기의 선택

3.1 문자 인식기 선택시 고려 사항

현재 상용으로 나와 있거나 연구용으로 개발된 다양한 문자 인식기 중 어느 것을 선택하여 데이터베이스 구축에 사용할 것인가를 결정해야 할 때 고려할 사항은 다음과 같다.

3.1.1 폰트

인식기에 따라 특정 폰트에 대해서는 인식 결과가 뛰어나나 다른 폰트에 대해서는 인식률이 현저히 낮아지는 경우가 있다. (그림 1), (그림 2)에서와 같이 여러 가지 서체의 다양한 문서를 입력해야 하는 경우, 문자 폰트에 많은 영향을 받지 않는 인식기가 좋은 성능을 가졌다고 말할 수 있다.



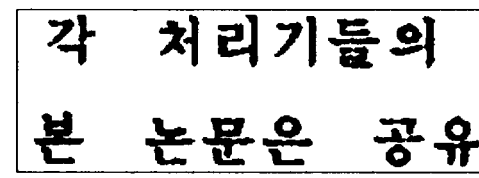
(그림 1) 기울임꼴 폰트를 사용한 문서 영상

(그림 2) 세로 방향으로 길쭉한 고딕체 문서 영상

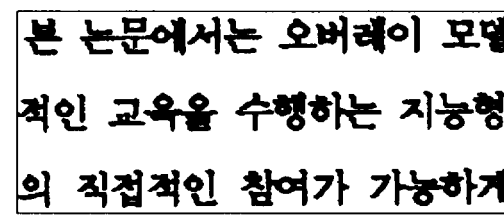
3.1.2 문자 간격

문자 간격이 넓은 문서에 대해서는 문자들의 구분이 명확하기 때문에 문자 분할이 수월하게 이루어질 수 있다. 그러나, 문자들의 간격이 좁아서 겹침이 발생하는 경우에는 문자 분할 시에 많은 오류가 발생한다. 예를 들면, (그림 3)의 문서 영상은 문자들의 간격이 충분히 넓어서 문자 분할 시에 발생하는 오류가 적었다. 그러나, (그림 4)에 나타난 문서 영상은 문자들의 접촉이 심하여 문자 분할 오류가 많이 발생되었다.

문자 분할이 잘못된 경우 대부분의 문자가 잘못 인식되므로, 문자 분할의 성능은 인식기의 성능에 크게 영향을 미친다. 인식기에 따라서 개별 문자 인식 성능은 매우 좋은 반면 문자 분할 성능이 좋지 않아서 전체적인 성능의 저하를 가져오는 경우가 있다. 따라서 문자 분할 성능이 좋은 인식기가 좋은 성능을 가졌다고 말할 수 있다.



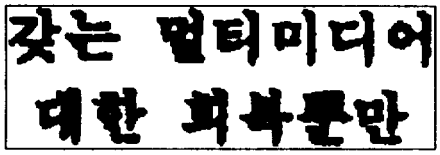
(그림 3) 문자간의 간격이 충분히 넓은 문서 영상



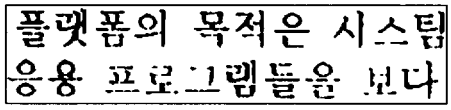
(그림 4) 문자간의 접촉이 많은 문서 영상

3.1.3 스캔 농도

문서 스캐닝 시에 사용된 화소 임계치는 문자 인식에 많은 영향을 미친다. 화소 임계치를 낮게 설정하여 문서 영상을 스캐닝하면 문자 획의 굵기가 필요 이상으로 굵어지게 되어 획간 접촉 발생 횟수가 많아진다. 반면에 화소 임계치를 높게 설정하면 문자 획의 굵기가 얇아지고 획의 끊어짐 현상이 발생한다. 예를 들면, (그림 5)는 낮은 임계치로 스캔된 문서 영상을 보여주며, 이로부터 획간 접촉이 심하게 발생하여 인식하기 어려운 문자가 많아졌음을 알 수 있다. ('회', '복', '뿐'이 이에 해당) 또한, (그림 6)은 높은 임계치로 스캔된 문서 영상을 보여주며, 이로부터 획의 끊어짐 현상이 많이 발생하여, 인식의 중요한 특징이 되는 부분들이 없어지는 경우가 발생됨을 알 수 있다.



(그림 5) 농도가 진하게 스캔된 문서 영상

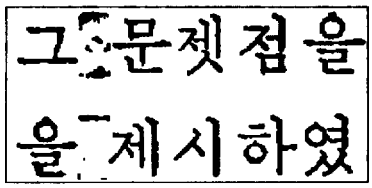


(그림 6) 농도가 엷게 스캔된 문서 영상

적절한 화소 임계치를 설정하였을 지라도 주어지는 문서의 인쇄 품질에 따라 획간 접촉이 심해지거나 끊어지는 현상이 발생할 수 있다. 따라서 스캔 농도에 따라 인식률의 차이가 심한 인식기보다는 스캔 농도에 의한 영향을 적게 받는 인식기가 다양한 인쇄 품질의 문서들을 연속적으로 인식하고자 할 때 보다 높은 인식률을 제공한다고 말할 수 있다.

3.1.4 잡음 처리 기능

문서 영상에는 여러 가지 이유로 인하여 잡음이 존재하게 된다. 예를 들어, (그림 7)은 잡음이 있는 문서 영상을 보여준다. 일부 인식기는 이러한 잡음들을 효과적으로 처리하는 반면, 잡음에 상당히 민감한 반응을 보이는 인식기가 있다.



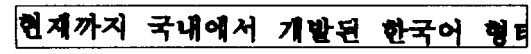
(그림 7) 잡음이 있는 문서 영상

3.1.5 DPI

문자 인식은 문서 영상 스캔 시에 사용하는 세밀도에 영향을 받는다. 너무 작은 DPI를 사용할 경우 문자 영상에서 나타나는 특징들이 충분히 표현되지 않아 인식이 어렵고, 너무 큰 DPI를 사용하면 문자 인식 시에 사용되는 데이터의 양이 필요 이상으로 커지는 단점이 있다. 따라서 작은 DPI를 사용하면서도 높은 인식률을 보이는 인식기가 좋은 인식기라고 말할 수 있다. 일반적인 문서의 경우 400 DPI 이상의 세밀도로 문서를 스캔해야 문자 인식에 필요한 정보를 충분히 얻을 수 있다. 본 연구에도 400 DPI를 사용하여 데이터베이스를 구축하였다.

3.1.6 기울어진 영상의 교정

스캔된 문서 영상에는 정도의 차이가 있을 수 있지만 (그림 8_과 같이 기울어짐이 존재한다. 대부분의 문자 인식기는 기울어짐 보정을 하고서 문자 인식을 수행하는데, 이러한 기능이 갖추어지지 않은 인식기나 기울어짐 보정의 성능이 좋지 않은 인식기는 데이터베이스 구축 작업에 실제로 사용하기가 어렵다.



(그림 8) 기울어진 문서 영상

3.1.7 문서 부분 및 이미지 부분의 분리 추출 능력

문서 영상은 텍스트, 이미지, 표 등과 같은 여러 가지 매체로 구성된다. 문자 인식기에 문서 영상이 주어지면 세그먼트 분할을 수행한 후에, 각각의 세그먼트를 해석한다. 이 과정에서 세그먼트의 특성을 잘못 해석하는 경우 추후의 인식 과정이 제대로 수행되지 못한다. 예를 들어, 텍스트 세그먼트를 이미지 세그먼트로 잘못 인식하면 그 텍스트에 대한 인식을 수행하지 않으며, 이미지 세그먼트를 텍스트 세그먼트로 잘못 인식하면 많은 양의 오류 텍스트를 생성한다.

3.1.8 사용자 인터페이스

여러 개의 문서 영상이 주어졌을 때 이러한 영상들을 연속으로 인식하는 기능이 요구된다. 즉, 다량의 문서를 입력할 때, 매 문서 영상마다 수작업으로 작업 명령을 주어야 한다면 자동문서 입력의 의미가 사라지게 된다. 이러한 점을 고려하여 편리한 사용자 인터페이스와 연속 작업 수행 기능을 제공하는 인식기를 선택하는 것이 유리하다.

3.1.9 인식 속도

방대한 양의 자료를 입력하기 위하여는 빠른 속도의 인식 능력이 필요하다.

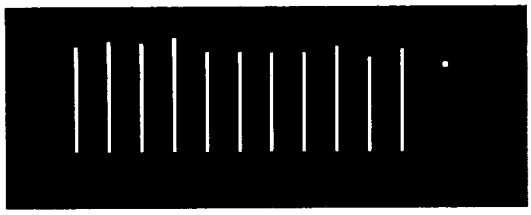
3.2 기존 문자 인식기의 인식 성능 비교

본 절에서는 아르미3.0 전문가용, 글눈96, 스피드리더1.2, MY-QREDER 전문가용의 4가지 상용 문자 인식기를 대상으로 앞에서 살펴본 고려 사항에 대하여

인식 실험을 수행하였다. 인식 대상 문서로는 KT 테스트 컬렉션[1]에 포함된 1000편의 문서 중 앞 절에서 언급한 고려사항에 기준하여 194편을 선택하였다. <표 1>은 문서 선택 기준이 되는 문서 유형과 유형별로 선택된 문서의 개수, 그리고 문서 유형별 상용 문자 인식기들의 인식률을 보여주며, (그림 9)는 이러한 인식률을 그래프로 보여주고 있다.[1)]

〈표 1〉 문서 유형별 상용 문자 인식기의 인식률

문서 유형	문서의 특성	문서 수	인식률 (%)			
			스피드리더	아르미	글눈	MY-QR
A	고딕체 문서	34	83.35	53.16	67.45	84.46
B	흐린 영상의 문서	19	87.61	26.43	65.02	82.90
C	줄 간격이 좁은 문서	14	86.26	54.47	54.79	67.02
D	줄 간격이 넓은 문서	8	90.81	71.61	78.52	89.03
E	기울어진 영상을 가진 문서	7	79.52	19.99	68.31	61.20
F	특이한 폰트를 사용한 문서	9	79.61	62.80	73.17	80.94
G	문자 크기가 작은 문서	5	78.82	14.96	58.75	64.69
H	문자간 간격이 좁은 문서	7	79.23	39.06	61.66	66.36
I	문자 정보가 유실된 문서	11	83.37	37.83	67.48	84.60
J	활자가 뭉개져 있는 문서	38	75.67	53.79	62.88	79.19
계		152	81.89	47.10	65.25	78.83



(그림 9) 문서 유형별 상용 문자 인식기의 인식률 그래프

위의 실험 결과에서 보여진 바와 같이 평가 대상이 된 4종의 상용 문자 인식기 중 거의 모든 문서 유형에 대해 스피드리더의 인식률이 가장 높았다. 특히 문자의 크기가 작거나, 문서가 기울어진 경우, 줄간 간격이 좁은 경우 등에 있어서 타 시스템보다 월등한 능력을 보여줌으로써, 이 시스템이 문서를 분석하고, 문자를 분리해내는 기술이 상대적으로 뛰어남을 입증하고 있다. ㈜다니엘텍의 MY-QREADER는 인식률은 스피드리더에 이어 두 번째로 좋았으나, 인식 가능한 화상 파일을 Windows Bitmap 파일과 Tiff 파일로 제한하고, 이미 스캔 된 화상 파일을 일괄처리 할 수 없어서 사용상 불편함을 초래하였다. 합산컴퓨터(주)의 아르미의 경우, 문서를 분석하여 문자 분리에 성공하였을 경우 가장 높은 인식률(87.14%)을 보였으나, 많은 수의 문서에 대하여 문서의 분석에 실패하고 시스템이 다운되는 오류를 보였다. 따라서 전체 인식률의 심각한 저하를 초래하였을 뿐 아니라, 여러 개의 문서를 연속적으로 인식해야 할 경우 상당한 불편이 예상된다[2)].

실험에 사용된 모든 문자 인식 시스템에 있어서 문서 인식 결과에 많은 수의 특수 문자가 포함되어 있는 등 후처리 기술은 상당히 미흡함이 발견되었다. 대상 문서의 영역이 좁혀져 있지 않은 경우, 사전을 사용하여 인식된 문서를 검증하는 단계의 후처리 기술은 적용하기 어렵다. 그러나 '영문자와 한글이 동일한 단어 내에 섞여 쓰이지 않는다'는 등의 경험적 지식을 사용하거나, 실제 문서에서 거의 사용되지 않는 특수 문자는 걸러내는 등의 후처리 기술을 접목시킴으로써 인식률을 현저히 높일 수 있으리라 기대된다.

4. 문자 인식에 의한 데이터베이스 구축 실험

4.1 대상 문서

본 실험의 대상 문서로는 KT 테스트 컬렉션을 구성하고 있는 논문을 사용하였다. KT 테스트 컬렉션은 정보과학회논문지, 한국정보과학회 학술발표대회논문집, 정보관리학회지에 수록된 1000편의 논문들에 대해 저자, 발행년도, 국문 및 영문 초록 등을 수작업으로 입력하여 구성한 작은 규모의 데이터베이스로 한글 정보 검색의 연구를 위해 널리 사용되고 있다. KT 테스트 컬렉션의 논문을 대상 문서로 선택함으로써 얻을 수 있는 장점은 다음과 같다.

- KT 테스트 컬렉션에 포함된 논문은 수년에 걸쳐 수집된 두 종류의 논문지 및 학술 발표집으로 구성되어 있으므로 다양한 인쇄 품질에 대한 인식 실험을 수행할 수 있다.
- 수작업으로 구축된 데이터베이스가 이미 존재하고 있으므로 문자인식 문서와 원래의 문서를 비교하여

1) 문서의 선택은 인식이 까다로운 문서를 위주로 하였기 때문에 대상 문서 전체에 대한 평균 인식률에 비해 상당히 낮다.

2) 이와 같은 현상은 아르미 전문가용 4.0에서 많이 개선되었으며, 이에 대한 실험 내용은 [8]를 참조하라.

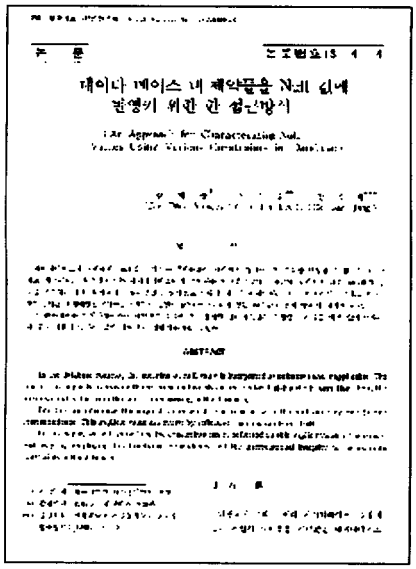
문자 인식률을 쉽게 측정할 수 있다.

- KT 테스트 컬렉션은 30개의 질의와 각각의 질의에 대한 적합 문헌리스트를 포함하고 있다. 따라서 본 연구를 통해 구축된 데이터베이스는 추후 문자 인식된 데이터베이스에 대한 검색 방법 연구에 좋은 자료로 활용될 수 있다[9].

본 연구에서는 KT 테스트 컬렉션의 1000편의 문서 중에서 정보관리학회지 2권 2호에 수록된 논문 6편과 11권 1호에 수록된 한편의 논문을 제외한 993편의 논문을 스캔하였다. 스캔된 논문들 중 970편의 논문으로 데이터베이스를 구축하였으며, '한글 요약이 없는 논문'과 '한글 요약에 한자가 섞여 있는 논문' 23편은 문자 인식 과정에서 제외하였다.

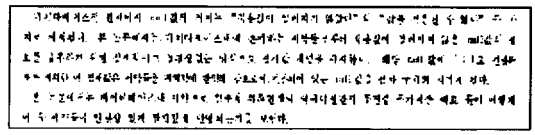
4.2 문자 인식 데이터베이스의 구축

본 연구에서는 3.2절의 실험 결과를 참고로 하여 가장 우수한 문자 인식 결과를 보여준 스피드리더 1.2를 사용하여 다음과 같이 문자 인식 데이터베이스를 구축하였다. 첫째, 대상 문서의 요약이 있는 첫 페이지를 400 DPI 이진 영상으로 스캔하였다. 스캔된 영상은 KT 테스트 컬렉션에 수록된 번호 별로 'p<번호>.pcx'의 형태로 저장하였다. (그림 10)은 스캔된 문서 영상의 한 예이다.

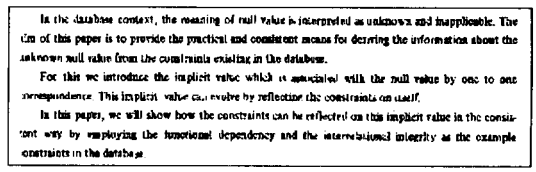


(그림 10) 문서 영상 예(p0050.pcx, 1:10)

둘째, 스캔된 문서 영상으로부터 다시 한글 초록 부분과 영문 초록 부분의 영상을 분할하여 저장하였다. 국문 초록 부분은 'h<번호>.pcx'의 형태로 저장하였고, 영문 초록 부분은 'e<번호>.pcx'의 형태로 저장하였다. (그림 11)과 (그림 12)는 각각 국문 초록 영상과 영문 초록 영상의 예를 보여주고 있다.



(그림 11) 국문 초록 영상 예(h0050.pcx, 1:5)



In the database context, the meaning of null value is interpreted as unknown and inapplicable. The aim of this paper is to provide the practical and consistent means for deriving the information about the unknown null value from the constraints existing in the database.

For this we introduce the implicit value which is associated with the null value by one to one correspondence. This implicit value can evolve by reflecting the constraints on itself.

In this paper, we will show how the constraints can be reflected on this implicit value in the consistent way by employing the functional dependency and the interrelational integrity as the example constraints in the database

(그림 12) 영문 초록 영상 예(e0050.pcx, 1:5)

분할된 두 가지의 영상 중에서 한글 초록 부분은 스피드리더 문자인식기를 이용하여 인식하였고, 그 결과를 텍스트 화일로 저장하였다.

화일의 구성 방식은 정보 검색용 텍스트 데이터베이스로서 mark-up 유형이다. KT 테스트 컬렉션의 포맷과 동일하며, <id>와 <abstract>만으로 구성된다. <id>와 <abstract>가 데이터베이스의 필드 명이며, 일련번호 및 초록 내용이 각 레코드의 필드 값이 된다. (그림 13)에서는 문자 인식된 결과로 만들어진 텍스트 화일의 일부를 보여주고 있다.

```
<id>0049
<abstract> 언어 시스템의 ERA모"♣ 구성과 메
타정 의로 부터 Knowledge Data B2Se를 구싶
하이' 운용되♣♣ 흐로그램으--1- 부터 소프
트웨어 도구 SE니을 이용하여 자동-♣--♣-!
♣서 추출아였다. 본 논문에서는 )"AxCAL으!
어시스댐을 대 상으로 하여 ERA모델, 메타정
의, 필터피 원영 및 생성핀 -♣♣서를 보였다.

<id>0050
```

<abstract> 데이터베 이스적 견 지에서 null 값의 의 미는 '"적용값이 알려 지 지 않았다"' 와 "값을 적용할 수 없다" -♣ 가 지로 해석된다. 본 논문에서는, 데 이타베이스내에 존재하는 치약들로부터 적용값이 알려지 지 않은 mll값의 정 보를 유추하기 위한 실시적 이고 일관성 있는 방식 으로 잠재값 개념을 제시한다. 해당 n빌l 값에 1 : 1로 연관되 도록 정의한 이 잘재값은 치약들을 자체값에 반영해 감으로써,연관되어 있는 nlll 값을 점 하 구체하시키게 된다. 본 논문에서는 데이터베이스내 제약 으로 함수적 의존관계와 릴러l 이션간의 무결성 두가지를 예로 들어 어 게 이 두 치약들이 일관성 있게 잠재값에 반영 되는가를 보인다.

(그림 13) 문자 인식 텍스트 화일의 예

4.3 문자 인식 데이터베이스의 인식률

문자 인식 분야에서 인식률을 측정하는 방법에는 문자 단위 인식률과 어절 단위 인식률의 두 가지가 있다. 문자 단위 인식률은 낱개 문자들을 대상으로 전체 문자 중 몇 %의 문자가 정확하게 인식되었는가를 측정하며, 어절 단위 인식률은 띄어쓰기에 의해 분리된 문장 내의 각 어절을 대상으로 전체 어절 중 몇 %의 어절이 정확하게 인식되었는가를 측정하는 방법이다. 본 논문에서는 문자단위 인식률을 측정하였다.

KT 테스트 컬렉션에 포함된 문서들을 대상으로 문자 인식 데이터베이스를 구축하였기 때문에 원문 텍스트에 대한 데이터베이스 화일이 존재한다. 따라서 문자 단위의 인식률 측정은 UNIX에서 사용하는 ***diff*** 명령어를 사용하여 편리하게 구할 수 있다. 본 논문에서 사용한 인식률 측정 알고리즘은 다음과 같다.

A를 문자 인식 시스템에 의해 인식된 결과 파일이라 한다;
B는 KT 테스트 컬렉션에 포함된 오자가 없는 원문 파일이라 한다;
A에서 둘 이상의 연속적 공백 문자를 하나의 공백 문자로 치환한다;
A, B의 개항 문자를 공백 문자로 치환한다;
A를 문자 단위로 읽어 한 라인에 한 문자만 써지도록 A'에 기록한다;
/* ***diff*** 명령어는 두 파일을 라인 단위로 비교하여 다른 부분을 찾아낸다. 문자 단위 비교를 위해서는 한 라인에 한 문자가 들어가도록 하여야 한다. */
B를 문자 단위로 읽어 한 라인에 한 문자만 써지도록 B'에 기록한다;
diff A' B'을 수행하고 결과를 파일 Diff에 기록한다;
Diff의 각 라인에 대해
  If 첫문자 = 'a' /* 새로운 문자의 삽입 */
    무시한다;
  If 첫문자 = 'd' /* 문자 유실 */
    유실된 문자의 수를 오류의 수에 더한다;
  If 첫문자 = 'c' /* 문자 치환*/
    치환된 문자의 수를 오류의 수에 더한다;
문서 B의 인식률 = (문서 B'의 문자 수 오류의 수) *100 / 문서 B'의 문자 수

본 연구에서 구축한 문자 인식 텍스트 데이터베이스의 인식률별 돗수 분포는 <표 2>와 같으며, 평균 인식률은 90.54%였다. 90.54%의 문자 인식률은 결코 만족할 만한 수준이라 말할 수 없다. 특히, <표 2>에서 알 수 있듯이 데이터베이스 수집에 사용한 문자 인식기의 특성, 특정 폰트나 문자 간격, 인쇄 품잘, 스캐닝 상태 등에 따라 인식률이 거의 70%에도 미치지 못하는 데이터도 존재한다. 현저하게 낮은 인식률을 보이는 문서의 경우 인식 과정에서 따로 분리하여 수작업에 의한 입력으로 전환하는 것이 바람직하다.

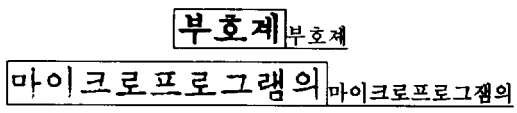
〈표 2〉 문자 인식 데이터의 인식률별 돗수 분포

인식률 (%)	데이터갯수
0~10	2
10~20	0
20~30	1
30~40	0
40~50	5
50~60	6
60~70	17
70~80	81
80~90	208
90~100	649
합계	970

4.4 문자 인식 오류의 유형

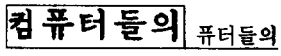
문자 인식 과정에서 발생하는 오류의 유형은 수작업으로 입력된 오류의 유형과는 그 종류를 달리 한다. 수작업된 데이터베이스에서 나타나는 오류에는 키보드를 잘 못 쳐서 발생하는 오타 오류와 본문의 단어를 잘 못 읽어서 발생하게 되는 오류의 두 종류가 대종을 이룬다. 반면에 문자 인식 데이터베이스에서 나타나는 오류는 빈칸이 없는 곳에 빈칸을 집어넣거나 하나의 문자를 둘로 나누어 인식하는 등 사람이 생각치 못하는 유형이 자주 발생한다. 문자 인식 기술을 사용하여 데이터베이스를 구축하는 과정에서 발생한 오류의 유형은 다음과 같다.

- 문자가 다른 문자로 인식되는 경우 : (그림 14)와 같이 문자를 잘못 인식하여 다른 문자로 치환된 경우로서 가장 많이 발생하는 오류의 유형이다.



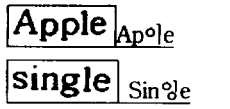
(그림 14) 문자가 다른 문자로 인식되는 경우

- 문자가 인식되지 않고 없어지는 경우 : 흔하지 않은 경우로서 문자가 인식되지 않은 채로 없어져버린 경우이다. (그림 15)는 '컴'자가 인식되지 않고 없어진 경우를 보여주고 있다.



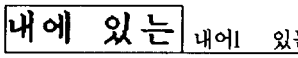
(그림 15) 문자가 인식되지 않고 없어지는 경우

- 없는 문자가 나타나는 경우 : 잡음에 의하여 많이 발생하는 경우로서, 잡음이 있는 부분을 문자로 인식하는 경우이다. 굉장히 드물게 나타나는 경우이다.
- 연속된 여러 문자가 다른 문자로 인식되는 경우 : 문자 분할이 잘못되어 여러 개의 연속된 문자가 하나의 문자로 합쳐져서 인식된다. 영어와 한글을 섞어서 사용한 문서에서 자주 발생한다.



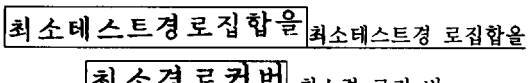
(그림 16) 연속된 여러 문자가 다른 문자로 인식되는 경우

- 하나의 문자가 연속된 여러 문자로 인식되는 경우 : 문자 분할의 잘못으로 하나의 문자가 여러 개의 문자로 인식되어 나타나는 경우이다. 역시 영어와 한글을 섞어서 사용한 문서에서 자주 발생한다.



(그림 17) 하나의 문자가 연속된 여러 문자로 인식되는 경우

- 공백 문자가 추가되는 경우 : 공백이 아닌 부분을 공백으로 인식하는 오류로서 정보 검색에 많은 영향을 줄 수 있는 오류이다.



(그림 18) 공백 문자가 추가되는 경우

5. 결 론

본 논문에서는 문자 인식 기법을 이용하여 활자화되어 있는 방대한 문서로부터 데이터베이스를 효율적으로 구축하기 위하여, 현재 사용 가능한 문자 인식기가 어느 정도의 인식률을 보이고 있으며, 여러 유형의 문서를 인식할 때 어떤 특성을 고려하여 인식기를 선택하여야 할 것인가에 대한 조사를 수행하였다. 이와 병행하여 현재 사용중인 대표적인 상용 문자 인식 시스템을 대상으로 다양한 유형의 문서에 대하여 인식 능력을 비교 검토하였다. 이 때 사용된 실험 자료는 KT 테스트 컬렉션에 포함된 970편의 논문 요약문을 MICRO-TEK MRS-1200ZS 스캐너를 사용하여 400 DPI의 정밀도로 스캔함으로써 구축되었다.

비교 대상이 된 4종의 문자 인식 시스템 중 가장 우수한 인식률을 보인 인식 시스템을 선택하여 본 연구에서 구축한 문자 인식 데이터베이스에 대한 평균 인식률을 조사하였으며, 실험 결과 평균 90.54%의 문자 인식률을 보여, 수작업된 데이터베이스의 인식률이나, 해외의 문자인식기 인식률에 크게 못미치는 것을 알 수 있었다. 또한 문자 인식기의 특성, 특정 폰트나 문자 간격, 인쇄 품질, 스캐닝 상태 등에 따라 인식률이 거의 70%에도 미치지 못하는 데이터도 존재한다. 현저

하게 낮은 인식률을 보이는 문서의 경우 데이터베이스로서의 역할을 수행할 수 없으므로 인식 과정에서 따로 분리하여 수작업에 의한 입력으로 전환하는 것이 바람직하다.

문자 인식된 텍스트에서 발생하는 오류의 유형은 수작업에 의해 입력되었을 경우 발생하는 오류의 유형과는 상이한 특성을 보인다. 본 논문에서는 문자 인식 텍스트에서의 오류 유형을 조사하여 분류함으로써 추후 데이터베이스의 품질을 높이는데 유용하게 사용될 수 있도록 하였다. 또한, 이 자료는 문자 인식 데이터베이스에 대한 검색 엔진 개발 시에도 유용한 참고 자료로 사용될 것이 기대된다.

참 고 문 헌

[1] 김성혁 외 5인, "자동 색인기 성능 시험을 위한 Test Set 개발", 정보관리학회지 제11권 제1호, pp.81-101, 1994.

[2] 이성환, "오프라인 필기체 문자 인식 기술의 현황 한글 인식을 중심으로", 정보과학회지 제11권 제5호, pp.51-65, 1993.

[3] D. Haskin, B. Gottesman, S. Plain, D. Jecker, and J. Morris, "Not all OCR software is created equal," internet document, http://www.zdnet.com/pcmag/features/ocr/_open.htm(PC Magazine, 1999. 01).

[4] 최정훈, "우리나라의 문자인식 수준은 몇점?", 윈도우세계, 95년 1월호, pp.230-239.

[5] 김명진, 장원식, "OCR 소프트웨어 4종 한글, 영어 인식 능력 완벽 테스트", Hello-PC, 1997년 5월호, pp.368-383.

[6] "OCR 패키지 하나 둘 셋", 컴퓨터매거진, 1994년 8월호, pp.150-166.

[7] 박승현, "문자인식 어디까지 왔나", 컴퓨터매거진, 1996년 5월호, pp.218-227.

[8] 강은영, 김민수, 김우성, 한선화, 김진형, "오프라인 인쇄체 문자인식기 구현 및 성능 비교에 관한 연구", 정보과학회 논문지 제출 중.

[9] 이준호, 이충식, 한선화, 김진형, "문자인식에 의해 구축된 한글 문서 데이터베이스에 대한 정보검색", 정보처리학회논문집 제 6권 4호, pp.833-840, 1999.

한 선 화

e-mail : shhahn@kordic.re.kr

1987년 성균관대학교 정보공학과 (학사)

1989년 한국과학기술원 전산학과 (석사)

1997년 한국과학기술원 전산학과 (박사)

1997년~현재 연구개발정보센터 선임연구원

관심분야 : 데이터베이스/마이닝, Intelligent Tutoring, 에이전트, HCI

이 준 호

e-mail : joonho@computing.soongsil.ac.kr

1987년 서울대학교 전산학과(학사)

1989년 한국과학기술원 전산학과 (석사)

1993년 한국과학기술원 전산학과 (박사)

1993년~1994년 한국과학기술원 인공지능연구센터 연구원

1994년~1997년 연구개발정보센터 선임연구원

1997년~현재 숭실대학교 컴퓨터학부 부교수

관심분야 : 정보검색, 정보시스템, 데이터베이스

이 충 식

e-mail : cslee@ai.kaist.ac.kr

1994년 한국과학기술원 전산학과 (학사)

1996년 한국과학기술원 전산학과 (석사)

1996년~현재 한국과학기술원 박사과정

1998년~현재 동경공과대학 방문학생

관심분야 : 패턴인식, Neural Network, Genetic Algorithm

김 진 형

e-mail : jkim@cs.kaist.ac.kr

1971년 서울대학교 공과대학(학사)

1973년~1976년과학기술연구소(KIST) 전산실 연구원

1976년~1977년 미 California State,

도로국, 프로그래머

1979년 UCLA 전산학과(석사)

1981년~1985년 Hughes Research Center, Malibu, Senior Computer Sceintist

1983년 UCLA 전산학과(박사)

1990년~1991년 미 IBM Watson Research Center 초빙연구원

1985년~현재 과학기술원 전산학과 교수

1991년~현재 과학재단 지정 과학기술원 인공지능 연구센터 부소장

1995년~현재 출연(연) 연구개발정보센터 소장

1997년~현재 공학한림원 회원

관심분야 : 문자인식, 지능형 인터페이스, 인공지능