

# 프로젝션 화면 내에서의 Advanced Paper Document

홍광진<sup>0</sup> 정기철

송실대학교, 정보과학대학, 미디어학부  
{hongmsz<sup>0</sup>, kcjung}@ssu.ac.kr

## Advanced Paper Documents in a Projection Display

Kwangjin Hong<sup>0</sup> Keechul Jung

School of Media, College of Information Science, Soongsil University

### 요 약

최근 컴퓨터 환경은 매우 빠르게 발전하고 있지만, 사람들은 여전히 종이로 된 문서를 사용한다. 본 논문은 프로젝션 화면 내에서 종이 문서의 기능을 향상시킨 Advanced Paper Document (APD)를 제안한다. APD는 종이 문서와 디지털 문서를 결합시키기 위해서 부가적인 장치 없이, 사용자의 손을 이용한 문서의 선택과 조작 등의 다양한 명령을 수행한다. APD는 종이 문서와 디지털 문서의 장점을 모두 가지고 있으며, 사용자에게 자연스럽고 직관적인 환경을 제공한다. 실험 결과를 통해서, APD가 종이 문서와 디지털 문서의 기능을 결합하는 데에 적합하며, 손의 추출 및 인식과 문서의 검색에서 만족스러운 성능을 보여줌을 확인하였다.

### 1. 서 론

종이로 만들어진 오프라인 문서는 가격이 저렴하고, 휴대가 간편하며, 디지털화 된 온라인 문서에 비해 가독성이 높다는 장점 때문에, 컴퓨터 환경이 발달하고 있음에도 불구하고 여전히 가장 보편적으로 사용되고 있다. 반면에 온라인 문서는 오프라인 문서보다 많은 정보를 가지고 있으며, 각각의 정보를 다양한 방법으로 표현할 수 있고, 문서의 검색이 간편하다는 장점을 가지고 있다. 따라서 최근 오프라인 문서에 온라인 문서의 기능을 더해줌으로써, 두 문서의 장점을 모두 가지도록 하기 위한 연구가 다양하게 이루어지고 있다.

Sato 등이 제안한 오그먼트데스크 (AugmentedDesk)[1]나 Koike 등이 제안한 인핸드데스크 (EnhancedDesk)[2] 등은 두 문서의 장점을 통합하기 위한 연구의 한 예라 할 수 있다. 디지털 데스크(DigitalDesk) [3]와 유사한 이들 시스템은 우선, 사용자가 보고자 하는 오프라인 문서의 특정 위치에 부가적인 마커를 부착함으로써, 사용자에게 해당 문서에서 보여주고자 하는 다양한 정보들을 원하는 위치에 표시해 줄 수 있다. 하지만, 이러한 시스템은 마커가 부착되지 않거나, 다른 물체에 의해 마커가 가려진 오프라인 문서에 대해서는, 사용자가 선택을 하여도 시스템에서 인식을 하지 못한다는 단점이 있다.

또 다른 연구로는, Sato 등이 제안한 오그먼트데스크 상에서의 물체의 인식에 대한 연구[4]가 있다. Sato 등이 제안한 방법은 앞서 설명한 시스템과 달리, 오프라인 문서에 마커를 부착하지 않고 사용자의 손을 이용하여 문서 내의 영상을 선택하여 등록하거나, 검색함으로써 사용자에게 보다 자연스러운 컴퓨팅 환경을 제공할 수 있다. 하지만 이들은 선택이 가능한 영역을 60×60 픽셀의 정사각형 영역으로 한정하였기 때문에 다양한 형태와 크기의 문서를 검색하지 못한다는 단점을 가지고 있다.

본 논문에서 우리는 Advanced Paper Document (APD)를 제안한다. 우리가 제안하는 APD도 역시 오프라인과 온라인 문서, 두 가지 모두의 장점을 가지고 있다. 반면에, 앞에서 살펴본 시스템들과 달리 마커, 태그 등의 부가적인 장치를 사용하지 않고, 프로

젝션 화면 내에서 온라인과 오프라인 문서를 결합시킨 상호 작용성을 가지는 컴퓨팅 환경을 제공한다. 이러한 환경을 제공하기 위해서, APD는 기하 보정과 색상보정 및 전경 물체의 3차원 정보를 이용하여 손을 검출하고, 검출된 영상에서 20개의 특징 값을 추출한 뒤, k-NN 방법을 사용하여 입력된 손의 모양을 인식한다. 이렇게 추출, 인식된 손을 마우스와 같은 포인팅 장치처럼 사용하여 프로젝션 디스플레이 시스템 (Projection Display System: POS) 내에 위치한 다양한 크기의 문서를 선택한다. 문서가 선택되면, "정보 보기", "캡처하기", "문서 수정", "문서 스캔" 등의 명령이 들어 있는 "메뉴 상자"가 오프라인 문서 옆에 보여지게 되는데, 사용자는 역시 손을 사용하여 원하는 메뉴를 선택하고, 해당 메뉴에 대한 명령을 실행시킨다.

### 2. APD

본 논문은 추가적인 장치를 사용하지 않고 사용자의 손만을 사용하는 APD를 제안한다. APD를 사용함으로써, 사용자는 자신이 선택한 종이 문서에 대한, 종이 문서에 담을 수 없는 다양한 정보를 실제 문서 위에 겹쳐서 볼 수 있는 직관적이면서도 자연스러운 환경을 제공받게 된다.

본 논문에서 제안하는 시스템에서, 손은 마우스와 같은 포인팅 장치의 역할을 한다. 따라서 프로젝션 화면 내에서의 손의 검출이 우선적으로 이루어져야 한다. 빛의 밝기와 색이 지속적으로 변화하는 프로젝션 화면 내에서 손을 검출하기 위하여, 기하 보정과 색상 보정 및 전경 물체의 3차원 정보를 사용한다. 손을 검출한 후에는, 문서의 선택이나, APD 시스템에서 제공하는 명령의 수행을 위해서, 손 모양을 인식하는 과정이 필요하다. 입력된 손의 모양을 인식하기 위해서, 손 검출 결과 영상에서 20개의 특징 값을 추출한다. 이렇게 추출된 특징 값들은 k-NN 방법에 의해 분류되어, 입력된 손의 모양을 결정하게 된다. 시스템에서 손의 위치와 모양을 인식하면, 사용자는 PDS 내에서 자신이 원하는 문서를 선택할 수 있다. 선택된 문서는 X-Y recursive cut 알고리즘을 사용하여, 단락 별로 분할되고, edit distance와 pixel matching 알고리즘을 사용하여, 분할된 문서 영상을 미리 스캔 하여 저장해둔

문서 영상과 비교한다. 최종적으로, APD 시스템은 해당 문서와 가장 유사도가 높은 문서를 선택하여, 그 문서에 대한 메뉴 상자를 보여주고, 사용자는 원하는 메뉴를 선택하여 그에 해당하는 명령을 수행할 수 있다.

**2.1 손 검출**

손을 포인팅 장치로 사용하기 위해서는, 프로젝션 화면 내에서의 손의 좌표가 필요하다. 손의 좌표를 측정하기 위해, 프로젝션 화면에서 손을 검출한다. 본 논문은 기하 보정과 색상 보정을 사용하여 전경 물체(손)를 검출한다. 이 방법은 프로젝터로의 입력 영상(프레임 버퍼 영상)과 카메라를 사용하여 입력 받은 영상(카메라 영상)을 사용한다. 카메라 영상 내의 왜곡 중에서 가장 큰 영향을 미치는 왜곡 두 가지를 기하 왜곡과 색상 왜곡이라고 가정한다[5].

기하 왜곡을 해결하기 위해서, Ashdown 등[6]이 제안한 좌표 축 변환을 위한 투영 변환을 사용하고, 색상 왜곡을 해결하기 위해서, 프로젝션 된 화면의 상단에 위치하는 밝기 마커(brightness color markers)를 이용한다. 밝기 마커는 10단계의 회색 값을 가지는 10개 블록으로 이루어져 있다.

보정된 영상을 사용하여, 보다 정확한 전경 물체의 검출을 위해서, 전경 물체의 3차원 정보를 이용한다. 전경 물체에 대한 3차원 정보는 스테레오 카메라에서 가까울수록 밝게, 멀수록 어둡게 디스퍼리티 맵에 표현된다. 디스퍼리티 맵을 이용하면, 색상 기반의 전경 물체 검출 결과를 향상시킬 수 있다. 이러한 과정을 거쳐서 전경물체(손)를 검출한다.

프로젝션 화면 내에서 문서를 선택할 때, 사용자는 양손을 모두 사용한다. 전경물체 검출 영상에서 나타나는 두 손은 서로 겹치지 않으므로, 각각 Minimum Boundary Rectangle (MBR)로 묶어줌으로써 두 손을 분리할 수 있다. 이렇게 분리된 두 손 중에서, 오른손 영역의 왼쪽 면에서 가장 가까운 지점과 왼손 영역의 오른쪽 면에서 가장 가까운 지점이, 문서를 선택하는 기준이 되는 손가락의 위치와 동일하다는 점을 이용하여, 손가락의 위치를 찾는다.

**2.2 손 모양 인식**

문서나 메뉴 등을 선택하기 위해서 APD는 두 가지의 오른손의 모양을 사용한다. 이전 단계에서 분리된 오른손 영역에서, 손 모양의 구분을 위한 특징값으로 사용하기 위해, 그림 1에서처럼 중, 횡 방향의 경정 픽셀의 런헝스(run-length)를 구한다. 손 모양이 무엇인지 인식하기 위해, 우리는 k-NN 방법을 사용한다.

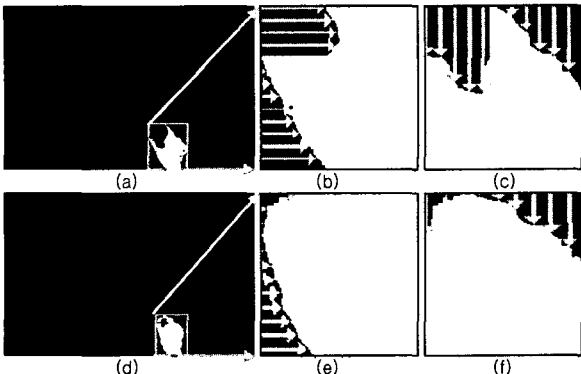


그림 1. 두 종류의 손 모양: (a) 기본 형태, (b) (a)의 횡 방향 경정 픽셀 런헝스, (c) (a)의 종 방향 경정 픽셀 런헝스, (d) "선택" 명령을 위한 손 모양, (e) (d)의 횡 방향 경정 픽셀 런헝스, (f) (d)의 종 방향 경정 픽셀 런헝스.

그리고, 그림 2에서처럼 클러스터링을 이용하여 두 모양을 구분한다. 그림 2(a)와 그림 2(b)에서 위쪽의 밝은 색으로 표시된 그래프는 그림 1(a)와 같은 기본 형태의 손 모양에 대한 특징 값의 분포를 나타내고 있고, 아래쪽의 어두운 색으로 표시된 그래프는 그림 1(d)와 같은 "선택"을 나타내는 손 모양에 대한 특징 값의 분포를 나타낸다. 카메라를 통해서 영상을 입력 받으면, 손에 대한 특징 값을 추출하고, 각각의 템플릿의 최 근접 값과 비교하여, 가장 많은 개수의 특징 값이 분류되는 부류를 입력된 손의 모양으로 결정한다.

분포를 나타내고 있고, 아래쪽의 어두운 색으로 표시된 그래프는 그림 1(d)와 같은 "선택"을 나타내는 손 모양에 대한 특징 값의 분포를 나타낸다. 카메라를 통해서 영상을 입력 받으면, 손에 대한 특징 값을 추출하고, 각각의 템플릿의 최 근접 값과 비교하여, 가장 많은 개수의 특징 값이 분류되는 부류를 입력된 손의 모양으로 결정한다.

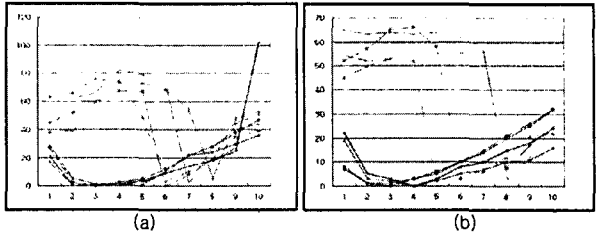


그림 2. 특징값의 클러스터링: (a) 종 방향 경정 픽셀 런헝스, (b) 횡 방향 경정 픽셀 런헝스.

**2.3 문서 검색**

문서를 검색하기 위한 기존의 연구는, 크게 OCR을 사용하여 글자를 인식하고 이것을 비교하는 방법과, OCR을 사용하지 않고 문서 영상의 구조를 비교하는 방법으로 나눌 수 있다. OCR을 사용하지 않는 방법은 또한, 고해상도 영상을 사용하는 경우와 저해상도 영상을 사용하는 경우의 두 가지로 나눌 수 있다. 고해상도 영상을 사용하는 경우에는 글자를 구분할 수 있다. Peng 등[7]이 제안한 컴포넌트 블록의 비교를 통한 문서 검색 방법이 여기에 해당한다. 반면에 저해상도 영상을 사용하면, 글자를 구분하지 못하고, 단어의 길이나 문서의 구조만을 구분할 수 있다. Hu 등[8]은 레이아웃의 유사성을 기반으로 하여 검색하는 방법을 사용하였다.

본 논문은 저해상도의 이진 영상을 사용하여 문서를 검색한다. 입력 영상은 기하 보정과 색상을 보정을 통해 왜곡을 보정한 카메라 영상의 이진영상을 사용하고, 입력 영상과의 비교에 사용할 영상은 스캐너를 사용하여 저장시킨 문서의 이진영상을 사용한다. 문서의 검색에 앞서서, X-Y recursive cut 알고리즘[9]을 사용하여, 검색 영상과 카메라 영상에서 문자영역을 분할한다.

이렇게 분할된 각각의 컴포넌트들을 비교함으로써, 사용자가 선택한 문서와 가장 유사한 기준에 저장된 문서를 검색한다. 문서의 유사도를 구하기 위해 두 가지의 방법을 사용하고, 두 방법의 성능과 속도를 비교한다. 두 방법에 대한 비교 결과는 3장의 실험 및 결과에서 설명한다.

첫 번째 방법은, 입력 받은 영상과 저장되어있는 영상의 서로 대응하는 픽셀을 비교하는 방법이다. 먼저 두 영상을 비교할 때 발생할 수 있는 오차를 줄이기 위해, 문서의 단락을 돌려싸고 있는 여백을 최소화한다. 그리고, 비교하는 두 영상의 크기를 일치시켜, 서로 대응하는 픽셀을 비교한다. 두 픽셀이 서로 일치할 때를 1, 그렇지 않을 때를 0 이라 두고, 비교를 통해 얻어지는 값을 모두 더하여 두 문서 사이의 유사도를 구한다.

두 번째 방법은 Hu 등[8]이 제안한 두 문서의 edit distance를 구하고 이를 이용하여 문서를 검색하는 방법이다. 이 방법은 입력 받은 문서 영상을  $m \times n$  행렬로 변환시켜, 두 영상의 서로 대응하는 행 사이의 edit distance를 구하여, 그 값을 모두 더하여 두 문서 사이의 유사도를 구한다. 본 논문에서는, 두 개의 문서 영상을  $19 \times 26$  행렬로 변환시켜 유사도를 구한다. 영상을 행렬로 변환시키기 위해서, 일정한 크기의 영역 안에, 검정색 픽셀이 50% 이상 포함되어있을 경우 1, 50% 이하일 경우 0 이라 하였다.

이러한 방법을 사용하여 유사도가 가장 높은 문서를 선택하고, 선택된 문서의 메뉴상자를 프로젝션 화면상의 종이문서 옆에 연다. 사용자가 메뉴상자의 한 메뉴를 선택했을 때, 시스템은 문서의 정보나 문서와 관련된 참조목록을 보여주거나 문서의 선택된 영역을 스캔 하는 등의 명령을 수행한다.

3. 실험 및 결과

제안된 방법은 실험 환경 내에서 만족할 만한 성능을 보여주었다. 실험에 사용된 시스템은 프로젝션 된 화면과 손, 종이 문서 등을 입력 받기 위한 캠코더, 전경물체에 대한 3차원 정보를 얻기 위한 스테레오 카메라, 오프라인 문서에 온라인 정보를 붙여주기 위한 프로젝터, 책상, 그리고 입력 받은 영상들을 이용하여 원 하는 정보를 얻어 내기 위한 일반 PC로 이루어져있다. 영상처리시스템은 Intel® Pentium4 2.66Hz CPU와, ATI® Radeon 7000으로 구성 되어 있고, 프로젝션 화면은 BenQ® HD2100을 사용하여 1152x864 픽셀의 32비트 트루 컬러 영상을 사용한다. 카메라 영상은 Sony® DCR-VX2000를 사용하여 입력 받고, 디스패리티 맵은 Point-Grey® Digidlops Vision System을 사용하여 입력 받는다. 문서는 HP® Scanjet 5P scanner를 사용하여 스캔 받는다. 실험에 사용 된 시스템은 Microsoft® Visual C++ + 6.0와 DirectX 9.0 SDK를 사용하여 구현되었다.

본 논문에서 제안한, APD 시스템은 초당 5프레임의 속도로 화면을 캡처하고, 입력된 영상을 분석하여, 사용자가 선택한 문서를 검색하였다. 2.3절에서 이야기한 것과 같이, 본 논문은 두 가지의 문서 검색 방법을 사용하고, 두 방법의 속도와 성능을 비교하였다. 그림 3에서처럼, 두 방법 모두 유사한 문서 검색에서 만족할 만한 결과를 보여주었다.

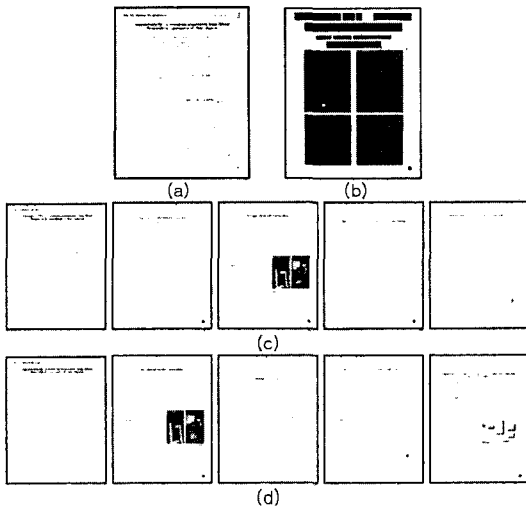


그림 3. 문서 검색 결과: (a) 입력 영상, (b) (a) 문서 분할 영상, (c) pixel matching 방법을 사용한 결과, (d) edit distance 방법을 사용한 결과.

수행 시간 면에서는, 표 1과 같이, edit distance를 사용하는 방법이 평균 567ms로, pixel matching을 사용하는 방법의 수행 시간인 평균 740ms 에 비해 더 빠르게 문서를 검색하였음을 알 수 있다. 다이나믹 프로그래밍을 이용한 edit distance를 구하는 방법이 단순한 pixel의 차이를 계산하는 방법보다 두 문서를 비교 시간이 적게 걸린 이유는, edit distance를 구하기 위해 혼합된 문서 영상을 행렬로 표현하면서, 두 문서 사이의 비교해야 할 값의 개수가 pixel matching에 비해 1/10로 줄어들었기 때문이다.

표 1. 문서 검색 방법의 단계별 평균 수행 시간 (1/1000초)

	입력 영상 변환	저장된 문서(44개)와 비교	계
Edit distance	171	396	567
Pixel matching	168	572	740

다음의 그림 4는 사용자가 문서를 선택하고, 메뉴 상자에서 그 문서에 대한 저자, 제목, 참고문헌 등의 요약된 정보를 보여주는 메뉴를 선택하여, 그 결과를 보기까지의 과정을 보여준다.

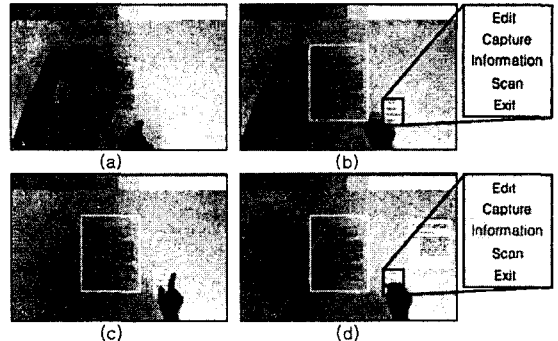


그림 4. Advanced Paper Document: (a) 일반적인 손 형태, (b) 문서의 선택과 메뉴 상자, (c) 메뉴의 선택, (d) 메뉴 상자 명명 실행 ("정보보기")

4. 결론

본 논문은 APD를 제안하였다. 사용자에게 자연스럽게 직관적인 컴퓨팅 환경을 제공하기 위하여, APD는 손을 마우스와 같은 포인팅 장치로 사용하였다. 손을 포인팅 장치로 사용하기 위해서, 전경물체(손)의 좌표와 손의 모양을 인식하는 것이 필요하고, 또한 사용자가 선택한 오프라인의 문서에 온라인 정보를 첨가를 하기 위해서, 해당 문서와 일치하는 정확한 문서의 검색이 필요하다. 본 논문에서 제안한 손의 위치 검출과 손 모양의 인식, 그리고 문서의 검색 방법은 실험의 결과를 통하여 만족스러운 성능을 보여 줌을 확인할 수 있었다.

반면에, 실험 결과에서 이야기한 것처럼, 현재의 시스템은 사용자에게 초당 5프레임의 속도 화면을 업데이트하기 때문에 문서나 메뉴를 선택할 때 약간의 딜레이가 발생한다. 추후 연구를 통해, 시스템의 속도 문제를 해결하고, 사용자에게 보다 자연스러운 User Interface를 제공할 수 있는 방법을 제안할 것이다.

5. 참고문헌

- [1] Sato Y., Kobayashi Y., Koike H., "Fast Tracking of Hands and Fingertips in Infrared Images for AugmentedDesk Interface," The 4th AFGR, 462-467, 2000
- [2] Koike H., Sato Y., Kobayashi Y., "Integrating Paper and Digital Information on EnhancedDesk: a Method for Real-time Finger Tracking on AugmentedDesk System," CHI 2001, Vol. 8, Issue. 4, 307-322, 2001
- [3] Wellner P., "The DigitalDesk Calculator: Tactile Manipulation on a Desktop Display," ACM Symposium on User Interface Software and Technology, 27-33, 1991
- [4] Sato Y., Kobayashi Y., Koike H., "Interactive Object Registration and Recognition for Augmented Desk Interface," ACM SIGCHI 2001, 371-372, 2001
- [5] Kang H., Kim S., Lee C., Jung K., Park M. H., "Foreground Object Detection in Projection Display," Journal of the Institute of Electronics Engineers of Korea, Vol. 41-CI, No. 1, 27-37, 2004.
- [6] Ashdown M., Robinson P., "The Escritoire: A Personal Projected Display," Journal of WSCG, Vol.11, No. 1, 33-40, 2003
- [7] Peng H., Long F., Siu W.C., Chi Z., Feng D. D., "Document Image Matching based on Component Blocks," ICIP, Vol. 2, 601-604, 2000
- [8] Hu J., Kashi R., Wilfong G., "Comparison and Classification of Documents based on Layout Similarity," Information Retrieval, Vol. 2, No. 2/3, 227-243, 2000
- [9] Jung K., Han J., "Hybrid Approach to Efficient Text Extraction in Complex Color Images," Pattern Recognition Letters, Vol. 25, Issue 6, 679-699, 2004