

지문원지의 영역분할 및 도표 인식

정 윤 주*, 이 영 화*, 이 준 제**, 이 주 형***, 심 재 창*
*안동대학교 컴퓨터공학과, **동서대학교, ***(주)한국정보시스템
전화 : (0571) 850-5713 / 팩스 : (0571) 850-5480

Fingerprint region and table segmentation in fingerprint document

Yun Ju Chung*, Young Hwa Lee*, Jun Jae Lee**, Ju Hyung Lee** and Jae Chang Shim*

*Dept. of Computer Eng. Andong National Univ.,

Dept. of Computer Eng. Dongseo Univ., *Korea Infomation System.

E-mail : chungyj@andong.ac.kr

Abstract

In this paper, a method for extracting the fingerprint regions and the table from fingerprint document which is the size of A4 including ten fingerprints images in a table is presented. The extraction of each fingerprint region is carried out by segmenting the foreground fingerprint region using a block filtering method and detecting its center point. The table extraction, by detecting a horizontal line using line tracing, and detecting a vertical line by its orthogonal equation. Here, T-shaped mask is proposed for finding the starting points of the vertical line intersecting horizontal line by the form of 'T'.

Experimental results show above 95% correct rate of extracting the fingerprint region and table.

I. 서론

지문인식에 대한 연구는 전처리 단계와 인식단계로 나눌 수 있는데, 그 중 전처리(Preprocessing) 단계에서는 지문 영역을 배경에서 분리하는 분할과 지문 영상의 이진화, 특징점을 찾기 위해 적용하는 세션화 등이 있다[1]. 일반적으로, 주민등록번호, 성명, 지문분류코드 등의 문자 정보와 양손 10개의 지문이 날인되어 있는 지문원지에서 지문을 인식하기 위해서는 문서 전체를 스캐너로 읽은 다음, 원지 영상 내의 각 지문 영역들만을 고해상도로 다시 한 번 더 입력받기 위해서는 각 지문의 정확한 영역 추출이 요구된다. 본 논문에서는 원지 내의 지

문 영역의 중심 위치를 자동으로 검출하여 이 작업에서 소요되는 시간과 노력을 최소화하고자 한다.

각 지문 영역의 중심위치를 자동 검출하기 위해서 지문원지 내의 글자나 선, 잡음 등 불필요한 정보를 최대한 제거하고 검은 화소가 밀집되어 있는 지문 영역만을 배경으로부터 분리하고, 이 영역의 상하좌우 한계를 검출한 다음, 검출된 영역 내의 수직과 수평 방향의 거리를 평균하여 중심위치를 설정하는 방법을 적용한다. 특별히, 이 단계에 앞서, 처리 속도를 증가시키고 잡음과 글자 등 불필요한 정보를 1차적으로 제거하기 위하여 1600×1600 픽셀 크기의 영상을 400×400으로 축소하였다. 만약, 자동 검출된 각 지문 영역의 중심위치가 정확하지 않다고 판단되면 사용자가 개개의 지문 영역의 중심위치를 옮길 수 있도록 반자동 사용자 인터페이스도 함께 구현하였다.

또한, 본 논문에서는 지문을 포함하고 있는 도표에 대한 정보도 자동 검출할 수 있는 방법을 제안한다. 이 도표 정보의 추출은 첫째, 도표내의 지문 영상의 품질이 아주 낮아서 정확하게 지문의 위치를 찾지 못할 경우, 지문을 둘러싸고 있는 도표의 위치 정보를 검출하여 대략적인 지문의 위치를 결정하는데 이용할 수 있다. 둘째, 문서 내에서 주민등록번호나 지문 분류코드 등의 특정 문자 정보의 위치를 찾는 데도 이용될 수 있다. 마지막으로, 입력된 지문원지 영상이 기울어진 경우, 검출한 수평선의 기울기 정보를 이용하여 영상을 바로 세워줄 수 있다.

II. 지문 영역의 중심 찾기

지문원지 내에서 각 지문은 도표 안에서 왼손, 오른손 각각 1행에 다섯 개씩 날인되어 있는데, 이 지문들

을 배경으로부터 분리하고 각 지문 영역들의 중심위치를 찾는 것이 주된 목적이다. 지문 영역의 중심을 찾는 과정은 그림 1과 같다.

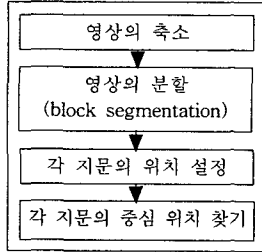
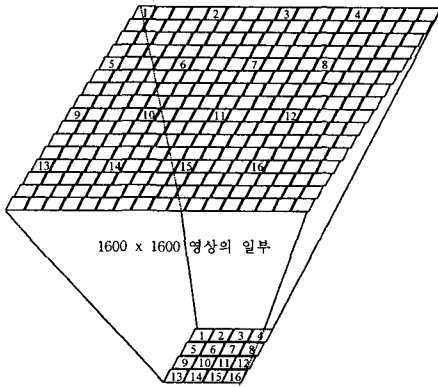


그림 1. 지문의 중심위치 찾기

2.1 영상의 축소

일반 문서에서 문자를 인식하는 작업과는 달리 지문 원지에서 지문 영역의 중심 위치를 검출하는 과정은 영상의 축소로 인한 손실에 거의 민감하지 않다. 따라서, 지문 영상을 배경으로부터 분리하기에 앞서 그림2와 같이 1600×1600 크기의 영상을 4×4 블록 단위로 나누어 그 첫 화소의 값만을 400×400 크기의 버퍼에 차례대로 줄이는 방법으로 영상을 축소하였다[6].



400 x 400 영상의 일부
그림 2. 영상의 축소 방법

2.2 영상의 분할(Segmentation)

지문의 위치를 파악하기 위해서는 영상 내에서 글자나, 선, 잡음과 같은 불필요한 정보는 제거하면서 관심이 있는 지문 영상의 품질은 향상시켜야 한다. 이 목적에 적합한 전처리작업을 수행하기 위해서 축소된 400×400의 영상을 다시 3×3 블록 단위로 나누어 9개 화소 중 흰 화소의 수가 6개 이상이면 그 블록 전체를

화소값 255로 채우고, 그렇지 않으면 화소값 0으로 채워서 문서 내의 글자나 도표, 잡음 등과 같은 불필요한 부분은 최대한 제거하고, 검은 화소가 밀집되어 있는 지문 영역만을 검출하는 방법을 제안한다. 이 과정에서 사용한 임계값 6은 다량의 데이터에 적용한 결과가 가장 우수한 경우로 표 1과 같다.

표 1. 임계값의 결정에 관한 실험 결과

Threshold	5	6	7
Result*	468/500	473/500	209/500
Success Rate(%)	93.6	94.6	41.8

*Result는 (성공한 지문의 수/실험한 지문의 수)임.

2.3 지문의 위치 검출 및 중심 위치 찾기

지문의 위치를 검출하기 위하여, 1차적으로 지문이 날 인되어 있는 도표에서 행을 기준으로 한 5개씩의 지문 중 최고 상한과 하한을 검출하고, 다음으로 열을 기준으로 한 2개씩의 지문의 좌측과 우측 한계를 찾는다.

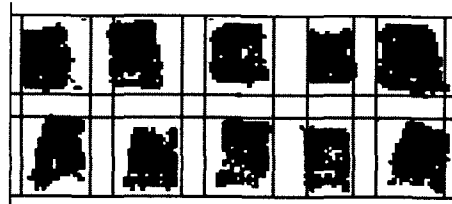


그림 3. 지문영상의 1차 영역 설정

그림 3과 같이 각 지문의 1차 영역이 검출되면, 각 지문 영역의 내부를 상하좌우에서 스캔하여 각 방향에서 제일 먼저 검출되는 검은 화소의 위치를 해당 지문의 상하좌우 한계로 설정하고, 각 지문 영역의 중심 좌표는 해당 지문 영역의 상하좌우 한계의 평균값으로 설정한다. 설정된 중심 위치를 기준으로 500DPI로 스캔할 경우 512×480픽셀의 영상이 될 수 있는 크기의 사각형을 그림 4와 같이 분할한다.

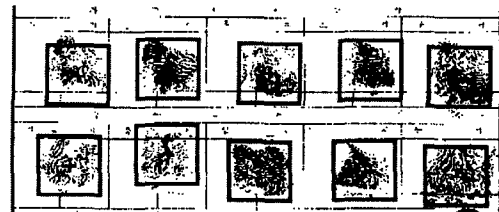


그림 4. 지문 영역의 중심점 찾기의 결과

III. 도표의 검출

일반적으로 지문은 도표 내의 해당 칸 안에 날인하도록 되어 있기 때문에, 지문원지 영상에서 지문을 포함하고 있는 도표에 대한 정보는 본 논문에서 찾고자 하는 지문 영역의 중심 위치와 상관관계가 매우 높다.

따라서 지문 영상의 품질이 낮아서 그 중심 위치를 제대로 찾지 못할 경우 지문이 날인되어 있는 도표에 대한 정보를 대신 이용할 수 있을 것이다. 도표에 대한 정보를 검출하는 과정에서는 먼저 직선을 추적하는 방법으로 도표를 구성하고 있는 수평선들을 검출한 후, 수평선에 직교하는 수직선들을 검출하였다.

3.1 수평선 검출

입력받은 지문원지의 지문 후보 영역 내에서 왼쪽 100 픽셀만큼의 폭에 대하여 다음과 같은 알고리즘으로 가능한 모든 수평선들의 위치를 찾는다.

지문 후보영역 내에서 왼쪽 100픽셀 만큼의 폭을 가로 방향 우선으로 스캔하면서 다음을 수행한다.

- step 1. 아래의 과정에 의해 발견한 검은 픽셀의 수를 나타내는 카운터를 0으로 설정한다.
- step 2. 만약 최초로 검은 픽셀이 발견되면,
 - 2.1 현재 위치를 기준 위치로 설정하고, 카운터를 1 증가시킨다.
 - 2.2 5픽셀 간격으로 계속해서 검은 픽셀이 발견되는 동안 이 단계를 반복한다.
 - 2.2.1 만약 카운터가 20미만이면 기준 위치를 가로 방향으로 5 증가, 카운터를 1 증가시킨다.
 - 2.2.2 아니면 반복 루프를 벗어난다.
 - 2.3 5픽셀 거리에서 검은 픽셀을 발견하지 못하고, 카운터도 20 미만이라면,
 - 2.3.1 계속, 6~10번째 픽셀을 조사하여 검은 픽셀을 발견하면, 기준 위치로 설정하고, 카운터를 1 증가시키고, 위 2의 단계부터 반복한다.
 - 2.3.2 기준 위치에서 바로 위쪽 라인의 5픽셀 떨어진 곳에서 검은 픽셀을 발견하면, 기준 위치로 설정하고, 카운터를 1 증가시키고, 위 2의 단계부터 반복한다.
 - 2.3.3 기준 위치에서 바로 아래쪽 라인의 5픽셀 떨어진 곳에서 검은 픽셀을 발견하면, 기준 위치로 설정하고, 카운터를 1 증가시키고, 위 2의 단계부터 반복한다.
- 2.4 카운터가 20개 이상이면,
 - 2.4.1 위 단계2에서 최초로 검은 점이 발견된 지점을 이 수평선의 시작점으로 하고,
 - 2.4.2 카운터가 20 이상이 될 때의 기준 위치를 이 수평선의 끝점으로 하고,
 - 2.4.3 선의 굵기가 평균 3픽셀 정도이므로 같은 선을 거듭 검출하지 않기 위하여, 스캔하기 위한 라인의 위치를 5만큼 증가시키고, 왼쪽 첫 번째 픽셀부터 위의 전 과정을 다시 반복한다.

다음 단계에서는 찾아낸 수평선 후보들만을 대상으

로 위와 방법은 동일하지만 추적 범위를 영상의 폭만큼으로 확장시킨 알고리즘을 적용하여 각 수평선 후보들의 길이를 조사한다. 확장 추적 결과, 이전 단계에서 찾아낸 수평선 후보들은 대부분 도표를 구성하고 있는 수평선의 일부일 가능성이 크지만, 그림 5와 같이 지문 원지 영상이 기울어진 채로 입력되어 도표의 가장 왼쪽 수직선이 없는 경우나, 그림 6과 같이 지문이 가장 왼쪽 수직선에 겹쳐서 날인된 경우 지문 영역에서도 수평선을 검출하게 된다. 실험에 사용된 400개의 지문원지 영상을 전부 조사한 결과 지문 영역에서 검출된 수평선의 경우, 그 길이가 300픽셀을 넘지 않는다는 것을 발견하여, 수평선의 길이가 300픽셀 미만인 것은 수평선에서 탈락시켰다.

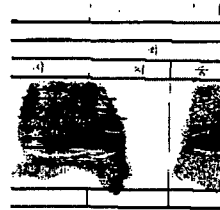


그림 5. 왼쪽 수직선이 없는 경우

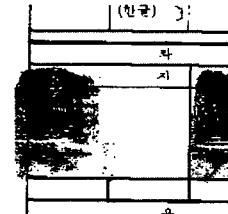


그림 6. 지문이 선에 겹쳐진 경우

3.2 수직선 검출

지문이 날인되어 있는 도표의 수직선은 일정 간격으로 계속해서 검은 픽셀을 추적한 수평선 검출과는 달리, 도표가 직사각형이라는 특성을 이용하여, 3.1절에서 검출한 수평선 중 가장 왼쪽에서 시작하는 수평선의 시작 좌표를 교점으로 그 수평선과 직교하는 직선의 방정식을 계산하여 첫 번째 수직선을 검출하였다. 나머지 수직선 검출을 위하여 영상을 분석한 결과, 도표를 구성하고 있는 선들의 굵기가 평균 3픽셀 정도임을 알 수 있다. 따라서 다음 그림 7과 같은 'T'자형 마스크를 제안하고, 이를 이용하여 수평선과 수직선이 'T'자 또는 '┌'형태로 만나는 점들을 검출하여, 이 점들을 교점으로 하며, 첫 번째 수직선과 기울기가 같은 직선의 방정식을 계산하였다.

1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0

그림 7. T자형 마스크

3.3 기울기 보정

지문원지가 스캐너로 입력받는 과정에서 기울어졌다면 위 3.1절에서 검출한 수평선의 기울기 정보를 이용하여 영상 안에 있는 픽셀들을 기울어진 반대 방향으로 회전시켜 픽셀들을 재배치하는 기하학적 변환을 이용할 수 있다. 기하학적 연산에는 두 가지 형태가 있는데, 첫 번째는 처리된 영상에 임의의 왜곡을 가져오지 않는 어파인(affine) 변환이 있고, 두 번째는 영상에 왜곡을 가져오는 굴곡(warping) 변환이 있다[4].

본 논문에서는 어파인(affine) 변환을 이용하여 영상을 원점을 기준으로 기울어진 반대 방향으로 회전시켰으며, 따라서 입력 영상에서의 직선은 출력 영상에서도 직선으로 남아 있다.

IV. 실험 및 고찰

실험에 사용된 입력 영상의 종류는 200DPI 해상도로 입력받은 Tiff 포맷의 지문원지 400개이며, 각 입력 영상은 1600×1600 크기의 이진 영상이다. 사용한 컴퓨터 기종 PentiumMMX-200 CPU, 32MB RAM이었으며, 컴파일 환경은 Visual C++ 6.0 Enterprise Ver.을 이용했다.

주민등록번호나 성명, 지문 분류 코드 등의 문자 정보와 양손 다섯 개씩의 지문이 두 줄로 날인되어 있는 지문원지 영상을 400매를 대상으로 실험을 수행한 결과는 표 3, 표 4와 같다.

표 3. 중심점 찾기 실험의 결과

The number of images	400
The number of fingerprints	4000
The number of succeed fingerprints	3828
Success rate(%)	95.69
Run time(/Sec.)	0.11

표 4. 지문 후보 영역의 도표 검출 결과

The number of images	The number of succeed images	Success rate(%)	Run time
400	383	95.75	0.05/Sec.

지문의 중심 위치를 검출하는 과정에서 영상을 축소하여 처리를 하므로 원래 크기 영상을 이용할 때보다 4.8배정도 처리 속도가 빨라졌고, 선명하지 않은 글자나 잡음 등은 1차적으로 제거되는 효과가 있었다. 지문이 정확하게 칸 안에 날인되지 못하고 양손 지문이 지그재그로 표시되어 있는 경우에는 각 지문 영역의 중심을 정확하게 찾지 못하는 경향이 있는데, 이 문제

는 사용자가 마우스를 이용하여 지문 영역의 중심 위치를 옮겨 정확한 위치를 찾아줄 수 있도록 사용자에 의한 반자동 분할법도 구현하였다.

도표 검출에서 실패한 경우는 저해상도(200DPI)로 입력받은 지문원지의 수평선이 계속 연결되지 않고 끊어져서 수평선을 끝까지 추적하지 못하기 때문이거나, 지문이 수평선과 수직선이 'T'자 형태로 만나는 부분에 겹쳐서 날인되어 정확하게 수직선을 찾지 못하기 때문이다. 수평선 검출의 경우 영상의 오른쪽에서부터 왼쪽으로 수평선을 역추적하는 과정을 추가해 준다면 그 성공률을 높여 줄 수 있으며, 수직선 검출의 경우 위쪽에서 'T'자형으로 만나는 지점뿐만 아니라 아래쪽에서 '⊥'자 형으로 만나는 지점에 대한 검출도 정확한 표 추출에 이용될 수 있다.

V. 결론

본 논문에서는 지문원지로부터 지문 영상을 입력받는데 소요되는 인력과 시간을 절감하는데 도움을 주고자 블록 단위의 필터를 이용하여 지문원지에서의 지문 영상의 위치를 자동으로 검출하는 방법을 제안하였다. 도표 검출을 위한 방법으로는 직교 방정식과 'T'자형 마스크를 이용하여 지문 후보 영역의 도표 정보를 검출한다.

400개의 지문원지 파일에 대해 제안된 방법을 적용한 결과 각 지문의 중심 위치는 95.69%, 도표 검출은 95.75%의 높은 정확도를 나타내었다. 추출된 정보는 지문인식의 전처리 단계에서 적용된다.

참고문헌

- [1] A. Jain, L. Hong, R. Bolle, "On-Line Fingerprint Verification", IEEE transactions on pattern analysis and machine intelligence, Vol. 19, No. 4, April 1997
- [2] A. Jain, L. Hong, S. Pankanti, R. Bolle, "An Identity-Authentication system using fingerprints", Proceedings of the IEEE, Vol. 85, No. 9, September 1997
- [3] 이용하, 이준재, 이주형, 심재창, "영역채색을 이용한 지문특징점 추출 알고리즘", 대한전자공학회 추계종합학술대회 논문집(B) 제19권 제2호, 1996, 1452-1455
- [4] 천인국, 윤영택 "영상처리 기초편", 기한재, 1998, 75-83쪽
- [5] J. R. Parker, "Algorithms for Image Processing and Computer Vision", 1996, pp. 294-300
- [6] Rafael G. Gonzalez, Richard E. Woods, "Digital Image Processing", 1992, pp. 443-444