

# 지문 특징점 추출을 위한 방향성 연구

안효창\*, 한태규\*, 박성현\*, 김영섭\*\*, 이상범\*\*

\*단국대학교 전자컴퓨터공학과

\*\*단국대학교 전자컴퓨터공학과

e-mail:youcu92@dankook.ac.kr

## A Study on Orientation Field for Minutiae Extraction from Fingerprint Images

Hyo-Chang Ahn\*, Tae-Kyu Han\*, Sung-Hyun Park\*,  
Young-Seop Kim\*\*, Sang-Burm Rhee\*\*

\*Dept of Electrical & Computer Engineering, Dankook  
University

\*\*Dept of Electrical & Computer Engineering, Dankook  
University

### 요 약

지문 인식 알고리즘은 크게 특징점을 추출하여 특징점의 좌표, 특성, 방향각 등을 정보로 하여 인식하는 특징점 기반 알고리즘과 지문의 융선 패턴을 분석하여 인식하는 패턴 기반 알고리즘으로 나뉜다. 본 논문에서는 기존의 특징점 기반 지문인식 알고리즘과 비교하여 개선된 전처리 방법을 이용하여 보다 빠르면서도 정확한 지문 특징점 추출 알고리즘을 제안한다. 지문영상에서 방향성 정보를 추출을 위해 지문영상을 일정한 크기를 갖는 국부영역으로 나눈 후, 각 국부 영역의 융선 방향을 계산하여 방향성 정보를 얻는다. 추출된 방향성 정보를 이용하여 영상 개선 및 특징점을 추출하는 지문 인식 알고리즘을 제안하고자 한다.

### 1. 서론

오늘날 사회가 고도로 정보화되고 복잡해짐에 따라 개인 정보의 유출과 이로 인한 피해가 심각한 문제로 대두되고 있다. 지금까지 개인 인증을 위하여 사용되고 있는 암호나 열쇠를 이용한 토근 방식의 인증 기술은 생체 인식 기술로 점차 대체되고 있다. 생체 인식 기술은 도난이나 해킹의 위험이 적고 사용자 편의성에 있어 기존의 기술보다 우월하며 개인의 고유한 특성을 근거로 인증하므로 보다 높은 보안을 보장한다. 현재 생체인식은 지문, 음성, 얼굴, 망막, 홍채, 서명 인식 등이 있으며, 그 중에서 가장 널리 사용되고 있는 것은 지문 인식이다[1][2].

지문 인식 알고리즘은 특징점 기반 알고리즘과 패턴 기반 알고리즘으로 크게 나뉜다. 특징점 기반 알고리즘은 특징점의 좌표, 특성, 방향각 등을 정보로 하여 지문 인식을 하고, 패턴 기반 알고리즘은 지문의 융선 패턴을 분석하여 지문을 인식을 하며, 주로

특징점 추출이 쉽지 않은 저화질 영상에서 사용한다 [1][3][5].

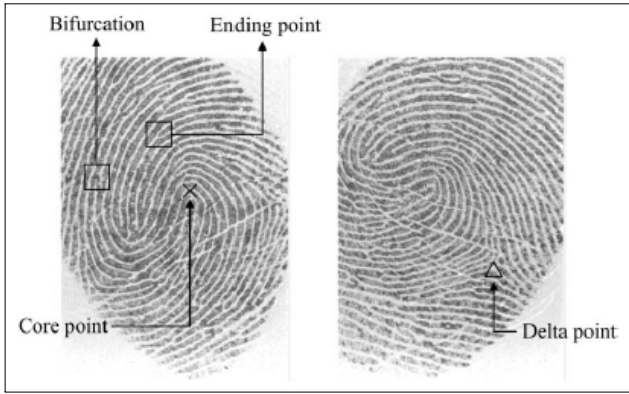
지문을 국부적으로 해석하면 일정한 방향을 갖는 융선으로 이루어진 패턴으로 볼 수 있다. 일반적으로, 특징점 기반 지문인식 알고리즘에서는 지문의 융선에 의해 구성되는 특징점(Minutiae) 즉, 분기점(Ridge bituration)과 끝점(Ridge Ending)을 추출한다. 끝점은 융선이 끊어지는 곳이며 분기점은 융선이 두 개로 나뉘어 지는 곳이다. 이밖에 지문의 분류를 위해 중심선과 삼각주를 추출하기도 한다 [1][2][3].

본 논문에서는 기존 연구결과를 바탕으로 지문 영상 개선을 위한 전처리 기법과 기존의 특징점 기반의 지문 인식의 단점을 지문의 특성에 따른 적응적 방향성 정보추출을 통하여 보완함으로써 효율인 지문 인식 알고리즘을 제안한다.

## 2. 관련 연구

### 2.1 지문의 구조

일반적으로 한 손가락에는 특징점이 30~50개 정도가 분포하며, 사람마다 모두 특징점의 종류와 위치, 방향이 다르게 나타난다. [그림 1]은 지문의 대표적인 특징점을 포함하고 있는 지문의 구조를 나타내고 있다. 각 특징점들의 특징은 다음과 같다[1][2].

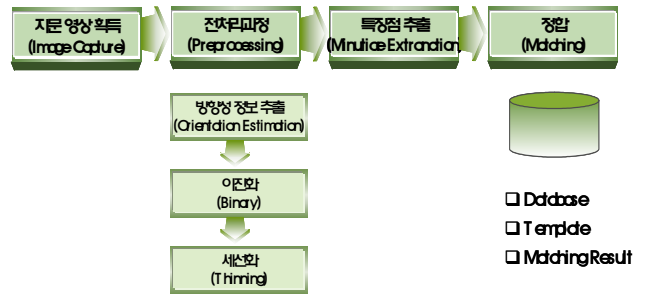


[그림 1] 지문의 구조

- 1) 융선(Ridge): 피부가 융기한 선으로 땀구멍이 융기하여 만들어진 것으로 피부계층 중 진피부분이 손상되지 않는 한 평생 동안 변하지 않는다.
- 2) 중심점(Core Point): 융선의 굴곡이 위쪽으로 향한 상부중심점(Upper Core)과 아래쪽으로 향한 하부중심점(Lower Core)로 분류된다.
- 3) 삼각점(Delta Point): 지문의 융선 흐름이 세 방향에서 모이는 곳으로 지문의 특징점 중의 하나로 이용된다.
- 4) 단점(Ending Point): 융선이 부드럽게 흐르다가 끊어지는 점으로 지문인식에 사용된다.
- 5) 분기점(Bifurcation Point): 융선이 부드럽게 흐르다가 갈라지는 점으로 지문인식에 사용된다.

### 2.2 지문인식 시스템의 구성

지문인식 시스템의 구성은 다음 [그림 2]과 같다. 지문영상 획득 후 여러 단계의 전처리 과정을 거친 다음, 방향성 정보를 추출하고, 이 방향성 정보를 바탕으로 특징점을 추출하게 된다. 이러한 과정을 거쳐 기존의 지문 데이터베이스에 저장된 지문 형상과 비교 후 인증 유무를 판단하게 된다[4].



[그림 2] 지문인식 시스템 구성도

### 2.3 기존의 방향성 추출 알고리즘

지문의 융선 방향성 정보를 추출하는 방법은 Gray Level 영상으로부터 직접 분류하는 방법과 이진화 영상을 만들고 이진 영상으로부터 추출하는 방법이 있다[3][7].

미소형태 히스토그램에 의한 방향 결정법은 Gray Level 영상으로부터 이진화 영상을 만들고, 이 영상으로부터 미소형태(MicroPattern)를 사용하여 융선의 방향성 정보를 추출한다. 이 방법은 처리 방법이 간단하지만 영상을 이진화하는 과정이 먼저 수행되어야 하고, 이진화 결과에 따라 큰 차이를 보이는 단점을 가지고 있다[7][8].

Run Length를 이용한 방향 결정법도 이진화된 영상에서 융선의 기울기는 화소값이 존재하는 부분에서 융선의 X축 방향의 길이와 Y축 방향의 길이의 비로써 구할 수 있지만, 영역 융선 방향이 양의 방향인지, 대표성분이 높고 그른지를 쉽게 구별해 내지 못하는 단점을 가지고 있다[7].

또한 본 논문에서 이용하는 Sobel 연산을 이용한 방향 결정법은 Gray Level 영상으로부터 융선과 골의 경계가 되는 모서리 부분에서 현격한 차이를 보이는 Gradient값을 이용하여 구할 수 있다. [그림 3]의 마스크를 이용하여  $G_x(i, j)$ ,  $G_y(i, j)$ 의 값을 구하며, 이를 이용하여 방향값을 정한 후 한 블록내 점들의 방향을 평균하여 블록의 방향으로 정한다[5][7].

-1	-2	-1	-1	0	1
0	0	0	-2	0	2
1	2	1	-1	0	1
G <sub>x</sub> 를 구하기 위한 마스크			G <sub>y</sub> 를 구하기 위한 마스크		

[그림 3] Sobel 마스크

이 외에도 Least Difference of Lightness 법, 최적화 함수를 이용한 방향 결정법, Window frame 마스크를 이용한 방향 결정법 및 Slit Sum 마스크를 이용한 방향 결정법 등이 있다[7].

### 3. 제안한 알고리즘

지문 영상에서 특징점을 추출하기 위하여, 본 논문에서는 전처리 과정을 통해 얻은 개선된 지문영상에서 방향성 정보를 얻고, 이를 기반으로 하여 지문 인식의 효율을 높이고자 한다.

#### 3.1 방향성 추출 알고리즘

지문 영상에서의 특이점뿐만 아니라 방향성 정보는 지문 인식 시스템에서 인식률을 높일 수 있는 지문 영상 고유의 특징요소가 된다. 본 논문에서는 방향성 정보를 추출하기 위해 Sobel 연산자를 이용하여  $G_x(i, j)$ ,  $G_y(i, j)$ 의 값을 구하며, 이 Gradient를 이용하여  $\theta(i, j)$ 를 구할 수 있다[5].

추정된 방향 필드는 상처나 잡음에 의한 영향에 의하여 계산된 방향  $\theta(i, j)$ 이 맞지 않을 수 있다. 중심점이나 삼각점과 같은 특이점이 없는 곳에서는 용선의 방향이 서서히 변하므로 지역 필터링을 통하여 잘못 추정된 방향 필드를 보정할 수 있다. 지역 필터링을 통해서 용선의 방향  $\alpha(i, j)$ 를 구할 수 있으며, 스무딩되지 않은 방향 필드  $\theta(i, j)$ 보다 주변 블록에 대한 변화폭이 적다.

Step 1) 전체 지문 영상을 8×8화소의 세부 블록으로 나누는 후, 영상의 가 화소에서 x, y방향으로 3×3의 Sobel 연산자를 이용하여  $G_x(i, j)$ ,  $G_y(i, j)$ 의 값을 각각 구한다.

Step 2) 다음 식(1)을 이용하여 각 화소에서 방향성 정보  $\theta(i, j)$ 를 구한다.

$$\theta(i, j) = \frac{1}{2} \tan^{-1} \frac{2G_x(i, j) - G_y(i, j)}{G_x^2(i, j) - G_y^2(i, j)} \quad (1)$$

Step 3) 지역 통과 필터링을 수행하기 위해 추정된 방향성 정보에 다음 식(2)을 이용하여 연속된 벡터 필드로 변화해 준다.

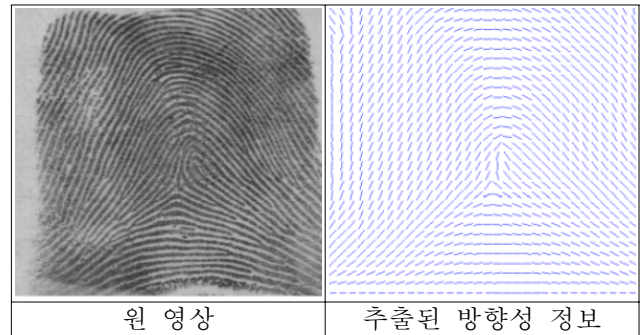
$$\begin{aligned} \Phi_x(i, j) &= \cos(2\theta(i, j)) \\ \Phi_y(i, j) &= \sin(2\theta(i, j)) \end{aligned} \quad (2)$$

Step4) x와 y방향의 벡터 필드 성분이 구한 후, 다음 식(3)을 이용하여 지역 필터링을 한다.

$$\begin{aligned} \Phi'_x(i, j) &= \sum_{u=-W_{\phi}/2}^{W_{\phi}/2} \sum_{v=-W_{\phi}/2}^{W_{\phi}/2} W(u, v) \Phi_x(i-uw, j-vw) \\ \Phi'_y(i, j) &= \sum_{u=-W_{\phi}/2}^{W_{\phi}/2} \sum_{v=-W_{\phi}/2}^{W_{\phi}/2} W(u, v) \Phi_y(i-uw, j-vw) \end{aligned} \quad (3)$$

Step 5) 다음 식(4)를 이용하여  $(i, j)$ 에서의 용선 방향  $\alpha(i, j)$ 를 구한다.

$$\alpha(i, j) = \frac{1}{2} \tan^{-1} \left( \frac{\Phi'_x(i, j)}{\Phi'_y(i, j)} \right) \quad (4)$$



[그림 4] 지문 영상의 방향성 정보 추출

#### 3.2 특징점 추출

지문 영상에 세선화과정을 거친 후, 지문 영상에서 보다 나은 특징점을 추출할 수 있다[6]. 3×3 마스크를 이동하면서 중심 픽셀  $C_0$  주변 픽셀들의 밝기 변화 횟수를 조사하면 그림 과 같이 고립점(Isolated Point), 단점(Ending Point), 분기점(Bifurcation)으로 구분된다.

$C_1$	$C_2$	$C_3$
$C_8$	$C_0$	$C_4$
$C_7$	$C_6$	$C_5$

[그림 5] 3×3 특징점 추출 마스크

$C_1$	$C_2$	$C_3$	$C_1$	$C_2$	$C_3$	$C_1$	$C_2$	$C_3$
$C_8$	$C_0$	$C_4$	$C_8$	$C_0$	$C_4$	$C_8$	$C_0$	$C_4$
$C_7$	$C_6$	$C_5$	$C_7$	$C_6$	$C_5$	$C_7$	$C_6$	$C_5$
고립점			단점			분기점		

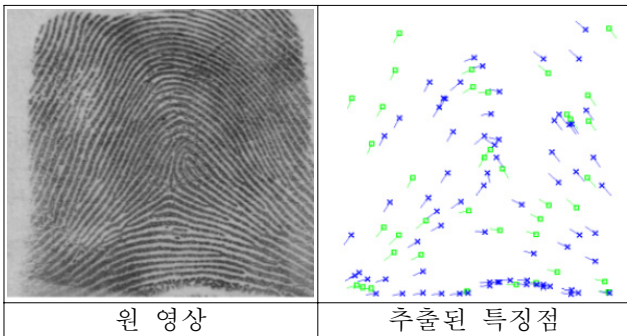
[그림 6] 각 특징점 추출을 위한 3×3 특징점 추출 마스크

지문 인식에 필요한 특징점 추출은 다음 식(5)의  $CN$ (Crossing count Number)에 의하여 구한다.

$$CN = \sum_{i=1}^8 |C_{i+1} - C_i| \quad (5)$$

선을 이루는 화소값을 '0'인 부분을 '1'로 하고, 배경을 이루는 화소값을 '255'를 '0'으로 하여 화소의 값이 '1'인 중심 픽셀  $C_0$ 를 기준으로 해서 마스크를 할 경우,  $CN$ 값이 고립점은 '0', 단점은 '2', 분기점은 '6'인 것을 알 수 있다.

특징점의 방향각은 용선의 진행방향을 나타내주는 값으로, 검출된 특징점 좌표에서 세선화된 지문 용선을 일정한 거리만큼 따라간 후 찾을 수 있다. 즉, 용선을 따라갈 때 끝점은 가지가 한 개가 되고 분기점은 세 개가 되는데, 방향각은 특징점 좌표와 가지 끝점의 위치관계로 결정된다.



[그림 7] 지문 영상의 특징점 추출

#### 4. 실험 결과

본 논문에서는 전처리 과정을 통해 향상된 지문 영상에 세분화된 Block에서의 Sobel 연산을 통해 Gradient를 구하고, 이를 저역 필터링을 수행하여 방향성 정보를 추출하므로, 기존에 제안된 지문인식 알고리즘보다 처리 속도가 평균 0.3초정도 빠르고 안정적이었으며, 또한 본인 인식률에서 0.2% 향상된 결과를 보여주었다.

제안한 알고리즘의 성능 평가를 위해 Pentium 4

1.8GHz의 PC에 Windows XP 운영체제를 사용하였고, 알고리즘 구현은 Visual C++ 언어를 사용하였다.

#### 5. 결론 및 향후 연구과제

본 논문에서는 지문 영상 향상을 위한 전처리 기법과 기존의 특징점 기반의 지문 인식의 단점을 방향성 추출을 통하여 보완함으로써 인식 성능을 높일 수 있는 지문 영상에서의 특징 요소 추출 방법을 나타내었다. 결과는 기존의 알고리즘보다 다소 향상되었지만, 지문 영상의 보다 정확한 획득 방법에 관한 연구와 손상된 지문 영상에 대한 복원 문제에 대하여 보다 많은 연구가 이루어져야 할 것으로 본다.

#### 참고문헌

- [1] Anil K. Jain, Lin Hong, Sharath Pankanti, Ruud Bolle, "An Identity-Authentication System Using Fingerprint", Proceeding of the IEEE, Vol. 85, No. 9, pp. 1365~1388, 1997
- [2] Anil K. Jain, Lin Hong, "Intergrating Faces and Fingerprints for Personal Identification", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 20, No. 12, pp. 1295~1307, 1998
- [3] K. Karu and A. K. Jain, "Fingerprint classification", Pattern Recognition, Vol. 29, No. 3, pp. 389~404, 1996
- [4] Anil Jain, Lin Hong, Ruud Bolle, "On-Line Fingerprint Verification", IEEE Transactions on Pattern Analysis Machine Intelligence, Vol. 19, No. 4, pp. 302~313, 1997
- [5] Pedro R. Vizcaya, Lester A. Gerhardt, "A Nonlinear Orientation Model for Global Description of Fingerprints", Pattern Recognition, Vol.29, No.7, pp. 1221~1231, 1996
- [6] Anil Jain, Salil Prabhakar, Lin Hong, "A Multichannel Approach to Fingerprint Classification", IEEE Transactions on Pattern Analysis Matching Intelligence, Vol. 21, No. 4, pp. 348~359, 1999
- [7] Anil Jain, Lin Hong, Yiefei Wan, "Fingerprint Image Enhancement: algorithm and Performance Evaluation", IEEE Transactions on Pattern Analysis Matching Intelligence, Vol. 20, No. 8, pp. 777~789, 1998