

Core point와 Flow-line 추적을 이용한 지문 영상의 분류

준회원 박철현*, 오상근*, 이경환*, 정회원 김현순*, 박길흠*

Fingerprint Classification Using Core Points and Flow-line Tracing

Chul-hyun Park*, Sang-keun Oh*, Kyung-hwan Lee* *Associated Members*

Hyun-soon Kim*, Kil-houm Park* *Regular Members*

요 약

지문영상의 분류는 데이터베이스의 용량이 클 경우 검색시간을 효율적으로 단축시킬 수 있는 핵심적인 기술이다. 따라서 본 논문에서는 core point와 flow-line 추적을 이용한 효율적인 지문 영상 분류 기법을 제안한다. 제안한 방법은 특히 압착 날인된 지문 영상의 분류에 적합한 방법으로 크게 2단계로 이루어져 있다. 첫 번째 단계에서는 먼저 Poincaré index를 이용하여 core point를 찾아내고 이를 바탕으로 개략적인 분류를 수행한다. 그 다음 두 번째 단계에서는 core point를 중심으로 flow-line을 추적하여 그 결과를 가지고 세부적인 분류를 수행한다. 세부 분류 단계에서는 평활화된 블록의 방향정보를 이용한 효과적인 flow-line 추적 알고리즘과 이를 이용한 새로운 분류 방법이 제안된다. 제안한 방법은 회전이나 이동 그리고 약간의 잡음에 강인한 지문 분류 방법으로 지문입력기를 통하여 획득된 700장의 지문 영상에 적용해 본 결과 93.6%의 분류율을 나타내었다.

ABSTRACT

Fingerprint classification is an essential technique to reduce searching time, in case of a large database. In this paper, an efficient fingerprint classification technique using core point(s) and flow-line tracing is proposed. The proposed method, which is adequate for a dab fingerprint, consists of two steps. First, core points are detected by using Poincaré index and fingerprint images are coarsely classified based on the number of detected core point(s). Next, Fine classification is performed by tracing flow-line from the detected core point(s). In the second step, an effective flow-line tracing algorithm using directional information of blocks is presented and a new classification method based on the traced flow-lines is proposed. Experimental results show that the proposed method is relatively robust to rotation, shift and moderate noise compared to other approaches and has a classification accuracy of 93.6%.

I. 서론

지문은 그 고유한 특성 때문에 오래 전부터 범죄 조사, 신원 확인 그리고 접근 제어 등 다양한 분야에서 응용되어 왔다. 최근 고성능 지문입력기의 개발로 인해 양질의 지문 영상을 얻을 수 있게 되어 지문을 이용한 여러 가지 응용분야의 연구에 활기

를 띄고 있다. 일반적으로 지문은 그림 1에서와 같이 arch(A), tented arch(TA), left loop(LL), right loop(RL), twin loop(TL)와 whorl(W)의 6개 주 분류로 나누어질 수 있다.^[1,2] 전체적인 지문 인식 시스템에서 세부적인 정합과정이 이루어지기 전에 지문 영상의 분류가 이루어진다면 검색할 대상을 제한시킴으로써 검색 시간을 획기적으로 줄일 수가 있다.

* 경북대학교 전자공학과 (nagne@palgong.knu.ac.kr)
논문번호 : 00359-0915, 접수일자 : 2000년 9월 15일

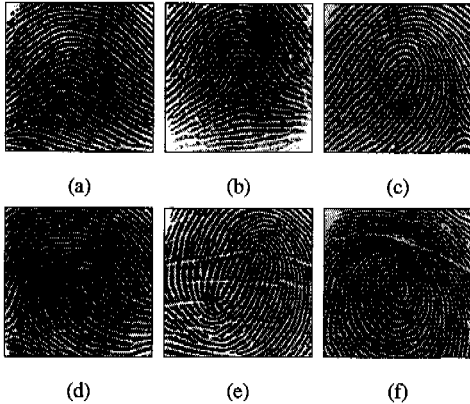


그림 1. 지문의 부류 (a) Arch (b) Tented arch (c) Left loop (d) Right loop (e) Twin loop (f) Whorl

기존의 지문 영상의 분류 기법은 크게 구문론적 (syntactic)인 방법, 구조적(structure-based)인 방법, 특이점(singularities)에 기반한 방법 그리고 주파수 영역(frequency-based)에서의 접근방법 등으로 나눌 수가 있다. 구문론적인 방법은 형식적인 문법(formal grammar)을 이용하여 지문을 분류하는 방법으로 입력 지문 영상이 부류 특유 문법에 의해 생성된 문자열로 묘사되었을 경우 어느 부류에 속하는지 결정하게된다.^[3] 구조적인 방법은 일반적으로 추정된 방향 패턴을 이용하여 지문을 분류하는 방법으로 신경망을 이용한 방법^[4]과 은닉 Markov 모델(hidden Markov model)을 이용한 방법^[5] 등이 있다. 특이점에 기반한 방법은 그림 2에서와 같이 지문 영상에서 core point와 delta point를 찾아내고 이 특이점들의 개수와 위치를 이용해 지문을 분류하는 방법^[1]이고 주파수 기반 방법은 지문 분류를 위해 영상의 주파수 스펙트럼을 이용하는 방법이다.^[6]

특이점에 기반한 방법의 경우 방법은 간단하나 지문 영상 내에 특이점이 존재하지 않거나 존재한 다하더라도 영상의 잡음 성분으로 인해 검출되지 않을 경우 분류를 할 수 없다는 단점이 있다. 예를 들면 압착 날인한 지문 영상의 경우 일반적으로 영상의 가장 자리 부근에 위치한 delta point가 거의 나타나지 않고 나타난다 하더라도 영상의 가장 자리에 존재하기 때문에 날인할 때의 압력이 작아서 그 부근의 영상의 화질이 떨어져 검출 시 거의 나타나지 않는 경우가 대부분이다. 최근에 소개된 방법들 중에 Gabor 필터를 이용하여 FingerCode를 생성시킨 후에 이를 이용하여 지문을 분류하는 방법^[2]이 있으나 이 방법은 Gabor 필터링을 수행할

때 시간이 많이 소요되는 단점이 있다. 그리고 방향 패턴이미지의 분할을 이용한 분류방법^[7]은 영상의 방향패턴 이미지를 분할한 뒤에 각각의 분할된 영상을 그래프 형태로 표현하여 이를 모델그래프와 정합 시킴으로써 지문을 분류하는 방법이다. 이 방법은 방향 패턴의 분할이 용이하지 않다는 점과 각각의 분할된 영상을 그래프로 표현하는 과정이 매우 복잡하다는 단점을 가지고 있다.



그림 2. 특이점들(O: Core point, Δ: Delta point)

따라서 본 논문에서는 일반적으로 압착날인의 경우 delta point는 유실되는 경우가 많으나 core point는 대부분 존재한다는 점과 flow-line이 영상의 부류를 결정하는데 중요한 전체적인 영상 패턴(global pattern)을 잘 표현한다는 점에 착안하여 core point와 flow-line 추적을 이용한 지문 영상 분류 방법을 제안한다. 제안된 방법은 원리적으로 Kawagoe와 Tojo의 방법^[8]과 동일하나 단계적인 구현에 있어서의 차별적인 방법을 사용한다. 제안하는 방법은 크게 두 단계로 이루어지는데 첫 번째 단계는 Poincaré index를 이용하여 core point를 검출하는 단계이다. 이렇게 구해진 core point의 개수를 바탕으로 영상을 개략적으로 분류한다. 그리고 core point로부터 방향정보를 이용하여 flow-line을 추적한다. 최종적으로 구해진 flow-line 정보를 이용하여 세부적인 분류를 수행한다. Kawagoe의 논문에서는 직접적인 방법에 대한 언급이 없으나 flow-line의 정확한 추적이 성능에 큰 영향을 미치므로 본 논문에서는 영상의 방향 패턴을 이용한 flow-line 추적 알고리즘을 소개하는 데 주력하였다. 그리고 추적된 flow-line 정보를 이용하는 새로운 척도(measure)를 이용하여 영상을 분류하는 방법을 제안한다. 제안된 방법의 과정을 그림 3에서 흐름도로 나타내었다.

II. Core Point와 Flow-line 추적을 이용한 지문 영상의 분류

제안한 방법은 2단계 분류 방법으로 첫 번째 단

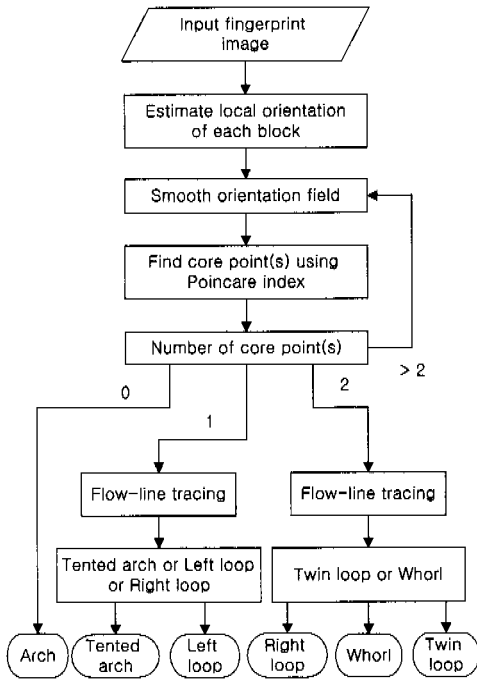


그림 3. 전체 지문 영상 분류 과정의 흐름도

계는 Poincaré index를 이용한 개략적인 분류 단계이고 두 번째 단계는 flow-line 추적을 이용한 세부 분류 단계이다.

2.1 Core point를 이용한 개략적인 분류

영상의 core point를 찾는 방법은 여러 가지가 있으나 제안한 방법에서는 Poincaré index를 이용하여 core point를 검출한다. 검출하는 과정은 다음과 같다.

- 1) $M \times N$ 영상을 $w \times w$ 크기의 부 블록으로 나눈 뒤에 각각의 블록 (x, y) 에 대해서 다음의 식(1)로부터 대표 방향 $\theta(x, y)$ 을 구한다.^[9]

$$\theta_0 = \frac{1}{2} \tan^{-1} \left[\frac{\sum_{i=1}^w \sum_{j=1}^w 2G_x(i, j)G_y(i, j)}{\sum_{i=1}^w \sum_{j=1}^w (G_x^2(i, j) - G_y^2(i, j))} \right] \quad (1)$$

여기서 w 는 블록의 크기이고 G_x, G_y 는 각각 x, y 방향 그래디언트(gradient)의 크기이다. 실험에서는 각각의 그래디언트 값을 구하기 위해 3×3 Prewitt 연산자를 사용하였다.

- 2) 구해진 방향패턴을 벡터 형태 즉, $(\cos 2\theta_0, \sin 2\theta_0)$ 로 변환한 뒤 각각의 성분 별로 3×3 average 필터를 이용하여 블록 단위로 평

활화(smoothing)한다. 그리고 평활화된 벡터 값을 다시 방향 값으로 변환시킨 후에 8개의 방향으로 양자화한다. 이렇게 블록 (x, y) 에서의 평활화되고 양자화된 방향패턴을 $\theta(x, y)$ 로 표시한다.

- 3) 각각의 블록 (x, y) 에 대하여 Poincaré index를 구한다. Poincaré index는 식(2)와 같이 현재 블록을 기준으로 2×2 블록에 대해 반 시계 방향으로 즉, $(x, y) \rightarrow (x+1, y) \rightarrow (x+1, y+1) \rightarrow (x, y+1) \rightarrow (x, y)$ 순으로 방향의 변화를 모두 더함으로써 구해진다.^[11] Poincaré index가 $1/2$ 면 그 위치가 core point의 위치가 된다.

$$\text{Poincaré}(x, y) = \frac{1}{2\pi} \sum_{k=0}^3 \Delta(k) \quad (2)$$

$$\Delta(k) = \begin{cases} \delta(k), & \text{if } |\delta(k)| < \pi/2 \\ \pi + \delta(k), & \text{if } \delta(k) \leq -\pi/2 \\ \pi - \delta(k), & \text{otherwise,} \end{cases}$$

여기서 $\delta(k)$ 는 두 블록의 방향의 차이인데 예를 들어, $\delta(0)$ 는 $\theta(x+1, y) - \theta(x, y)$ 과 같이 구해진다. 나머지 값들도 다음 블록과 현재 블록의 방향 차이로 구하되 크기에 따라 적절하게 $\Delta(k)$ 로 변경된다.

- 4) 만약 두 개보다 많은 수의 core point가 검출되면 단계 2)로 돌아간다.

그림 4에서 Poincaré index를 이용하여 core point를 검출하는 과정 별 영상을 나타내었다. 일단 Poincaré index를 이용하여 core point가 검출이 되면 검출된 core point의 개수를 이용한 개략적인 분류가 이루어진다. Arch 형태의 지문인 경우 core point가 검출되지 않으며 tented arch, left loop 그리고 right loop의 경우 한 개의 core point가 검출된다. 나머지 whorl과 twin loop의 경우 2개의 core point가 검출된다. 따라서 core point의 개수에 의한 개략적인 분류가 식(3)과 같이 이루어진다.

$$\begin{matrix} A, & \text{if } n=0 \\ TA \text{ or } LL \text{ or } RL, & \text{if } n=1 \\ TL \text{ or } W, & \text{if } n=2 \end{matrix} \quad (3)$$

여기서 n 은 core point의 개수이다.

2.2 Flow-line 추적을 이용한 세부적 분류

2.2.1 Flow-line 추적 기법

제안한 방법에서는 core point에서의 flow-line 추

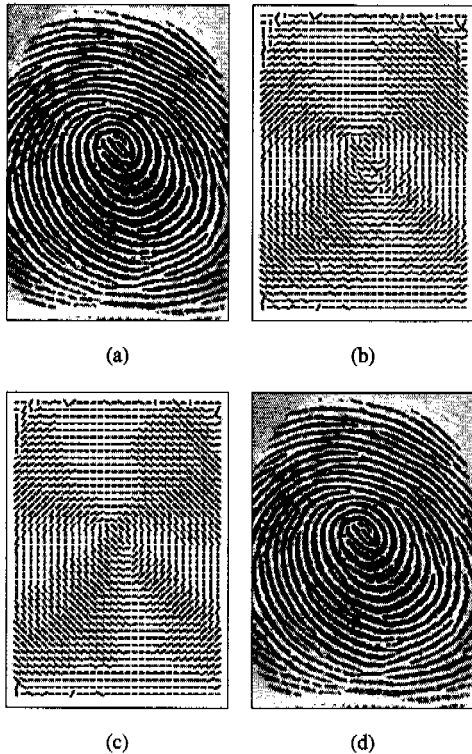
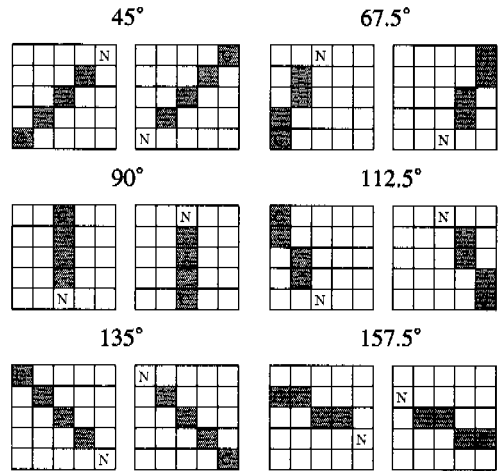
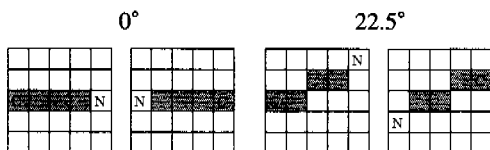


그림 4. core point 검출 과정 (a) 원영상 (b) 방향패턴 (c) 평활화된 방향패턴 (d) 검출된 core point

적을 위해 각 위치에서의 평활화된 방향패턴을 이용한다. 이 방향 정보는 core point 검출 시 이미 저장되어 있는 정보를 이용한다. 현재 픽셀 위치에서 다음 픽셀 위치의 결정은 그림 5에서의 같이 이루어진다. Flow-line의 방향은 각 방향별로 2개의 방향이 존재하는데 flow-line의 종류와 진행방향을 고려하여 두 방향 중의 하나를 선택하게 된다. 일 회에 이동되는 거리는 블록의 크기(8×8)보다 작으면서 각각의 방향을 정밀하게 표현할 수 있는 크기로 결정하였다. 22.5°, 67.5°, 112.5° 와 157.5° 경우에는 정확한 방향의 표현이 불가능하므로 약간의 오차가 존재한다. 그러나 방향패턴 자체가 블록의 대표 값으로 평활화되고 양자화된 값이기 때문에 flow-line을 정확하게 추적하는데는 큰 어려움이 없다.



C : location of current pixel, N : location of next pixel

그림 5. Flow-line 추적 시 다음 픽셀의 위치 결정

일단 core point의 개수가 하나인 경우 core point를 기준으로 왼쪽(L, left)과 오른쪽(R, right)으로 구분하여 flow-line 추적을 수행한다. 그리고 core point의 개수가 두 개인 경우는 첫 번째 core point의 왼쪽(UL, upper left)과 오른쪽(UR, upper right) 그리고 두 번째 core point의 왼쪽(LL, lower left)과 오른쪽(LR, lower right)로 구분하여 flow-line 추적을 수행한다. 원칙적으로 L과 UL이 그리고 R과 UR이 동일하다. UL, UR, LL, LR 정보는 core point에서의 초기 진행 방향을 결정하는 데 사용된다. 그리고 이후의 flow-line의 진행방향 결정은 현재 위치에서의 이전 픽셀의 위치를 기억하여 상대적인 x, y 좌표의 이동여부를 통하여 결정하게 된다. 그런데 core point 부근에서는 방향패턴의 급격한 변화로 인해 위에서 언급한 진행방향 정보만으로는 정확한 flow-line 추적이 어렵다. 따라서 앞의 언급한 UL, UR, LL 그리고 LR의 정보를 이용하게 되는데 지문 영상의 경우 flow-line의 각도 변화 방향이 존재한다. 일반적으로 UL과 LR의 경우는 각도가 점점 증가하는 방향이고 flow-line의 방향이 반시계 방향으로 감아 들어가는 형태이다. UR, LL의 경우는 반대로 각도가 점점 감소하는 방향이고 flow-line의 방향이 시계 방향으로 감아 들어가는 형태이다. 이러한 각 영역에서의 조건을 만족시키도록 추적 방향을 결정하면 core point 부근에서의 에러를 해결할 수 있다. 정리하면 픽셀의 진행방향이 우선 조건이 되나 각도의 변화가 심할 경우 flow-line의 종류를 고려한 방향으로 추적이 진행된다.

Flow-line 추적 시 방향 결정이 어떻게 이루어지는가에 대한 예가 그림 6에 나타나 있다. 현재 픽셀에서의 방향이 0° 인 경우처럼 현재 픽셀에서의 방향이 이전 픽셀에서의 방향에 대해 좌우 45° 범위 이내에 들어오는 경우에는 flow-line이 그 범위 안쪽의 방향으로 이동하게끔 방향을 결정하면 된다. 따라서 a' 방향이 아닌 a 방향으로 flow-line이 진행되어 나간다. 그리고 현재 픽셀에서의 방향이 135° 인 경우처럼 현재 픽셀에서의 방향이 이전 픽셀에서의 방향에 대해 좌우 45° 범위를 벗어나는 경우에는 어느 core point의 어느 쪽 flow-line인지 알고 있으면 정확한 방향을 결정할 수 있다. 그림의 flow-line은 첫 번째 core point의 오른쪽(UR) flow-line이므로 flow-line의 방향이 시계방향으로 감아 들어가는 형태가 된다. 따라서 flow-line은 b' 방향이 아닌 b 방향으로 진행되어 나간다.

실험에서는 두 번째 core point에서의 방향 패턴이 지문 분류를 위한 flow-line을 찾기 위한 시작점으로 적절치 않아 위치를 core point간의 직선의 각도를 고려하여 바깥쪽으로 한 블록 크기정도 이동시킨 지점을 새로운 core point로 결정하였다. Flow-line 끝점의 결정은 core point의 개수가 한 개인 경우에는 영상의 오브젝트 가장 자리까지 추적이 이루어지고 core point의 개수가 두 개인 경우에는 두 개의 core point를 잇는 직선의 이동분선을 만날 때까지 추적이 이루어진다.

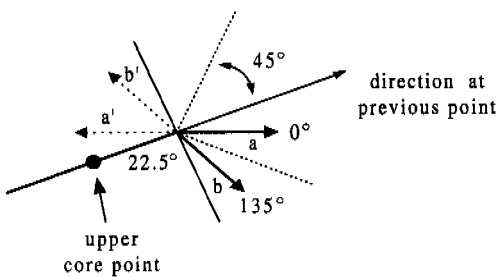


그림 6. Flow-line의 방향 결정

2.2.2 Flow-line을 이용한 tented arch, left loop, right loop의 구분

Tented arch, left loop, right loop는 core point의 개수가 하나이므로 flow-line도 왼쪽, 오른쪽 각각 하나씩 추적된다. 각각의 flow-line의 끝점을 (Lr[0], Lc[0]), (Rr[0], Rc[0])로 표시하고 core point의 좌표를 (Cr[0], Cc[0])로 표시할 경우 분류는 다음과 같이 이루어진다.

$$\begin{aligned} TA, & \text{ if } Lc[0] < Cc[0] \text{ and } Rc[0] > Cc[0] \\ LL, & \text{ if } Lc[0] < Cc[0] \text{ and } Rc[0] < Cc[0] \\ RL, & \text{ if } Lc[0] > Cc[0] \text{ and } Rc[0] > Cc[0] \end{aligned} \quad (4)$$

끝점이 core point를 기준으로 모두 왼쪽에 존재하면 left loop, 반대로 오른쪽에 존재하면 right loop가 된다. 그리고 왼쪽 끝점이 core point의 왼쪽에 오른쪽 끝점은 core point의 오른쪽에 존재하게 되면 tented arch가 된다. 그런데 회전이 심하게 될 경우 loop 형태임에도 불구하고 왼쪽 끝점이 core point의 왼쪽에 오른쪽 끝점은 core point의 오른쪽에 존재하게 되는 경우가 존재하게 된다. 이 경우에는 다음의 조건을 이용하여 분류한다.

$$\begin{aligned} TA, & \text{ if } \min(Cc[0] - Lc[0], Rc[0] - Cc[0]) > T \\ LL, & \text{ else if } (Cc[0] - Lc[0]) > (Rc[0] - Cc[0]) \\ RL, & \text{ otherwise} \end{aligned} \quad (5)$$

각 끝점에서 core point까지의 거리가 특정 문턱치(T) 이상이면 tented arch이고 거리가 문턱치에 미치지 못할 경우 어느 방향으로 더 치우쳤는가를 파악하여 left loop인지 right loop인지 결정하게 된다. 그림 7은 각각의 경우에 대해서 flow-line추적을 수행한 결과이다.

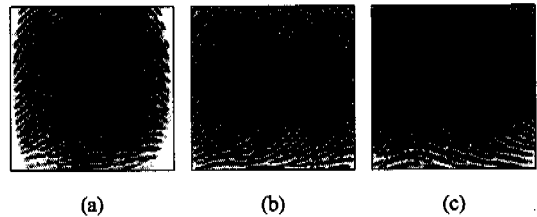


그림 7. flow-line 추적 결과 (a) Tented arch (b) Left loop (c) Right loop

2.2.3 Flow-line을 이용한 twin loop와 whorl의 구분

Twin loop와 whorl의 구분은 추적된 flow-line의 끝점과 두 개의 core point를 잇는 직선과의 거리로 이루어진 여러 개의 매개 변수(A1, A2, B1, B2)를 이용해서 이루어진다.

그림 8에서 A1, A2는 각각 core point를 잇는 직선의 중심점에서부터 첫 번째 core point의 왼쪽 오른쪽 flow-line의 끝점까지의 거리이고 B1, B2는 각각 core point를 잇는 직선의 중심점에서부터 두 번째 core point의 왼쪽과 오른쪽 flow-line의 끝점까지의 거리이다. 만약 끝점이 두 개의 core point를 잇는 직선 반대편으로 넘어가게 되면 파라미터

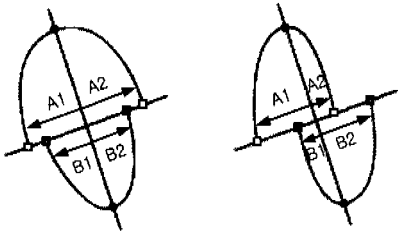


그림 8. Twin loop와 whorl의 구분을 위한 매개 변수들

값들은 음수가 된다. 각각의 매개변수 값들이 모두 결정이 되면 분류는 다음과 같은 조건에 의해 수행된다. 조건 (1)-(3)까지의 조건 중 하나라도 만족이 되면 whorl이고 그렇지 않은 경우에는 twin loop가 된다.

(조건1)

$$A1 > 0 \text{ and } A2 > 0 \text{ and } B1 > 0 \text{ and } B2 > 0$$

$$\text{and } \min(A1 + B2, A2 + B1) > T_1(l)$$

(조건2)

$$(A1 > 0 \text{ and } B1 > 0 \text{ and } |A1 - B1| < T_2(l))$$

$$\text{or } (A2 > 0 \text{ and } B2 > 0 \text{ and } |A2 - B2| < T_2(l))$$

(조건3)

$$(A1 > B1 \text{ and } A2 > B2)$$

$$\text{or } (A1 < B1 \text{ and } A2 < B2)$$

조건 1에서 A1, A2, B1 그리고 B2의 모든 값이 양수이고 (A1+B2)과 (A2+B1) 중에 작은 값이 문턱치 $T_1(l)$ 보다 크면 whorl이 된다. 반면에 지정한 값이 그림 9의 (a)에서와 같이 문턱치 $T_1(l)$ 보다 작은 경우에는 twin loop가 된다. 조건 1을 통해서 대부분 분류가 되나 몇몇 경우에 대해서 분류가 되지 않는 경우가 있어 조건 2와 3을 추가하였다. 조건 2는 그림 9의 (c), (d)와 같이 한쪽 편의 파라미터 값들이 모두 양수이면서 그 크기가 흡사하면 whorl 된다는 조건이다. 그리고 마지막 세 번째 조건은 그림 9의 (e), (f)와 같이 파라미터 값들이 아래쪽이나 위쪽 중에 한쪽의 끝점의 위치가 다른 쪽의 끝점보다 큰 경우 whorl이 된다는 조건이다. 그 외의 경우는 twin loop로 구분하게 된다. 여기서 l 은 core point 간의 거리이고 조건 1과 조건 2의 문턱치(T_1, T_2)는 거리(l)의 함수로 주어진다. 문턱치는 거리에 비례하는 값으로 실험적으로 결정하였다. 그림 10은 whorl과 twin loop의 경우 flow-line 추적의 결과를 수행한 영상이다.

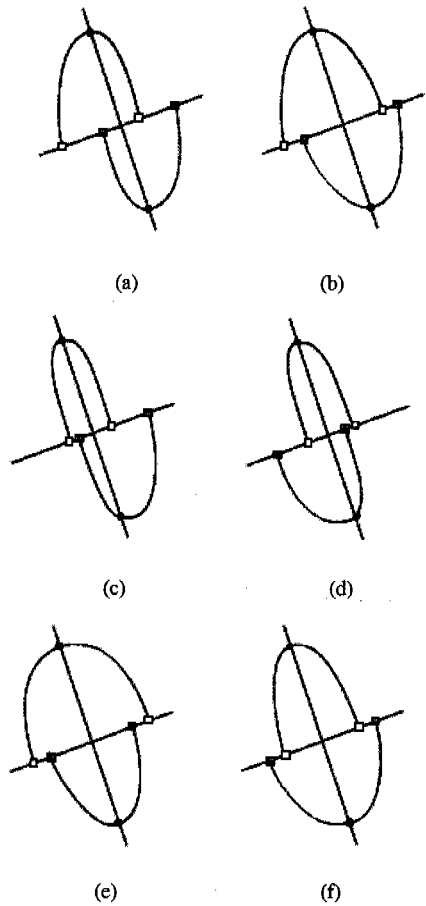


그림 9. Twin loop나 whorl이 되기 위한 조건들 (a) Twin loop (b)-(f) Whorl

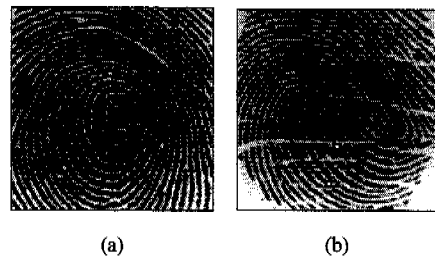


그림 10. flow-line 추적 결과 (a) Twin loop (b) Whorl

III. 실험 결과 및 고찰

제안한 분류 방법을 364×256 크기의 공간해상도를 가지고 256 gray scale을 가지는 700장의 지문 영상에 적용해 보았다. 실험은 Pentium III 700MHz의 CPU 속도를 가지는 IBM PC에서 수행되었다. 지문 영상의 획득은 휴노 테크놀로지사의 지문 입

력기(MagicSecure 2000)를 이용하여 한 사람에 대해서 5-10장의 영상을 획득하였다. 좌우 엄지손가락에 대해서 각각 똑바른 영상과 좌우로 회전 정도가 다른 2장의 영상을 취하여 손가락별로 총 5장의 영상을 획득하였다. Tented arch와 arch 형태의 지문 영상이 부족하여 일부는 엄지 외에 다른 손가락에서 취하였다(총 10장). 표 1에서 실험영상의 지문 부류별 분포 비율을 나타내었다. 제안하는 방법을 실험 영상에 적용한 결과가 표 2에 나타나 있다.

표 1. 지문 부류별 분포 비율

	A	TA	LL	RL	TL	W
Percentage[%]	4	1	20	15	26	34

표 2. 제안한 방법을 실험영상에 적용한 결과 분류율

	A	TA	LL	RL	TL	W	Total
Right	28	5	126	96	168	232	655
Wrong	2	0	14	9	17	3	45
Total	30	5	140	105	185	235	700

실험결과 93.6%의 분류율을 보였다. 분류 오류가 발생하는 경우는 지문 자체의 훼손이나 잡음으로 인하여 core point가 정확하게 검출되지 않거나 flow-line 추적이 적절하게 이루어지지 않은 경우나 사람이 보기에도 분류하기 모호한 형태의 지문이 존재하는 경우이다. 대부분의 오류는 whorl과 twin loop의 세부분류 과정에서 발생하는데 오류가 발생한 영상의 경우 whorl과 twin loop의 중간 형태이거나 적절치 못한 문턱치의 선택에서 오는 오류이다. 최근의 지문 영상의 분류에 관련된 문헌에서 자주 사용되는 NIST-4 데이터 베이스의 경우에는 twin loop가 존재하지 않는다. 따라서 twin loop와 whorl을 동일한 부류로 지정할 경우 오차 발생은 현저하게 줄어들게 된다. 그리고 reject option을 들 경우 영상의 오차율은 더욱 떨어지게 된다. 표 3에서 twin loop와 whorl을 동일한 부류로 지정할 경우의 분류율을 나타내었는데 약 95.8%의 분류율을 나타내었다.

Core point 검출과 flow-line 추적은 본 논문에서 제안한 방법을 사용하고 세부분류 단계에서 Kawagoe와 Tojo가 사용한 whorl과 twin loop를 구

표 3. 영상을 5개의 부류로 나누었을 경우 분류율

	A	TA	LL	RL	TL & W	Total
Right	28	5	126	96	417	672
Wrong	2	0	14	9	3	28
Total	30	5	140	105	420	700

분하는 조건^[8]을 사용할 경우 분류율이 약 82%로 낮게 나와 제안한 방법의 조건이 정교하게 지문 영상을 분류하는데 있어서 더 적절한 방법을 확인할 수가 있다.

실험에 사용된 지문 영상의 경우 공간 해상도가 364 × 256으로 비교적 낮을 뿐 아니라 압착날인된 지문영상이다. 따라서 대부분의 영상에 delta point가 나타나지 않아 기존의 특이점 정보를 이용한 분류 방법^[1]으로는 정확한 지문영상의 분류가 어렵게 된다. 본 논문에서는 특이점 중에서 일반적으로 지문 영상의 중심부에 위치해 거의 대부분 보존되는 core point 정보와 이 core point에서의 flow-line 정보를 이용해서 효과적으로 지문영상을 분류하여 93% 이상의 분류율을 나타내었다. 그리고 제안한 방법은 회전된 영상에 대해서도 높은 분류율을 나타내는데 이는 Poincaré index를 이용한 core point 검출 기법이 회전에 대해서도 core point를 잘 검출해 낼 뿐 아니라 flow-line을 이용하기 때문에 회전되더라도 성능에 큰 차이가 없이 잘 분류해 낼 수 있다. 단 left loop, right loop의 경우 심하게 회전되어 left loop 임에도 불구하고 flow-line의 끝점이 모두 오른쪽에 위치할 수도 있는데 이와 같은 경우에는 물론 오류가 발생하나 30도 이내의 회전에 대해서는 문제없이 분류해 낼 수 있다.

기존의 방법 중에 Gabor 필터링된 데이터를 가지고 FingerCode를 생성해 지문 영상을 분류하는 방법은 Gabor 필터링을 수행하는 시간만 적어도 수초가 소요되어 전체적인 처리 속도가 상당히 느린 반면에 본 논문에서 제안된 방법의 경우 core point를 검출하는 시간이 굉장히 짧은 뿐 아니라 flow-line 추적 또한 평활화된 블록의 정보를 이용하여 수행되기 때문에 처리 속도가 약 0.06초로 다른 방법에 비해 훨씬 빠르게 나타났다. 무엇보다도 사람이 지문을 분류할 때 사용하는 정보를 flow-line이 적절하게 잘 표현하고 있어서 미세한 분류에 유리하며 새로운 분류 기준을 마련할 경우 여섯 부류 이상의 분류에도 사용 가능하다는 장점이 있다.

IV. 결론

본 논문에서는 압착 날인된 지문 영상에 적합한 자동 분류 방법을 제안하였다. 제안된 방법은 크게 두 단계로 이루어져 있는데 첫 번째 단계는 Poincaré index를 이용한 core point의 검출 단계이다. 검출된 core point의 수를 바탕으로 1단계 분류를 수행한다. 1단계 분류를 통하여 core point가 존재하지 않는 arch형과 core point의 개수가 하나인 tented arch, left loop 그리고 right loop와 마지막으로 core point의 개수가 2개인 whorl과 twin loop 군으로 크게 세 가지 부류로 분류된다. 2단계는 flow-line 추적을 이용한 세부 분류단계로서 평활화된 블록의 방향 패턴을 이용한 flow-line 추적 알고리즘을 제안하였다. 제안한 방법을 압착날인된 700장의 지문 영상에 적용해 본 결과 93.6%의 분류율을 보였다.

일반적으로 지문 인식과정 중에 기준점을 정하는 과정이 포함되는데 core point가 기준점으로 많이 사용되고 있다. 제안한 방법은 이러한 기준점을 잡는 과정의 정보를 효율적으로 이용함으로써 개략적인 분류를 수행한다는 장점과 flow-line이 지문영상의 전체적인 패턴(global pattern)을 잘 표현한다는 장점을 이용한 방법으로 특히 압착 날인처럼 delta point가 유실되거나 회전된 영상에 대해서 우수한 성능을 나타내었다.

제안한 방법은 core point의 검출을 이용하기 때문에 지문영상에 잡음이 많은 경우 core point가 제대로 검출되지 않거나 잘못 검출되어 오류가 발생하게 된다. 따라서 잡음이 있더라도 core point를 효율적으로 찾을 수 있는 방법과 두 개의 core point 중 하나가 유실되더라도 flow-line을 이용해 효율적으로 분류할 수 있는 새로운 방법에 대한 연구가 필요하다.

참 고 문 헌

[1] K. Karu and A. K. Jain, "Fingerprint Classification," *Pattern Recognition*, vol. 29, no. 3, pp. 389-404, 1996.
 [2] A. K. Jain, S. Prabhakar, and L. Hong, "A Multichannel Approach to Fingerprint Classification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 4, pp.

348-359, Apr. 1999.

[3] K. Rao and K. Balck, "Type Classification of Fingerprints: A Syntactic Approach," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-2, no. 3, pp. 223-231, May 1980.
 [4] H.V. Neto and D.L. Borges, "Fingerprint Classification with Neural Networks," *Proc. IVth Brazilian Symposium on Neural Networks*, pp. 66-72, 1997.
 [5] A. Senior, "A Hidden Marcov Model Fingerprint Classifier," *Proc. 31st Asilomar Conf. Signals, Systems and Computers*, pp. 306-310, 1997.
 [6] A. P. Fitz and R. J. Green, "Fingerprint Classification Using Hexagonal Fast Fourier Transform," *Pattern Recognition*, vol. 29, no. 10, pp. 1587-1597, 1996.
 [7] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, "Fingerprint Classification by Directional Image Partitioning," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 402-421, May 1999.
 [8] M. Kawagoe and A. Tojo, "Fingerprint Pattern Classification," *Pattern Recognition*, vol. 17, no. 3, pp. 295-303, 1984.
 [9] A. K. Jain, L. Hong and R. Bolle, "On-Line Fingerprint Verification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp. 302-314, Apr. 1997.

박 철 현(Chul-hyun Park)

준회원



1996년 2월 : 경북대학교
전자공학과 졸업

1999년 2월 : 경북대학교
전자공학과 석사

1999년 3월~현재 : 경북대학교
전자공학과 박사과정

<주관심 분야> 영상 처리, 영상 인식, Biometrics

오 상 근(Sang-keun Oh)

준회원



1996년 2월 : 경북대학교

전자공학과 졸업

1998년 2월 : 경북대학교

전자공학과 석사

1998년 3월~현재 : 경북대학교

전자공학과 박사과정

<주관심 분야> 영상처리, 영상인식, 컴퓨터비전

이 경 환(Kyung-hwan Lee)

준회원



2000년 2월 : 경북대학교

전자공학과 졸업

2000년 3월~현재 : 경북대학교

전자공학과 석사과정

<주관심 분야> 영상처리, 영상인식, 디지털TV

김 현 순(Hyun-soon Kim)

정회원

1995년 2월 : 경북대학교 전자공학과 졸업

1997년 2월 : 경북대학교 전자공학과 석사

2001년 2월 : 경북대학교 전자공학과 박사

<관심분야> 영상 통신, 영상 인식, 의료 영상 처리

박 길 훈(Kil-houm Park)

정회원

한국통신학회 논문지 제24권 제9B호 참조

현재 : 경북대학교 전자전기컴퓨터공학부 교수