

## 체인 코드를 이용한 훼손된 지문의 인식에 관한 연구

조민환\*

### A Study on the Recognition of Defected Fingerprint Using Chain Code

Cho Meen Hwan \*

#### 요약

지문 인식 시스템에서는 대부분 융선의 끝점과 분기점을 이용한다. 그러나 융선을 체인 코드화하여 이것을 이용한 훼손된 지문의 인식에 관해 연구하였다. 처리의 결과와 순서는 다음과 같이 요약 할 수 있다. ① 몇 개의 훼손된 지문 획득, ② 전처리(잡음 제거를 위해 중간 값 필터링, 지역 또는 전역 히스토그램 균등화, 팽창과 침식, 세선화와 유사 이미지 제거), ③ 최소 자승법 처리 후 융선 재구성, ④ 체인 코드 벡터의 분포 계산, ⑤ 전 처리된 지문 이미지에서 각 벡터의 값들은 거의 같은 결과 값을 보였다. 이 결과로부터 우리는 특징점을 이용한 시스템과 결합하여 사용한다면 더욱 성공적인 지문 인식 시스템이 될 것이라 추측한다.

#### Abstract

Almost the system are usually taken by means of shapes and positions of ridge's end-points and bifurcation in the fingerprint recognition. but we studied about recognition of polluted fingerprint by chain code ridges. the results and sequence of processing are summarized as follows. ① Capture several kinds of polluted fingerprint image. ② Preprocessing(median filtering for removing noises, local and global histogram equalization, dilation and erosion, thinning and remove pseudo image), ③ Rebuild ridge line after Least Square Processing, ④ Compute distribution of chain code vector, ⑤ The results are almost same values of each vector of preprocessed fingerprint images. From the results, we can surmised more successful fingerprints recognition system in combination with other system by singular points

▶ Keyword : 지문인식(Fingerprint Recognition), 방향성코드(Chain Code), 융선(Ridge)

---

\* 창원전문대학 사무자동화과 조교수

## I. 서론

지문은 개인 식별을 위한 수단으로 많이 사용되어 왔다. 지문인식은 컴퓨터의 발달과 더불어 급속도로 연구가 진행되었으며, 인식 결과에 대한 신뢰도와 안정도가 다른 수단보다 높은 것으로 평가되어 많은 알고리즘들이 개발되어 왔다. 지문의 식별에 대한 연구는 대부분 융선(ridge)의 끝점, 분기점 등의 위치 및 방향 정보, 융선의 모양 등을 추출하여 인식의 기초 데이터로 사용한다[1][3].

지문인식에서 사용되는 융선의 흐름형태에 근거한 특징 정보는 일정한 영역을 대표하는 주된 흐름선을 추출하고, 이 흐름선의 전체 형태에서 융선의 구조적, 형태적 특징들을 추출하여 인식의 기초자료로 사용한다. 이 방법은 지문 융선의 전체적인 형태 정보를 이용하기 때문에 특징점들이 어느 정도 훼손된 경우라도 인식이 가능하며, Isenor와 Zaky는 융선을 그래프로 나타내어 조합을 실시하였다[2].

특징점에 근거한 방법으로 단점, 분기점등의 특징점을 찾아내어 특징점의 종류 및 위치 관계에 의해서 지문을 분류하는 방법으로 현재 많은 연구가 이루어졌다[5][6]. 이 방법은 잡음에 강한 장점이 있으나 전체 구조를 고려하기 힘들고 특징점을 잃어버린 경우에 식별하기 어려운 단점이 있다. 특징점 추출은 전처리(preprocessing), 특징점 추출(feature extraction), 후처리(postprocessing)등의 3단계로 구성되며, 전처리 과정은 이미지 필터링을 위한 지문 영상들의 공간의 방향과 가능한 변수들을 고려하여 융선(ridge)과 골(valley) 성분을 추출한다. 특징 추출을 위한 영상은 임계치 설정(thresholding), 세선화(thinning), 유사정보 제거 등 전처리를 한 후 상세점을 추출한다[4].

본 연구에서는 훼손된 동일한 지문형태를 중심으로 지문 이미지의 손상으로 발생 될 수 있는 특징 및 정보들이 이미지 인식에 최소한 영향을 주는 방법에 관하여 연구하였다. 훼손된 이미지의 세선화는 다른 전처리 과정과[7] 동일하나 세선화 이미지를 세선화 이미지를 최소 지승법을 사용하여 융선을 재구성하고 이를 기초로 체인코드(chain code)화 하였다. 체인코드의 벡터 성분의 분포는 동일한 지문일 경우 손상이 되었을 경우에도 거의 변화가 없을 것이란 가정에서 실험하였다. 그 결과 우리가 가정 한 것이 타당한 것이 입증되었다.

## II. 전처리(preprocessing)

### 1. 지문의 형태에 따른 종류와 전처리

지문은 보통 <Fig.1>과 같이 4가지로 분류되며 지문의 위·아래 중심점(upper·lower core)과 삼각주(delta)를 정의하고, 지문에 존재하는 삼각주의 존재와 위치와 이들의 상관 관계로부터 패턴을 분류하게 된다.

Arch란 삼각주가 존재하지 않는 형태이며, Tented Arch는 삼각주가 중심점과 같은 선상에 있는 지문이고, Left loop는 중심점이 삼각주의 왼쪽, Right Loop는 오른쪽에 놓인 형태이다.

Whole은 동심원 형태로 삼각주가 2개 중심점의 좌우 측에 있는 경우가 대부분이다[1].

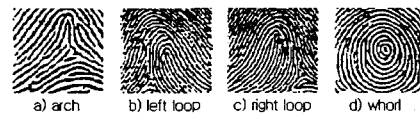


그림 1. 지문특성의 종류  
Fig. 1 Kind of fingerprints by feature

지문인식에서의 전처리 과정은 256레벨에서 흑백영상을 이진 영상으로 변환할 경우 지문을 채취할 때 많이 발생되는 잡음의 영향으로 지문의 선이 겹치거나, 잉크의 퍼짐 성질로 인한 결과 지문 채취 시 눌러지는 압력에 의해 지문의 형태가 다르게 나타나게 되며, 생활에 있어서 여러 가지 원인에 의해 지문이 손상되어 지게 된다(Fig 2). 이 결과 부분적으로 선이 사라지거나, 유사정보가 생성된다.

<Fig. 3>은 훼손된 지문들의 정보를 제거하고 방향코드 추출을 위한 처리의 순서를 보여준다.

본 논문에서는 전처리를 위해 CVIP 툴을 이용하여 전처리를 시도하였으며, phoitishop utility를 이용하여 원 이미지에 여러 가지 조작을 가하여 각 훼손된 지문이미지에서 결과를 비교하였으며, 방향코드 추출을 위하여 세선화 영상에서 최소 지승법을 이용하여 방향코드를 구하였다.

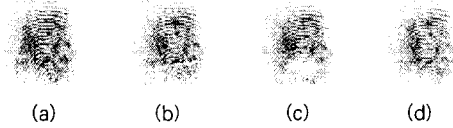


그림 2. 원본 이미지  
Fig. 2 Original Image

- |   |
|---|
| (1) 영상 데이터 입력(256 × 256)<br>(2) Median Filtering<br>(3) Local Histogram Equalization<br>(4) Histogram Equalization<br>(5) Binarization<br>(6) Morph Closing Circle<br>(7) Thinning<br>(8) Rebuild ridges by least square method<br>(9) Vector generation<br>(10) Decision |
|---|

그림 3. 처리순서  
Fig. 3 Process

## 2. 중간 값 필터(Median filtering)

저 대역 통과 필터는 가우시안 잡음을 제거하는 데는 적합하지만 임펄스 잡음을 제거하기 위하여 저주파 통과 필터를 사용한 후 기존의 에지들을 좀더 상세하게 보전 할 수 있는 중간 값 필터를 이용하여 지문 선의 폭이 범위를 벗어 나지 않게 하기 위하여 5×5크기의 마스크를 이용하여 마스크 크기의 25개의 값 중에서 가운데 값을 찾아 잡음을 제거하였다. <Fig. 4>는 중간 값 필터 처리결과를 보여준다.

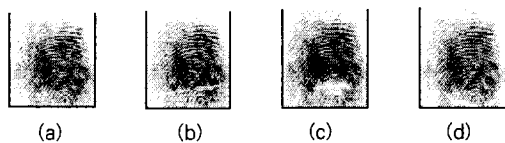


그림 4. 중간값 필터의 결과  
Fig. 4 Result of Median Filtering

## 3. 히스토그램 평활화

지문은 채취시 손가락의 힘이 영상에 끌고루 분포되지 않기 때문에 256 레벨의 분포가 다르게 나타나므로 부분별로 밝기가 달라진다. 이러한 경우 히스토그램 평활화 과정을 거쳐 균일한 값으로 향상시킬 수 있다.

본 논문에서는 8×8 마스크를 이용하여 지역 히스토그램 평활화 <Fig. 5>를 한 후 좀 더 효율적으로 방향코드를 추출하기 위하여 전역 히스토그램 평활화 <Fig. 6>를 한번 더 시도하였으며, <Fig. 7>의 히스토그램 그래프에서 지역 히

스토그램 평활화를 한 후 전역 히스토그램 평활화를 수행한 것이 이진화시의 정보손실을 줄일 수 있다는 것을 알 수 있으며, 동일한 지문일 경우 히스토그램의 형태가 유사함을 알 수 있다.

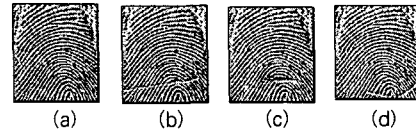


그림 5. 지역 히스토그램 평활화  
Fig. 5 Result of Local Histogram Equalization

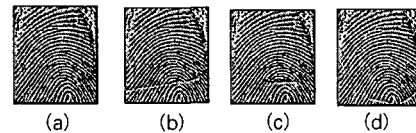


그림 6. 전역 히스토그램 평활화  
Fig. 6 Result of Histogram Equalization



(a) Histogram equalization  
(b) Local histogram equalization  
그림 7. 지역 히스토그램 평활화 후 전역 히스토그램으로 수행 결과  
Fig. 7 Histogram of local histogram equalization and histogram equalization

## 4. 이진화(Binarization)

방향 코드를 추출하기 위한 이진 단계인 평활화 과정을 거친 이미지는 임계치 및 이진화(binary) 과정을 거쳐 검은 색의 융선(ridge)과 흰색의 골(valley)로만 이루어진 이진화 이미지로 변환 되어야한다.

이진화는 히스토그램을 토대로 하여 특정 임계값을 설정하여 검정색과 흰색으로 나누게 된다. 임계치 설정을 위하여 <Fig 6>의 전역 평활화를 거친 영상에서 히스토그램의 최대 값과 최소 값을 분석하고 두 위치 값에서 중심 값을 파악한 다음 기준 레벨로 선정하면 <Fig 8>에서와 같이 가장 적절한 이진화 영상을 획득할 수 있다.

본 논문에서는 각 이미지의 히스토그램을 토대로 전체적으로 임계치 값을 128로 하여 이진화 이미지를 얻었다.

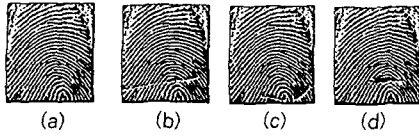


그림 8. 이진화 이미지  
Fig. 8 Binary images

M8	M1	M2
M7	M0	M3
M6	M5	M4

그림 10. 세선화를 위한 3×3블럭 마스크  
Fig. 10 3×3 block mask for thinning

5. Morphological filter

형태학적 열림(opening)과 닫힘(closing) 연산에서 열림은 일반적으로 영상의 외곽선을 부드럽게 하고, 좁은 지협들을 쪼개고, 얇은 돌출 부분들을 제거하고, 닫힘은 외곽선 부분들을 을 부드럽게 하는 경향이 있으나, 열림에 반하여 일반적으로 좁은 지협들과 길고 얇은 만들을 합치고 작은 구멍들을 제거하고 외곽선의 간격들을 메운다.

<Fig.9>의 이미지는 <Fig. 8>의 이진화 이미지에서 용선들의 거친 부분들을 부드럽게 하고, 잡음이 제거된 것을 볼 수 있다. <Fig. 9>는 이러한 결과를 이미지로 나타내었다.



그림 9. 이진 이미지의 형태상 이미지  
Fig. 9 Morphological filter image of binary image

III. 세선화 및 최소 자승법을 이용한 용선 재구성

1. 세선화(thinning)

세선화란 두꺼운 선을 하나의 선으로 표현하는 것으로 지문 용선의 연결성은 변화시키지 않고 선으로 변화시키는 것이 필요하다. 세선화는 지문 용선의 결합 관계 특징이 구해지기 때문에 지문 용선의 구조를 해석하는데 필요한 처리 과정이다. 본 논문에서는 이진화를 거친 영상의 세선화를 위해 3×3 마스크를 조사하여 각 영역이 세선화 될 때까지 각 영역의 경계선을 한번에 한 픽셀씩 벗겨내는 처리를 반복적으로 되풀이한다. <Fig. 10>는 세선화 알고리즘의 픽셀들의 배열을 마스크 형식으로 나타낸 것이다.

이웃 픽셀들이 특정 픽셀과 이웃 할 때(식 3-1)과 같이 계산한다.

$$N(M1) = \sum_{i=2}^9 Mi \dots\dots\dots (식 3-1)$$

여기서  $N(M0)$ 는  $M1, M2, M3, M4, M5, \dots, M8$ , 의 정렬된 순서에서 0→1천이의 회수이다. 여기서다음의 조건(식 3-2)를 만족하면 단계 1은 외곽선 점  $M0$ 를 삭제하기 위해 표시해 놓는다.

$$2 \leq N(M0) \leq 6 \wedge (N(M0) = 1) \\ \wedge (M1 \cdot M2 \cdot M3 = 0) \wedge (M3 \cdot M5 \cdot M7 = 0) \\ \dots\dots\dots (식 3-2)$$

단계 2에서  $2 \leq N(M0) \leq 6 \wedge N(M0) = 1$ 은 같으나  $M1 \cdot M2 \cdot M3 = 0$  은  $M1 \cdot M3 \cdot M7 = 0$ 로,

$M3 \cdot M5 \cdot M7 = 0$  은  $M1 \cdot M5 \cdot M7 = 0$ 로 바뀌게 된다. 세선화 과정의 알고리즘은 먼저 경계점들의 삭제를 표시하고, 표시된 점들의 삭제, 남겨진 경계점들의 삭제표시를 위해 2단계 적용, 표시된 점들의 삭제로 이루어지는데 이 기본 과정이 더 이상 삭제 될 점이 없을 때까지 반복적으로 적용된다.

<Fig 11>은 세선화된 지문의 골격을 나타낸 것이다.

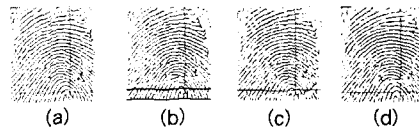


그림 11. 이진 이미지의 세선화  
Fig. 11 Thinning of binary image

2. 최소 자승법을 이용한 용선 재구성

<Fig. 11>에서 세선화된 용선의 모양은 매우 거칠어 이 영상에서 체인 코드를 만들 경우 코드의 연속적인 변화가 노이즈에 의해 많은 영향을 받게된다. 이러한 경우 최소 자승법에 위한 용선의 재구성은 노이즈의 영향을 최소화 할 수 있다. <Fig. 12>는 <Fig. 11>의 용선 이미지를 최소 자승법을 이용하여 재구성한 것을 보여주고 있다. 이때 사용된 최소 자승법은 2차 방정식을 사용하였다.

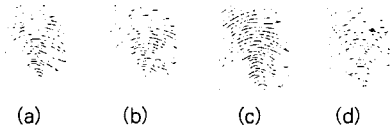


그림 12. 최소 자승법 의한 재구성  
Fig. 12 Rebuild ridges by least square method

### IV. 방향코드 생성 및 결과 분석

#### 1. 방향코드 생성

체인코드(chain code)는 미리 정한 길이와 방향을 가지는 직선의 선분들의 연결된 열로서 경계를 표현하는 데 사용된다.

선분들의 4-연결성, 8-연결성에 기반을 두며, (Fig. 13)과 같이 수평방향을 0벡터, 0벡터에서 시작하여 시계 반대 방향으로 벡터 1, 2, 3, ..., n, 등 순차적으로 표시하였다. 본 연구에서는 8-연결성을 기반으로 하여 방향코드를 생성하였다.

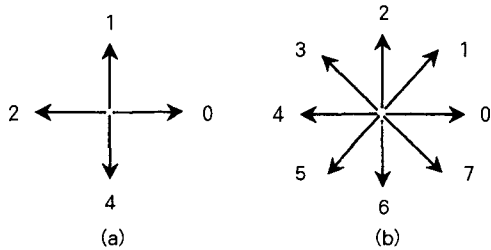


그림 13. (a) 4-방향체인코드, (b) 8-방향체인코드  
Fig. 13 4-direction chain code, 8-direction chain code

#### 2. 결과 분석

본 연구에서는 동일인의 지문일 경우 훼손된 이미지의 세션화는 다른 전처리 과정과 동일하나 세션화 이미지를 최소 자승법을 사용하여 (Fig. 12) (a)(b) (c)(d)의 이미지와 같이 용선을 재구성하고 이를 기초로 8-방향 체인코드(chain code)화 하였다. 체인코드의 벡터 성분의 분포는 동일인 지문일 경우 손상이 되었을 경우에도 거의 변화가 없을 것이라 가정에서 실험을 한 결과 <표 1>과 같은 결과를 도출하였다.

표 1. 각 이미지에 따른 방향 코드 개수  
Table 1. Images vs Quantity of Change Code  
(단위: %)

이미지 방향	a	b	c	d
0번 벡터	39.06	38.65	38.26	36.92
1번 벡터	18.82	17.68	16.53	18.75
2번 벡터	0.1	0.4	0.0	0.2
3번 벡터	0.0	0.0	0.0	0.0
4번 벡터	0.0	0.0	0.0	0.0
5번 벡터	0.0	0.0	0.0	0.0
6번 벡터	0.10	0.14	0.11	0.14
7번 벡터	41.9	43.12	45.05	44.01
처리후전체합	4091	4316	4373	4331

<표 1> 위 결과에서 보면 동일인의 지문일 경우 체인 코드의 벡터 성분의 분포는 거의 변화가 없다는 것을 알 수 있었다. <표 1>에서 원 이미지 (a)의 총 벡터수는 4091이며 훼손선 이미지 (b), (c), (d)도 또한 약 4,000개로 거의 변화가 없었다. 각 벡터 단위는 (i 벡터 / 전체 벡터) × 100로 % 단위로 표시하였다.

### V. 결론

지문 인식에 사용되는 식별 정보는 대개 용선 위 끝점과 분기점의 위치와 방향을 사용한다. 이 경우 용선의 일 부분이 지워지거나 훼손으로 인하여 붙어 버리는 경우 인식의 기초 자료인 끝점과 분기점이 새로이 생성되고, 일부는 없어지므로 인하여 잘못 인식되는 경우가 많이 있다. 본 논문에서는 지문이 일부 훼손되어도 용선의 전체적 형태는 변화하지 않으므로 용선의 방향코드에 기초한 인식 방법에 관하여 연구하였다. 연구 결과 지문이 훼손되어도 전체적 방향 코드의 분포는 변화가 없었고, 대부분 전처리 과정에서 발생하는 용선의 소멸, 생성 등에도 영향을 받지 않았다. 그리고 각 방향 코드의 퍼센트 결과는 훼손이 된 경우 0방향에서 오차율이 ±0.022였으며, 1 방향이 ±0.1, 6방향이 ±0.3, 7방향이 ±0.07이므로 전체적으로 큰 영향이 없음을 알 수 있다.

이러한 결과로 다른 지문인식 알고리즘과 결합할 경우 훼손된 지문의 인식율이 많이 증가할 것으로 입증된다.

## 참고문헌

- [1] Kalle Karu, Anil K.jain, "Fingerprint Classification," Pattern Recognition 29(3), 389-404, 1996
- [2] Masahiro Kawagoe, Akio Tojo. "Fingerprint Pattern Classification," Pattern Recognition 17(3), 295-404, 1984
- [3] Refael C. Gonzalez & Richard E. Woods, "Digital Image Processing". pp. 497-499, 530-531. 1992
- [4] L. Hong, Y. Wan, and A. Jain, "Fingerprint image enhancement: Algorithm and evaluation," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 20, no. 8, pp. 777-789, 1998
- [5] Shlomo Greenberg, Mayer Aladjem, Daniel Kogan and Itshak Dimitrov "Fingerprint Image Enhancement using Filtering Techniques" Eletrical and Computer Engineering Department, Ben-Gurion University of the Negev, Beer-Sheva, Israel
- [6] A.K Hrechak, "Automated Fingerprint Recognition Using Structural Matching", Pattern Recognition, 23 1990
- [7] 김봉일, 이상선, 김정규, "세션화된 지문 화상의 단계적 복원 처리" 정보 과학회지, 제4권, 1호, pp33-44, 1987

## 저자소개



### 조민환

1989년 3월 숭실대학교  
산업대학원 전자계산학과(공학석사)  
1994년 3월 현재 창원전문대학  
사무자동화과 교수  
2001년 2월 경남대학교 대학원  
컴퓨터공학과(공학박사)