

# 지문인식을 위한 다차원공간 인덱스의 설계

김갑영<sup>U</sup>, 심현보, 박영배  
 명지대학교 컴퓨터공학과

## Multidimensional Index for Fingerprint Identification

Kap-Young Kim<sup>U</sup>, Hyun-Bo Shim, Young-Bae Park,  
 Dept. of Computer Engineering, Myongji University

### 요약

지문은 가장 효율적인 사용자 인증 방법으로 이용되어져 왔다. 또한 컴퓨터의 발달과 더불어 지문 인식은 더욱 많이 연구되어졌고, 또한 급속도로 발전하였다. 그러나, 이런 대부분의 연구들은 지문 인식에서 특징점 추출 및 정합부분에 관한 연구가 주류를 이루고 있다. 그러나, 대단위 데이터베이스 검색 향상을 위한 방법의 연구는 아직도 미진한 실정이다. 본 연구는 기존의 지문 데이터베이스에 공간 인덱스를 추가시켜 지문을 효과적으로 추출하는 방법을 제시한다. 이 방법으로, 데이터베이스의 지문을 다차원공간 인덱스에 저장시킨다. 그리고, 지문을 검색할 때에는, 다차원공간상에서 미지 지문과 유사한 지문들을 추출하여, 후보지문을 만든다. 그리고, 매칭 작업은 이 후보지문들하고만 매칭 하여도 전체를 한 것과 같은 효과를 얻을 수 있다. 이러한 방법으로, 전체 지문 검색 시간을 단축시킬 수 있다.

### 1. 서론

지문은 독특함과 불변성의 특성을 지녔기 때문에 ER Henry가 파생법을 사용하여 지문인식을 사용한 이래 지난 100년간 가장 효율적인 개인 인증의 수단으로 사용되고 알려져 왔다. 지문 인식은 컴퓨터의 발달과 더불어 급속도로 연구가 진행되었으며 인식 결과에 대한 신뢰도와 안정도가 다른 수단보다 높은 것으로 평가되어 많은 알고리즘들이 개발되어 왔다. 그러나, 이러한 연구들은 대부분 지문 인식에서 지문 특징점 추출 및 정합부분에 관한 연구들이 주를 이루고 있다. 범죄 수사 와 같은 대단위의 지문이 필요하고, 이를 검색해야 하는 곳, 또 On-Line 시스템은 지문인식이 판단 및 인증의 수단으로 사용되기 때문에 정확한 판단 및 지문을 검색하는데 소요되는 시간이 매우 중요하다. 대량 데이터베이스에서 지문사용의 필요성이 점점 더 증가추세에 있으며, 그 중요성이 강조되고 있는 실정이다. 그러나, 이런 분야의 연구들은 그다지 많이 이루어지고 있지 않아 지문인식에 비해 대량 지문 데이터베이스에서의 지문 검색 방법은 많이 뒤떨어져 있는 실정이다.

본 논문은 지문 검색의 효율성을 높이기 위해 지문 데이터를 다차원 공간에 저장하는 방법과 이 저장 방법을 사용하여 어떻게 하면 효과적으로 지문을 검색할 수 있는지 그 방법을 제시하고자 한다.

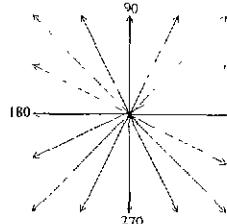
### 2. 다차원공간 인덱스

#### 2.1. 지문의 특징점 표현 방법

지문의 매칭과 검색을 위해 지문을 저장할 때는 특징점을 저장하는 것이 일반적이다. 특히 자동 지문 인식시스템에서는 주로 지문의 단점(Ridge Ending)과 분기점(Ridge Bifurcation)만을 가지고 인식을 한다. 물론 지문에는 다른 특징(짧은 용선,

고딕, 교차 등)들도 있지만, 자동지문인식시스템에서 이 단점과 분기점의 조합으로 나타낼 수 있기 때문에 포함시키지는 않는다.

특징점은 단점과 분기점으로 나누어지고, 그 점들은 그 위치(2차원 좌표)와 방향성(기울기)을 이용하여 나타낸다. 위치는 지문의 중심점으로부터 각각의 특징점들이 떨어진 각도를 X, Y값으로 표시하는 상대 좌표를 사용한다. 그리고 방향성은 단점이나 분기점의 용선 방향의 기울기를 몇 단위로 구분하여 구성한다. 본 연구에서는 방향성을 [그림 1]에서와 같이 16방향으로 나눈다. 그리고 단점과 분기점을 각각 16개의 방향성으로(각각을 22.5°의 각도로 나눈다) 단점은 0~15, 분기점은 16~31의 값으로 표현하며, 종래와 같이 특징점의 종류를 구분하는데 이 같이 나타내어지는 지문인식에서 특징점들은 X, Y, Z와 같은 값으로 표현하며, 여기서 X, Y는 위치가 나타내어졌는지를 나타내며, Z는 특징점의 종류가 되는데, 그 값은 16에서와 같이 나타낸다.



[그림 1] 특징점 방향성

좌표 값이 되고, Z는 특징점의 종류가 되는데, 그 값은 16에서와 같이 나타낸다.

[표 1] 지문 특징점 값

각도	0	22.5	45	67.5	90	...	...	315	337.5
특징점	단점	0	1	2	3	4	...	14	15
	분기점	16	17	18	19	20	...	30	31

예를 들어, 분기점이 지문 중심점으로부터 X축으로 30

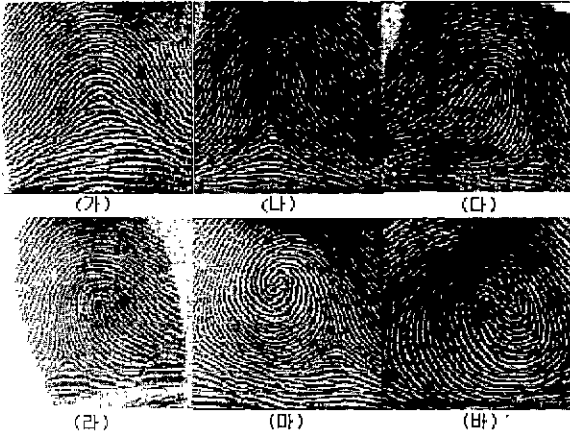
므로 150 떨어진 곳에 있고, 기울기가 45°라면, 그 표현은 (30, 150, 18)이 된다.

지문 표현의 일반식은 위에서 말한 특징점들의 집합으로 나타내게 되는데 그 표현 방법은 다음과 같다.

지문 =  $\{(X_1, Y_1, Z_1), (X_2, Y_2, Z_2), (X_3, Y_3, Z_3), (X_4, Y_4, Z_4), \dots, (X_{n-1}, Y_{n-1}, Z_{n-1}), (X_n, Y_n, Z_n)\}$

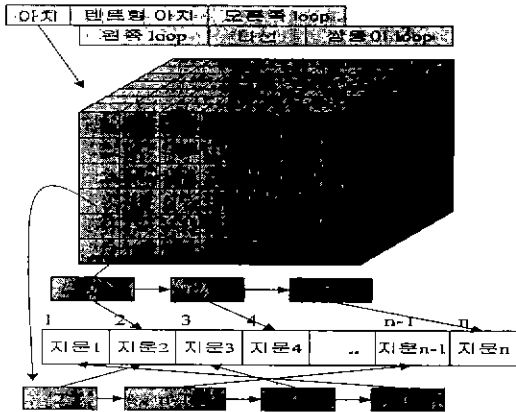
2.2. 지문의 저장 방법

2.1에서와 같이 표현된 지문들을 3차원 지문저장공간에 저장시킨다. 검색의 속도를 높이기 위해 이 3차원 저장공간을



[그림 2] (가)아치 (나) 텐트형 아치 (다) 오른쪽 Loop (라) 왼쪽 Loop (마) 나선 (바) 쌍둥이 Loop

[그림 2]와 같이 아치, 텐트형 아치, 오른쪽 Loop, 왼쪽 Loop, 나선, 쌍둥이 Loop등으로 분류되어진다.



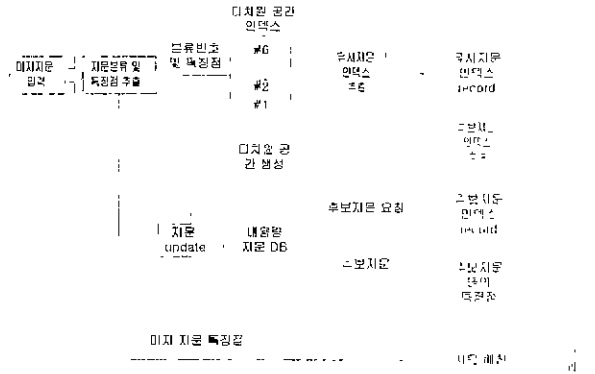
[그림 3] 다차원 공간 인덱스

공간상에 위치한 특징점들이 각각 어떤 지문의 특징점인지 판별할 수 있어야 한다. 그러기 위해 실제 저장 방식을 정리할 하면 [그림 3]과 같이 나타낼 수 있다. [그림 3]의 최상층 영역의 블록은 분류된 지문을 나타내며, 이 블록들 각각은 3차원 공간을 참조하게 된다. 그리고, 3차원 공간은 6면체 모양으로 표현하게 되는데, 각 블록들은 공간의 한 좌표가 되는 것이다. 그리고, 각 블록은 이 블록(좌표)을 포함하는 지문을 참조하게 된다. 그런데, 이 블록은 다른 지문과 공유될 수도 있기 때문에, 각 블록들은 이런 중복된 지문들을 모두 나타내기 위

해서, 같은 블록을 가지고 있는 지문들의 리스트를 만들어서 블록이 이 리스트를 참조하도록 하여 중복된 표현을 할 수 있게 한다. 이 리스트는 실제로 지문을 저장하는 것은 아니고, 저장된 지문 특징점들의 위치 정보(인덱스)만을 가지고는 된다. 이런 방식으로 지문들을 6가지로 분류된, 3차원 저장 공간에 저장하게 되는 것이다.

2.3. 검색 방법

지문 검색의 절차는 [그림 4]와 같다. 이 검색 방법의 특징은 실제 지문 매칭 작업이 후보 지문들과의 매칭으로 지문 검색이 이루어진다는 것이다.



[그림 4] 검색 절차

2.3.1. 지문 분류 및 특징점 추출

먼저 지문 이미지가 입력이 되면 이 지문의 종류를 판별한다. 본 논문에서는 구조적 방법[3]을 사용하여 지문 분류한다. 이 구조적인 방법은 전체적인 구조를 고려한다는 이점이 있으며, 작은 이능, 회전, 왜곡에 강하다는 장점이 있다. 이 분류된 지문은 각 분류별 다차원 공간 인덱스에서 검색된다. 그리고, 지문 이미지에서 특징점들을 추출할 때, 추출된 특징점들을 위에서 제시한 방법대로, 특징점의 X, Y, Z와 특징점의 종류 Z로 이루어진 다음과 같은 집합으로 표기한다.

$\{(X_1, Y_1, Z_1), (X_2, Y_2, Z_2), (X_3, Y_3, Z_3), (X_4, Y_4, Z_4), \dots, (X_{n-1}, Y_{n-1}, Z_{n-1}), (X_n, Y_n, Z_n)\}$

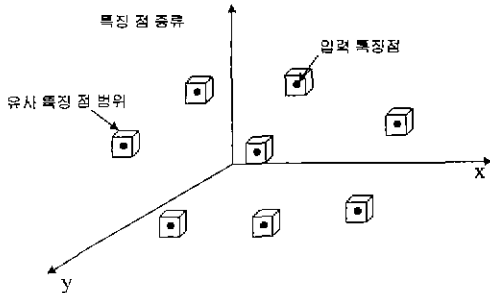
2.3.2. 유사지문 추출

위와 같이 이미 지문 특징점들의 집합을 가지고 다차원 공간 인덱스에 저장된 지문들을 검색한다. 검색 방법은 이 공간에서는 저장된 지문과 정확하게 매칭되는 것을 하나씩 찾내려고 비교하여 찾는 것이 아니라, 이미 입력 지문의 여러 가지 변형을 고려하여 그것과 비슷하다고 생각되는 지문들을 추출한다. 여기서 비슷한 지문이라는 것은 입력 지문의 특징점들과 3차원 공간상에서 근접한 위치에 특징점들을 가진 지문과 입력 지문과 유사성을 가지고 있다고 생각할 수 있다. 이 때, 이미 입력 지문의 모든 특징점들을 [그림 5]에서의 공간상의 공간상에 나열한다. 그리고, 이 특징점들과 유사성이 있는 지문과 가까운 지문들과 그 주변에 있는 특징점들을 그 지문들을 모두 추출한다. 이렇게 함으로써 불필요한 지문들까지 모두 찾지 않고 내가 찾고자 하는 지문과 유사한 지문만을 찾아내는 것이다. 입력된 지문의 각 특징점과 유사성을 갖는 특징점은 다음 식을 이용하여 찾을 수 있다.

$$(x, y, z) \Rightarrow (x', y', z')$$

(단:  $x - \alpha \leq x' \leq x + \alpha, y - \beta \leq y' \leq y + \beta, z - \gamma \leq z' \leq z + \gamma$ )

위 식과 같이  $\alpha, \beta, \gamma$ 의 값에 따라 정해지는  $x', y', z'$ 를 모두 만족하는 특징점들을 갖는 모든 지문들을 추출해 내는 것이다. 즉, 예를 들면, 입력 지문의 특징점이 (140, 150, 10)이고,  $\alpha, \beta, \gamma$ 의 값이 각각 5, 5, 3이라고, 할 때  $x'$ 는 135에서 145,  $y'$



[그림 5] 유사 지문 추출 범위

는 145에서 155,  $z'$ 는 7에서 13사이를 모두 만족하는 범위 안에 있는 특징점들을 갖는 지문들을 모두 추출하게 되는 것이다. 추출은 범위 안의 모든 인덱스가 가리키는 리스트들의 지문인덱스만 추출한다. 공간상에서 살펴보면 [그림 5]에서처럼 입력 특징점을 중심으로, 육면체 안에 포함되는 특징점들을 갖는 지문들을 추출하게 된다. 이렇게 추출된 지문 인덱스들은 중복되어 추출되는 것들이 있는데, 이것을 각 지문 인덱스에 따라 추출되는 빈도수로 기록하면 [표 2]과 같다

[표 2] 검출된 지문의 빈도수

지문번호	6	9	7	3	8	11	18	20	5	17	10	13	14	2	4	12
빈도수	4	30	15	5	3	2	4	4	3	2	3	3	2	2	2	2
지문번호	15	1	19	16												
빈도수	4	2	1	1												

이 육면체의 크기는  $\alpha, \beta, \gamma$ 값에 의해 결정되므로, 이 크기에 따라서 지문을 검색을 할 때, 속도와 정확성이 달라지게 된다. 육면체의 크기가 커지면 정확성은 증가하게 되지만 많은 지문 인덱스가 추출되어 검색의 속도가 느려질 수가 있다. 또 반대로 육면체의 크기가 줄어들면 추출되는 지문의 수는 감소하여, 검색의 속도는 높일 수 있으나 찾고자 하는 지문이 빠질 수도 있기 때문에 정확성은 떨어진다고 볼 수가 있다. 그렇기 때문에  $\alpha, \beta, \gamma$ 의 값을 정하는 것은 중요하다.

2.3.3. 후보지문 선택

내가 찾고자 하는 지문은 이 중의 하나일 것이다 그 중에서 제일 빈도수가 높은 지문으로 선택된 것이 우리가 찾고자 하는 지문일 확률이 가장 높다. 그리고, 그 다음으로 빈도수가 많은 지문이 그 다음으로 확률이 높을 것이다 즉, 빈도수가 높은 지문일수록 우리가 찾고자 하는 지문일 수 있는 확률이 높기 그렇

[표 3] 빈도수 정렬

지문index	9	12	7	3	6	15	18	20	5	8	10	13	1	2	4	11
횟수	30	20	15	5	4	4	4	4	3	3	3	3	2	2	2	2
지문index	14	17	19	16												
횟수	2	2	1	1												

기 때문에 반대로, 빈도수가 확연하게 적은 지문 인덱스들은

우리가 찾고자하는 지문일 확률이 거의 없다고 볼 수가 있다. 따라서, 확률이 거의 없는 지문들은 검색에서 제외시킬 수 있기 때문에, 찾고자 하는 지문일 확률이 높은 것들을 골라 후보 지문군으로 만든다. 그러기 위해서, 제일 많이 추출된 지문으로부터 적은 수의 지문으로 정렬을 해 지문을 [표 3]처럼 나열한다. 그리고, 이 나열된 것 중에서 앞에서부터 후보 지문들로 선택이 되는데, 이 후보지문의 선택도 그 선택되는 숫자에 따라 속도와 정확성이 달라지게 된다. 보통 이 후보를 선택을 할 때는 입력 지문 특징점 수의 2/3이상의 빈도 수를 가진 것만 선택하거나, 빈도수가 높은 것부터 일정한 값을 주어 선택하는 방법이 있을 수 있다.

2.3.4. 후보 지문 매칭

선택된 후보 지문들을 순서대로 지문 매칭기법을 사용해서 실제 내가 찾고자 하는 지문이 어떤 것인지를 확인해야 한다. 본 연구에서는 [1]에서 사용한 방법으로 실제 매칭작업을 수행해 나간다. 이 후보 지문들에서 지문을 찾지 못했을 경우에는 선택 지문이 데이터베이스 내에 없거나, 지문에서 오류가 발생할 경우이다. 이 오류는 분류가 잘못되었거나, 많은 특징점들이 겹침에서 추출되어져 발생할 수 있다. 이런 경우는 특별히 입력 지문 이미지가 손상이 많이 간 경우이거나, 너무 많은 회전 변화가 일어난 경우가 대부분이다. 전자의 경우 각각의 분류값들을 모두 검색하여야 하고, 후자의 경우는 모든 지문들을 검색해 주어야 하는 경우도 있다.

3. 결론

본 연구는 더 효율적으로 지문 데이터베이스를 검색하는 방법에 주안점을 두어 연구하였다. 이 빠른 검색을 위해 본 연구는 다차원 인덱스 저장 공간을 제안하였는데, 이 곳에 실제 지문들의 특징점들을 위치시키게 된다. 그리고, 검색을 할 때에, 입력 지문의 특징점을 이 공간에 투영시켜 입력 지문과 유사 지문들을 추출을 하고, 이 추출된 지문들 중에 확률이 높은 빈도수가 많은 지문들을 가지고 후보지문을 만들어 이 후보지문에서만 실제의 매칭 작업을 시행하게 된다. 이렇게 해서 필요한 검색에 소모되는 시간을 줄이게 되는 것이다

참고 문헌

- [1] Anil Jain, Lin Hong, Ruud Bolle, "On-Line Fingerprint Verification," IEEE Transactions on pattern analysis and machine intelligence, VOL.19, NO. 4, Apr 1997
- [2] Nalini K. Ratha, Kalle Karu, Shaoyun Chen, Anil K. Jain, "A Real-Time Matching System for Large Fingerprint Databases," IEEE Transactions on pattern analysis and machine intelligence, VOL 18, NO. 8, Aug. 1996
- [3] 문명권, 정규식, "구조적 방법에 근거한 지문영상의 손가락 인 분류," 정보과학회논문지(B) 제24권 제2호 1997. 2
- [4] Dario Maio, Davide Maltoni, "Direct Gray-Scale Minutiae Detection In Fingerprints" IEEE Transactions on pattern analysis and machine intelligence, VOL 19, NO. 1, Jan. 1997
- [5] <http://myhome.shnbiro.com/~freemen/index.html>
- [6] 채종진, 박래홍 "Ridge-line을 이용한 계층적 지문인식," 정보과학회 논문지 제18권 제5호 1991. 9
- [7] Raffaele Cappelli, Alessandra Lumini, Dario Maio, "Fingerprint Classification by Directional Image Partitioning," IEEE Transactions on pattern analysis and machine intelligence, VOL. 21, NO.5 May 1999