

이진화된 지문이미지의 효율적 처리를 위한 정합알고리즘

박종민, 김용훈, 조범준
조선대학교 컴퓨터공학과
e-mail: jmpark@ai.chosun.ac.kr

Matching Algorithms for processing Binarized Fingerprint Image efficiently

Jong-Min Park, Yong-Hun Kim, Beom-Joon Cho
Dept. of Computer Engineering, Chosun University

요 약

본 논문에서는 이진화된 지문 이미지를 효율적으로 처리하기 위한 자료 구조인 Union and Division을 이용하여 이진화와 잡음 제거, 그리고 특이점 추출 단계로 이루어진다. Union and Division을 사용하는 경우에, 용선을 추적 가능하다는 성질을 이용하여 동일 용선 상에 있는 두 개의 특이점을 잇는 선분을 기준점으로 하는 정합 알고리즘을 제안한다.

1. 서론

생체인식 시스템은 생체정보가 갖는 특성에 따라 인식율과 오인식율(FAR: False Acceptance Rate, FRR: False Rejection Rate), 효율성에 영향을 받는다. 이러한 특성은 보편성과 유일성, 내구성, 수집성, 수용성, 효율성, 속이기 용이성에 평가되어지며, 지문은 다른 생체정보와 비교하여 특별한 단점이 없다는 장점을 가지고 있는 것으로 알려져 있다[1]. 지문은 융선(ridge)과 골(valley)로 이루어져 있다. 육안으로 는 융기 되어 있는 부분이 융 선에 해당하고 침몰되어 있는 부분이 골에 해당한다. 그리고 지문에는 특이점으로 불리는 끝점(endpoint)과 분기점(bifurcation), 다리(bridge) 등이 있으며, 특징점으로 불리는 삼각주와 코어(core) 등이 있다[2]. 지문 인식 시스템은 지문인식 장치(fingerprint acquisition device)를 통하여 아날로그(analog) 지문으로부터 변환된 디지털(digital) 지문에서 특이점을 추출하고 추출한 특이점들에 대한 정보를 데이터베이스에 등록하는 오프라인(off-line) 처리와 사용자가 시스템 접근 시에 입력한 지문으로부터 특이점을 추출하여 데이터베이스에 저장되어 있는 특이점들과 정합(matching)하여 시스템 접근 여부를 결정하는 온라인(on-line) 처리로 이루어져 있다[3]. 지문은 아날로그 지문에서 디지털 지문으로 변환되는 과정에서 다

양한 요소들로부터 영향을 받는다. 이러한 요소들에는 습기와 상처 같은 지문의 상태, 온도와 습도 같은 자연 환경, 지문 입력 시의 누르는 압력과 방향 같은 입력 상태, 먼지와 사용 기간에 따른 지문인식 장치의 상태가 있다[4]. 이러한 요소들은 지문인식 장치에 입력되는 아날로그 지문을 항상 다르게 하고, 결과적으로 디지털 지문 이미지에 잡음(noise)이 생기게 하여 높은 인식율과 낮은 오인식율, 고 효율성을 보장하는 지문인식 시스템 구현을 어렵게 한다. 따라서 잡음이 있는 디지털 지문 이미지로부터 정확하고 효율적으로 특이점을 추출하고 정합 하는 알고리즘 개발은 매우 중요한 의미를 갖는다. 본 논문에서는 이진화된 디지털 지문 이미지를 효율적으로 처리하기 위한 자료 구조인 Union and Division을 이용한다. Union and Division은 기존 지문인식 시스템에서 이진화된 디지털 지문 처리 단위인 픽셀(pixel)을 픽셀들의 집합인 선분(line segment)으로 처리하도록 함으로서 처리속도를 향상시키고, 특이점 추출과 정합 알고리즘 설계에 영향을 미친다.

2 관련 연구

디지털 지문 이미지에 있는 픽셀을 정점으로 대응시키고 대응된 두 정점이 일치하면 인접(adjacent)하도록 하여 디지털 지문 이미지를 그래프(graph)로 표현

한 후, 정점들의 인접 관계를 이용하여 특이점을 추출한다. 그러나 아직까지 디지털 지문 이미지를 효율적으로 처리하기 위한 자료구조는 제안되지 않고 있다. 특이점 추출 알고리즘은 일반적으로 이진화와 평활화, 세션화 그리고 특이점 추출과 오류 특이점 제거 단계로 이루어져 있다[4]. 이진화 단계는 8-비트(gray scale)로 표현된 각 픽셀의 값을 0과 1 중 하나의 값으로 사상한다. 현재까지 중심 픽셀 값과 주변 픽셀 값들의 평균을 비교하여 이진화하는 방법을 포함한 다양한 이진화 방법들이 제안되었다[1]. 평활화는 이진화된 디지털 지문 이미지의 용선을 부드럽게 하는 단계이다. 평활화 단계는 생략되어지는 경우가 있으며, 모폴로지(morphology) 기법들이 제안되었다[5]. 세션화는 굵은 실선으로 표현된 용선을 가는 실선으로 표현하는 단계이며, 윤곽선을 검출하는 방법과 골격선을 검출하는 방법들이 제안되었다[6]. 세션화 단계에서는 실선을 추적 가능하도록 하기 위하여 실선의 방향에 대한 정보도 저장한다. 특이점 추출은 세션화 결과인 가는 실선을 추적하여 특이점을 추출하는 단계이다. 끝점과 분기점이 아닌 특이점들은 끝점과 분기점으로 표현 가능하고, 특이점이 100~150개가 끝점과 분기점이므로 대부분의 지문 인식 시스템은 끝점과 분기점을 추출한다[3]. 오류 특이점 제거는 특이점 추출 단계에서 추출된 특이점들 중에서 특이점이 아닌 것으로 판단되는 것들을 제거한다. 현재까지 한 특이점과 주변 특이점들의 거리가 너무 가까운 경우에 오류 특이점으로 간주하여 제거하는 방법을 포함한 몇 가지 오류 특이점 제거 방법들이 제안되었다[3,5].

이진화와 평활화, 세션화 그리고 특이점 추출과 오류 특이점 제거 단계로 이루어진 일반적인 특이점 추출 단계를 거치지 않고, 디지털 지문 이미지로부터 바로 특이점을 추출하는 방법들도 제안되었다. 하지만 이러한 방법들은 다단계를 거쳐 특이점을 추출하는 방법보다 정확도가 떨어지는 것으로 평가되고 있다[4]. 추출된 특이점들에 대하여 위치와 특이점 종류, 방향 정보가 데이터베이스에 저장되거나 정합에 이용된다. 정합 알고리즘에서는 데이터베이스에 저장되어 있는 특이점들과 시스템 접근 시 사용자가 입력한 지문으로부터 추출한 특이점들을 비교하여 일치 여부를 결정한다[4].

초기의 정합 알고리즘은 데이터베이스에 저장되어 있는 특징점과 사용자가 입력한 지문으로부터 추출한 특징점을 일치시킨 후, 시스템 접근 시 사용자가

입력한 지문으로부터 추출된 특이점들을 하나의 집합으로 간주하여 이동(movement)과 스케일링(scaling), 회전(rotation)과 같은 기하학적 연산을 하여 데이터베이스에 저장되어 있는 특이점과 시스템 접근 시 입력된 지문으로부터 추출된 특이점들의 일치하는 개수를 검사하는 방법이 주로 사용되었다. 이러한 방법은 특징점이 없는 사용자를 처리하기 어렵다는 단점이 있지만, 기준점으로 사용할 수 있는 삼각주와 같은 특징점이 있는 경우에는 정합이 정확하면서도 효율적으로 이루어지는 것으로 알려져 있다. 특징점이 없는 사용자들을 처리하기 위하여 특이점들의 상호 관계를 스페닝 트리(spanning tree)와 삼각화(triangulation)를 통한 그래프로 표현하여 정합하는 알고리즘들이 제안되었으며, 푸리에 변환(Fourier transform)을 포함한 변환 기법을 이용한 정합 방법들도 제안되었다[4].

3정합알고리즘

S1: 데이터베이스에 저장되어 있는 특이점들의 집합을 나타낸다.

S2: 사용자가 시스템 접근 시 입력한 지문으로부터 추출된 특이점들의 집합을 나타낸다.

Image(S): 정합 알고리즘은 S1과 S2를 이미지(image)로 간주하고, S1에 기하학적 연산을 행한다. S1 또는 S2인 S에 대하여, Image(S)는 S를 이미지로 간주한 것을 나타낸다.

S1(x1, x2): S1의 동일 용선 줄기 상에 있는 어떤 두 특이점 x1과 x2를 잇는 선분을 나타낸다.

S2(y1, y2): S2의 동일 용선 줄기 상에 있는 어떤 두 특이점 y1과 y2를 잇는 선분을 나타낸다.

d1(S1(x1, x2)): S1(x1, x2)의 길이를 나타낸다.

d2(S2(y1, y2)): S2(y1, y2)의 길이를 나타낸다.

α : 동일 용선 상에 있는 두 특이점을 결정하는 최대 횟수이다.

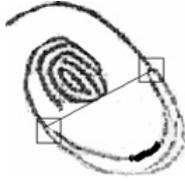
β : 사용자에게 시스템 접근을 허락하는 경우에, S2에 있는 특이점들 중에서 S1에 있는 특이점들과 일치하는 비율에 대한 역치(threshold) 값이다.

ϵ : S1의 동일 용선 줄기 상에 있는 두 특이점 x1과 x2에 대한 S1(x1, x2)과 S2의 동일 용선 줄기 상에 있는 두 특이점 y1과 y2에 대한 S2(y1, y2)를 일치시키기 위하여 S2에서 y1과 y2를 결정하는 횟수이다.

δ : 정합 알고리즘은 S1의 동일 용선 줄기 상에 있는 두 특이점 x1과 x2에 대한 S1(x1, x2)과 S2의 동일 용선 줄기 상에 있는 두 특이점 y1과 y2를 결정하는

후 $S1(x1, x2)$ 과 $S2(y1, y2)$ 를 일치시킨다. 이때, $d1(S1(x1, x2)) - d2(S2(y1, y2)) < \delta$ 이어야 한다.

제안하는 정합 알고리즘은 동일 용선 줄기 상에 있는 두 개의 특이점을 잇는 선분을 기준점으로 사용한다.



<그림 1> 동일 용선 줄기 상에 있는 두 개의 특이점을 이은 선분

Algorithm Matching

Input: S1과 S2.

Output: 시스템 접근의 허락을 의미하는 Accept나 시스템 접근 거부를 의미하는 Reject.

Begin

단계 1: $I=0, Bool=False$.

단계 2: **while** $I < a$ 이고 $Bool=False$ **do**

단계 2.1: S1의 동일 용선 줄기 상에 있는 두 개의 특이점을 결정한다.

단계 2.2: $I=I+1$.

단계 2.3: $J=0$.

단계 2.3: **while** $J < \epsilon$ 이고 $Bool=False$ **do**

단계 2.3.1: 단계 2.1에서 결정된 두 특이점을 $x1$ 과 $x2$ 라 하자. S2에서 $d1(S1(x1, x2)) - d2(S2(y1, y2)) < \delta$ 인 $y1$ 과 $y2$ 를 결정한다.

단계 2.3.2: $J=J+1$.

단계 2.3.3: 다음 조건 중 하나를 만족하는 경우에 한하여 단계 2.2.3을 수행한다.

① $x1$ 과 $y1$ 의 특이점 종류가 동일하고 $x2$ 와 $y2$ 의 특이점 종류가 동일하다.

② $x1$ 과 $y2$ 의 특이점 종류가 동일하고 $x2$ 와 $y1$ 의 특이점 종류가 동일하다.

단계 2.3.4:

Movement(Rotation(Scaling(Image(S2))))를 하여 $S1(x1, x2)$ 와 $S2(y1, y2)$ 를 일치시킨다.

단계 2.3.5: S1에 있는 특이점들과 S2에 있는 특이점들의 일치하는 비율이 β 이상이면 $Bool=True$.

단계 3: $Bool=True$ 이면 Accept를 출력하고, $I \geq a$ 이면 Reject를 출력한다.

End

알고리즘 Matching에서

Movement(Rotation(Scaling(Image(S2))))는 Image(S2)에 대하여, 스케일링과 회전, 이동을 차례로 행하는 것을 의미한다.

4 실험

실험은 펜티엄 PC에 Windows 2000 운영체제 환경에서 이루어졌다. 실험을 위한 모든 프로그램은 C++로 구현되었으며, NIST의 SDB에 있는 27,000 여종의 디지털 지문 이미지를 처리하였다.

Union and Division은 이진화와 평활화, 세션화, 특이점 추출, 오류 특이점 제거 단계로 이루어진 특이점 추출 알고리즘에서 동일한 평활화와 세션화, 특이점 추출 방법을 사용하더라도 효율적으로 특이점을 추출하도록 한다.

이진화 방법과 평활화 방법, 세션화 방법을 Union and Division을 사용하였을 때는 사용하지 않았을 때의 1/3에 대한 평균 수행 시간을 나타내었다. 제안하는 정합 알고리즘은 동일 용선 줄기 상에 있는 두 특이점을 잇는 선분을 기준점으로 사용한다. 따라서 얼마나 많은 지문에서 동일 용선 상에 두 특이점이 존재하는가를 검사하는 것은 제안한 정합 알고리즘에서 매우 중요하다. 특이점은 적어도 하나의 동일 용선 줄기 상에 적어도 두 개의 특이점이 존재하고 있었다.

5 결론

Union and Division은 각 열에 있는 린을 하나의 선분으로 간주하여 처리하도록 함으로서 디지털 지문 이미지를 처리하는데 소요되는 시간을 줄이게 된다는 점과 Union and Division이 용선을 추적 가능하게 한다는 점을 이용하여 동일한 용선 줄기 상에 있는 두 특이점을 잇는 선분을 기준점으로 하는 정합 알고리즘이 기존 정합 알고리즘과 비교하여 인식율이 높고 오인식율이 낮으며 효율성 면에서도 우수함을 실험 결과를 통하여 알았다.

참고문헌

- [1] The Science of Fingerprints: Classification and Uses United States Department of justice, Federal Bureau of Investigation, Washington, rev. 12-84, 1988.
- [2] A. Shimizu, M. Hase. Tmas. Inst. Electronic Comm. Engineers Japan, Part D, J67D(5), pp.627
- [3] Andrew K. Hrechak, James A. Mchugh,

"Automated Fingerprint Recognition using structural matching", Pattern Recognition, vol. 23, pp. 893-904, 1990.

- [4] L. Coetzee and E. C. Botha, "Fingerprint Recognition in Low Quality Images," Pattern Recognition, vol. 26, no. 10, pp. 1441-1460, 1993.
- [5] A. Farina, Z. M. Kovacs-vajna, Alverto Leone, "Fingerprint minutiae extraction from skeletonized binary images", Pattern Recognition, vol. 32, no. 4, pp. 877-889, 1999.