

실시간 기하학적 해싱을 위한 하드웨어 구현

김민⁺*, 문대성⁺, 반성범⁺, 정용화⁺, 김기철^{*}
⁺한국전자통신연구원 생체인식기술연구팀
^{*}서울시립대학교 전자전기컴퓨터공학부
e-mail : minkim@ece.uos.ac.kr

A Hardware Implementation for Real-Time GH

Min Kim⁺*, Daesung Moon⁺, Sung Bum Pan⁺, Yongwha Chung⁺, Kichul Kim^{*}
⁺Biometrics Technology Research Team, ETRI
^{*}Dept. of Electrical & Computer Engineering, University of Seoul

요 약

지문 인식(identification)은 사용자 등록 과정과 사용자 인식 과정으로 수행된다. 사용자 등록 과정은 데이터베이스에 사용자의 지문 정보를 저장하는 과정이고 인식 과정은 입력된 사용자의 지문 정보에 대하여 유사한 후보자 목록을 작성하는 과정이다. 지문 인식은 데이터베이스에 저장된 사용자 지문 정보를 전체적으로 검색할 경우 오랜 수행 시간이 요구되므로 이러한 문제점을 해결하기 위하여, 본 논문에서는 기하학적 해싱(geometric hashing) 알고리즘을 이용하여 지문 인식을 수행하며 지문 인식을 병렬적으로 처리하기 위해 데이터베이스의 구조를 k 비트-레벨 표현으로 나타내었다. 또한, 실시간으로 지문 인식을 처리하기 위해 지문 인식을 위한 전용 하드웨어를 설계하였다. 설계된 하드웨어는 등록자 수의 증가에 따른 우수한 확장성을 보이며 병렬적으로 처리되므로 고속 실시간 처리가 가능하다.

1. 서론

생체인식은 사용자가 소지하거나 암기할 필요가 없는 편리성 및 반드시 본인이 있어야 인식되는 높은 신뢰성 때문에 사용자 패스워드 또는 PIN(Personal Identification Number) 그리고 신용카드 등을 보완할 수 있는 새로운 보안 수단이다[1-3].

생체인식이란 개인의 생체적 혹은 행동적인 특성을 이용하여 개인을 인식하는 기술이다. 생체 특징을 이용한 기술로는 지문, 얼굴, 홍채, 망막 및 정맥인식 등이 있으며, 행동 특징을 이용한 기술로는 서명 및 음성 인식 등이 대표적이다. 생체인식 기술은 수년 간 도어록 등 출입통제 시스템과 관련된 물리적 보안에 적용되어 왔지만 앞으로는 인터넷 뱅킹, 전자상 거래, 원격 교육 그리고 정보 보호 등 컴퓨터 보안에도 적용 비율이 높아질 것이다. 이에 따라 네트워크를 지원하는 생체인식 기술을 이용한 개인 인증 방법은 공개키기반구조(PKI: Public Key Infrastructure)의 전자서명인증기술과 함께 중요한 개인인증 수단으로 자리잡게 될 전망이다.

생체 정보들 중에서 지문은 가장 오래된 연구를 통

해 신뢰성이 입증되었을 뿐만 아니라 사용의 편리함, 구현 장비의 저가, 소형화에 가장 부합하는 생체 특징으로 평가 받고 있다. 이와 같은 이유 때문에 지문은 생체인식에서 가장 널리 사용되고 있는 생체 정보이다.

지문의 이용 방법으로는 1:1(verification) 지문 인증과 1:N(identification, 1:many) 지문 인식이 있다[4]. 1:1 지문 인증은 본인 확인 과정으로 입력된 사용자 지문이 저장된 지문과 동일한 지문인지를 판단한다. 1:N 지문 인식은 데이터베이스에 N 명의 사용자 지문 정보를 저장한 후, 데이터베이스에서 검색과정을 통하여 입력된 사용자 지문 정보와 유사한 후보자 목록을 결과로 보여준다. 이와 같은 1:1 혹은 1:N 지문 인식은 특징 추출(feature extraction) 단계와 1:1 혹은 1:N 지문 정합(fingerprint matching) 단계로 구성된다.

본 논문에서 고려하는 1:N 지문 인식의 경우, 데이터베이스에 저장된 지문 데이터를 전체적으로 검색하는 것은 등록자의 수 만큼 오랜 수행 시간을 요구한다. 이와 같은 문제점을 해결하기 위하여 컴퓨터 비전에서 잘 알려진 기하학적 해싱(geometric hashing) 알고

리즘을 이용한다[5]. 기하학적 해싱 알고리즘은 등록자 지문의 기하학적인 모양에 따라, 등록자를 분류하여 데이터베이스 저장하므로 데이터베이스의 검색 시간을 줄일 수 있어 빠른 수행 속도를 보인다.

기하학적 해싱 알고리즘을 이용하여 데이터베이스의 검색 시간을 줄였지만, 데이터베이스에 저장되는 등록자의 수가 증가할수록 실시간으로 지문 인식을 수행하기에는 어려움이 따른다. 결국, 실시간으로 지문 인식을 수행하기 위해서는 지문 인식을 병렬적으로 처리해야 하고 전용 하드웨어로 설계해야 한다. 지문 인식을 병렬적으로 처리하고 등록자의 수의 증가에 따른 전용 하드웨어의 확장성을 고려하여, 데이터베이스의 구조를 k 비트-레벨 표현(k bits-level representation)으로 나타낸다.

본 논문의 구성은 다음과 같다. 본 장의 서론에 이어 제 2 장에서는 기하학적 해싱 알고리즘에 관하여 설명하고 제 3 장에서는 기하학적 해싱 기반 지문 인식 시스템의 구조를 보인다. 제 4 장에서는 지문 정합을 위하여 사용된 알고리즘을 설명한다. 제 5 장에서는 제 4 장의 알고리즘을 바탕으로 설계된 하드웨어 구조를 보인다. 마지막으로 제 6 장에서 본 논문의 결론을 내린다.

2. 기하학적 해싱 알고리즘

컴퓨터 비전에서 객체는 객체의 특징들(features)의 집합으로 나타내어 지는데, 객체의 특징들에는 점(point), 선(line), 코너(corner), 에지(edge) 등이 있다.

객체 인식(object recognition)이란 입력된 영상 장면(image scene)에서 관심 있는 객체를 인식하는 것을 말한다. 즉, 데이터베이스에 관심 있는 객체인 모델(model)의 특징을 저장한 후, 데이터베이스의 모델을 검색함으로써 입력 영상 장면에서 관심 있는 객체를 인식하는 것이다. 이와 같은 객체 인식 시스템을 모델 기반 인식 시스템(model-based recognition systems)이라 한다.

대부분의 모델 기반 인식 시스템은 입력 영상 객체의 특징과 데이터베이스에 저장된 하나의 모델의 특징이 같다는 가정(hypothesis)에 기초하여, 가정을 검증(verify)하게 된다. 이와 같은 과정은 데이터베이스에 저장된 모든 모델에 대하여 전체적으로 수행된다. 따라서, 처리 시간은 데이터베이스에 저장된 모델의 수에 비례하게 되고 모델의 수가 증가할수록 처리 시간은 오래 걸리게 된다.

이와 같은 문제점을 해결하기 위하여 Lamdan 과 Wolfson 은 색인(index)에 기초한 기하학적 해싱 알고리즘을 객체 인식에 도입하였다[5]. 기하학적 해싱 알고리즘은 객체를 나타내는 기하학적인 모양은 변화하지 않는다는 사항을 고려하여 색인 함수를 생성하고 색인 함수를 이용하여 데이터베이스를 효율적으로 검색하는 알고리즘이다. 모델 기반 인식 시스템에 기하학적 해싱 알고리즘을 이용하면, 모델의 개수가 증가하여도 색인 함수의 성능이 좋다면 처리 시간은 거의 영향을 받지 않는다.

기하학적 해싱 알고리즘은 두 단계로 나누어지는데,

전처리(preprocessing) 단계와 인식(recognition) 단계이다. 전처리 단계는 오프-라인으로 행해지며, 데이터베이스에 해당하는 해시 테이블(hash table)에 색인 과정을 통하여 모델의 특징을 저장하는 단계이다. 인식 단계는 색인을 이용하여 해시 테이블의 모델 정보를 검색하여 입력 객체와 유사한 후보 모델의 순위를 결과로 보여주는 단계이다.

3. 기하학적 해싱 기반 지문 인식 시스템

본 논문에서는 생체 정보 중에서 실생활에서 가장 많이 응용되고 있는 지문 정보를 이용하여 사용자를 인식한다. 지문이란 인간의 손바닥에 존재하는 땀구멍이 융기한 선으로 형성된 문형을 말하는 것으로, 융기되어 나타나는 융선(ridges)과 두 융선 사이의 패인 골(valleys)로 나타내어진다. 지문 인식의 방법으로는 영상을 기반으로 하는 방법[6]과 영상 내에 존재하는 특징점을 이용하는 특징점 기반[7]으로 나눌 수 있는데 지문 인식에 사용하는 특징점은 특별히 minutia 라고 부른다. 본 논문에서는 특징점 기반 사용자 인식 시스템을 구현하였다. Minutia 는 분기점(bifurcation)과 단점(ending)의 두 가지 타입으로 나뉘어 지는데 분기점이란 한 가닥의 융선이 두 가닥의 융선으로 갈라지는 곳을 말하고 단점이란 융선의 흐름이 끊기는 곳을 말한다.

기하학적 해싱 기반 지문 인식 시스템은 그림 1 과 같이 특징 추출(feature extraction)과 1:N 지문 정합(fingerprint matching)의 두 단계를 거치게 된다. 특징 추출 단계에서는 지문영상에서 특징점 정보들을 추출

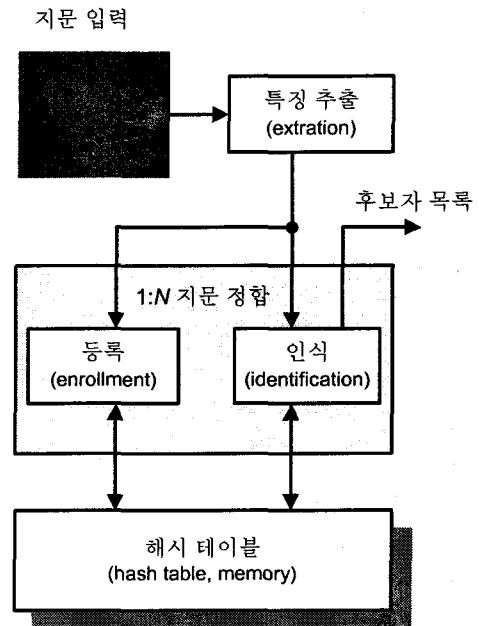


그림 1. 지문 인식 시스템.

하는 단계이다. 지문 정합은 사용자 등록(enrollment) 과정과 사용자 인식(identification) 과정으로 구성된다. 사용자 등록 과정은 지문의 기하학적인 모양에 따라서 데이터베이스에 지문 정보를 저장하는 과정을 말하고, 사용자 인식 과정은 데이터베이스에 저장된 지문 정보와 유사한 등록된 사용자 목록을 출력하는 과정이다. 본 논문에서는 그림 1 과 같은 지문 인식 시스템에서 지문 정합에 대하여 설명하기로 한다.

4. 기하학적 해싱 기반 지문 정합 알고리즘

기하학적 해싱 알고리즘을 이용하여 지문 정합을 처리하기 위해서는 세 가지 사항이 고려되어야 한다. 각각에 대하여 살펴보면 다음과 같다.

첫 번째 사항은 해시 테이블의 색인(index, key)을 생성하는 과정이다. 해시 테이블의 색인을 생성하기 위해서는 참조 프레임(reference frame)을 결정해야 한다[8]. 특징점 추출 단계에서 입력된 지문 영상에 대하여 특징점이 추출될 때, 특징점의 좌표, 각도 등 특징점의 정보는 입력된 영상의 잡음에 의하여 변화될 수 있다. 하지만, 사용자 지문의 특징점이 형성하는 기하학적인 모양은 크게 변하지 않으므로 본 논문에서는 특징점의 좌표와 각도를 이용하여 참조 프레임을 결정한다. 이와 같이 결정된 참조 프레임을 기반으로 하여 해시 테이블의 색인을 생성한다. 색인 생성 과정은 지문 정합의 등록 및 인식 과정에서 사용된다. 즉, 등록 과정에서는 해시 테이블에 사용자의 정보를 저장하기 위하여 사용되며 인식 과정에서는 해시 테이블에 저장된 데이터를 읽기 위하여 사용된다. 색인 생성 과정은 다음과 같다.

- 1) 기준점 선정: 입력된 지문의 특징점 중에서 하나의 특징점을 기준점으로 선정한다.
- 2) 회전: 선정된 특징점의 각도만큼 모든 특징점을 회전시킨다.
- 3) 축소: 회전에 의하여 변환된 특징점의 x, y 좌표에 대하여 2 로 나누어 줌으로써 변환된 좌표의 크기를 축소 시킨다.
- 4) 이동: 선정된 기준점을 해시 테이블의 중심 좌표의 주소로 이동시킨다.
- 5) 색인 생성: 선정된 기준점을 제외한 다른 특징점의 좌표가 해시 테이블의 색인이 된다.

두 번째 사항은 해시 테이블에 저장되는 사용자의 지문 정보이다. 해시 테이블에 저장되는 지문 정보에 따라서 해시 테이블의 크기와 지문 정합의 인식률 사이에 trade-off 가 존재한다. 즉, 지문 정합의 인식률을 높이기 위하여 많은 지문 정보를 저장하면 해시 테이블의 크기는 커질 것이며, 해시 테이블의 크기를 줄이기 위하여 적은 지문 정보를 저장하면 지문 정합의 인식률은 낮아질 것이다. 이와 같이 해시 테이블에 저장되는 지문 정보를 결정하는 사항은 중요하다. 본 논문에서는 해시 테이블의 크기와 지문 정합의 인식률을 고려하여, 사용자의 지문 정보에 해당하는 사용자 및 기준점의 ID(Identification)와 특징점의 타입(type)

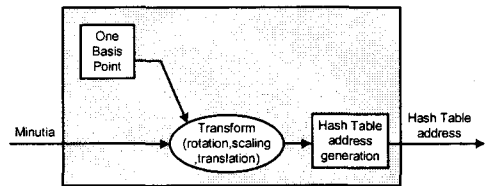
및 각도 값을 해시 테이블에 효과적인 방법으로 저장한다.

마지막 사항은 해시 테이블의 구조이다. 지문 정합을 실시간을 처리하기 위해서는 지문 정합을 병렬로 처리해야 한다. 이를 위하여 Chung 은 데이터베이스의 구조를 한 비트-레벨 표현(a bit-level representation)으로 나타내었다[9]. 데이터베이스에 저장되는 정보를 각각의 PN(Processing Node)마다 분리하여 저장하였지만, PN 간의 데이터베이스 충돌이 없으므로 병렬처리가 가능하다. 본 논문에서는 한 비트-레벨 표현 방법의 수정 방법인 k 비트-레벨 표현 방식으로 해시 테이블의 구조를 나타낸다.

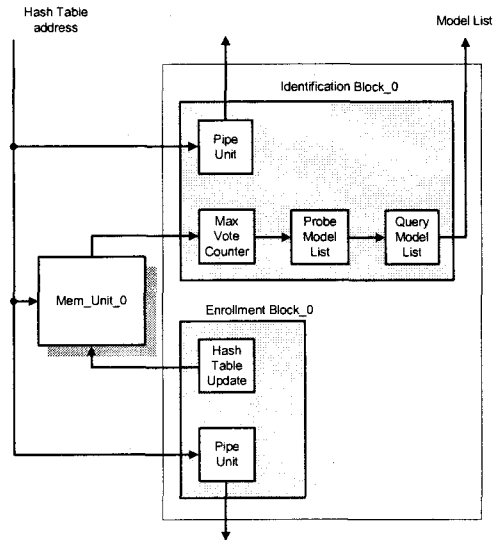
5. 기하학적 해싱 기반 지문 정합 하드웨어 구조

본 논문에서는 지문 정합 과정을 실시간으로 처리하기 위하여, 지문 정합을 전용하드웨어로 설계한다.

그림 2 는 지문 정합의 구조를 나타낸다. 지문 정합은 전처리 블록(preprocessing block)과 후처리 블록



(a) 전처리 블록 (preprocessing block)



(b) 후처리 블록 0 (postprocessing block 0)

그림 2. 지문 정합의 구조도.

(postprocessing block)으로 이루어져 있다. 전처리 블록은 제어 신호에 따라서 등록 과정과 인식 과정을 수행하게 되고 해시 테이블의 주소를 생성하게 된다. 후처리 블록은 등록 블록(enrollment block)과 인식 블록(identification block)으로 나누어진다. 등록 블록은 사용자의 지문 정보를 해시 테이블에 저장하는 등록 과정을 수행하고 인식 블록은 사용자의 지문과 유사한 후보자 목록을 출력한다. 메모리 유닛(mem_unit)에는 해시 테이블 정보가 저장되게 된다. 파이프 유닛(pipe unit)은 해시 테이블의 주소뿐만 아니라 제어 신호까지 레지스터를 통한 파이프라인 형태로 이전 후처리 블록에서 다음 후처리 블록으로 공급하는 역할을 수행한다.

후처리 블록은 확장성과 정합 과정의 병렬처리를 고려하여 설계하였다. 해시 테이블에 등록되는 사용자의 수가 증가할수록 파이프 유닛을 이용하여 후처리 블록을 병렬적으로 연결함으로써 해시 테이블의 데이터 충돌 없이 병렬적으로 정합 과정을 처리할 수 있다. 각각의 후처리 블록에서 처리된 결과 값은 다음 후처리 블록으로 전달되어 최종 결과 값이 출력된다. 그림 3은 m 개의 후처리 블록을 병렬적으로 연결한 지문 정합의 확장 구조를 나타낸다. 점선은 등록 과정에서의 데이터 흐름을 나타낸다.

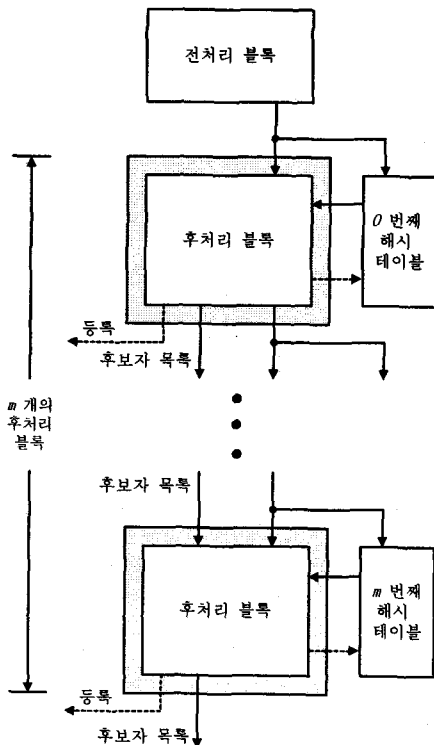


그림 3. 지문 정합의 확장 구조도.

6. 결론

본 논문에서는 지문 인식을 실시간으로 처리하기 위하여, 기하학적 해싱 알고리즘을 이용하여 지문 정합 과정을 수행하였다. 지문 인식은 데이터베이스에 사용자의 지문 정보를 등록한 후, 입력된 사용자 지문에 따라서 유사한 후보 목록을 작성하기 위하여 데이터베이스에 저장된 지문 정보를 전체적으로 검색하게 된다. 결국, 데이터베이스에 등록된 사용자의 수가 많을수록 지문 인식 과정을 수행하는데 오랜 시간이 요구된다. 이와 같은 문제점을 해결하기 위하여 기하학적 해싱 알고리즘을 이용하였다. 따라서, 본 논문에서는 사용자 지문의 기하학적인 모양을 고려하여 지문 정보를 데이터베이스에 저장하고 검색하였다. 또한, 기하학적 해싱 알고리즘을 기반으로 한 지문 정합 과정을 병렬적으로 처리하기 위하여 데이터베이스의 구조는 k 비트 레벨 표현으로 나타내었다. 마지막으로 지문 인식을 실시간으로 처리하기 위하여 지문 정합을 전용 하드웨어로 설계하였다. 설계된 하드웨어는 등록자 수의 증가에 따른 우수한 확장성을 보이며 병렬적으로 처리되므로 고속의 동작이 가능하다.

참고문헌

- [1] A. Jain, R. Bole, and S. Panakanti, *Biometrics: Personal Identification in Networked Society*, Kluwer Academic Publishers, 1999.
- [2] S. Pankanti, R. Bolle, and A. Jain, "Biometrics: The Future of Identification," *IEEE Computer*, vol. 33, pp. 46-49, 2000.
- [3] Biometric Consortium, *Proc. of the Biometric Consortium 2000 Conference*, 2000.
- [4] 반성범, 정용화, 조현숙, "IC 카드를 이용한 생체인식 기술 개발 동향," *전자통신동향분석*, 제 16 권 6 호, pp. 49-55, 2001년 12월.
- [5] Y. Lamdan, H. Wolfson, "Geometric Hashing: A General and Efficient Model-Based Recognition Scheme," *Proc. Int'l Conf. Computer Vision*, pp. 238-249, 1998.
- [6] A. Jain, L. Hong, "Filterbank-Based Fingerprint Matching," *IEEE Trans. on Image Processing*, vol. 9, pp. 846-859, 2000.
- [7] N. Ratha, A. Jain, "A Real-Time Matching System for Large Fingerprint Database," *IEEE Trans on PAMI*, vol. 18, pp. 799-813, 1996.
- [8] H. Wolfson, I. Rigoutsos, "Geometric Hashing an Overview," *IEEE Computational Science and Engineering*, vol. 4, pp. 10-21, Oct.-Dec. 1997.
- [9] Y. Chung, S. Choi, V. Prasanna, "Parallel Object Recognition on an FPGA-based Configurable Computing Platform," *Proc. of the 4th IEEE International Workshop on CAMP*, pp 143-152, Oct. 1997.