

특징점과 필터뱅크에 기반한 적응적 혼합형 지문정합 방법

(Adaptive Hybrid Fingerprint Matching Method Based on Minutiae and Filterbank)

정석재[†] 박상현^{**} 문성림^{***} 김동윤^{****}
 (Seok Jae Jeong) (Sang Hyun Park) (Sung Rim Moon) (Dong Yoon Kim)

요약 Jain 등은 한 지문 영상에 특징점 기반 지문 정합 방법과 필터뱅크 기반 지문 정합 방법을 적용해 두 정합 방법의 성능을 혼합하는 혼합형 지문정합 방법을 제안하고, 이 방법이 두 가지 각 정합 방법에 비해 높은 성능을 보인다는 것을 실험을 통해 입증하였다[1]. 그러나 이 방법은 혼합을 수행할 때 두 정합 방법을 별도로 수행한 후, 각 방법의 정합도(matching score)에 가중치를 부여해 최종 정합도를 결정하므로 두 정합 방법의 특성을 상쇄 시키는 결과를 얻게 된다. 본 논문에서는 두 가지 정합 방법을 특징값 추출 과정에서 혼합하는 방법을 제안하였다. 이 방법은 필터뱅크 기반 방법보다는 낮은 ERR(Equal eRror Rate)을 보이나 특징점 기반 방법보다 높은 ERR을 보였다. 이에 본 논문에서는 적응적인 정합도 혼합방법을 제안하여, 두 가지 방법의 특성을 살리도록 적응적으로 정합도를 선택하는 방법을 취했다. 이 방법을 이용해 Jain 등의 혼합형 방법보다 더 낮은 ERR을 얻을 수 있었다. 제안한 방법에 따라 NIST Special Database 14 지문 데이터로 실험한 결과 ERR에서 약 1%의 성능 향상을 보였다.

키워드 : 생체인증, 지문 인식, 지문 정합, 혼합형 지문 정합, 필터뱅크 기반 지문인식

Abstract Jain et al. proposed the hybrid matching method which was combined the minutia-based matching method and the filter-bank based matching method. And, their experimental results proved the hybrid matching method was more effective than each of them. However, this hybrid method cannot utilize each peculiar advantage of two methods. The reason is that it gets the matching score by simply summing up each weighted matching score after executing two methods individually. In this paper, we propose new hybrid matching method. It mixes two matching methods during the feature extraction process. This new hybrid method has lower ERR than the filter-bank based method and higher ERR than the minutia-based method. So, we propose the adaptive hybrid scoring method, which selects the matching score in order to preserve the characteristics of two matching methods. Using this method, we can get lower ERR than the hybrid matcher by Jain et al. Experimental results indicate that the proposed methods can improve the matching performance up to about 1% in ERR.

Key words : Biometric Authentication, Fingerprint Identification, Fingerprint Matching, Hybrid Fingerprint Matching, Filterbank-based Fingerprint Identification

1. 서론

다양한 종류의 생체정보들이 개인 인증의 용도로 사

용되면서 둘 이상의 생체정보를 이용한 생체 인증 시스템들이 제안되고 있다. 현재 연구되고 있는 생체정보에는 얼굴, 홍채, 지문, 손, 정맥, 음성, 걸음걸이 등이 있으며, 이들 정보 중 둘 이상의 생체정보를 이용해 개인 인증에 사용하면 인증 신뢰도를 높일 수 있다. 그러나 두 가지 이상의 서로 다른 생체정보를 이용해 개인 인증을 수행하기 위해서는 각 생체 인증 방법에서 요구하는 생체정보 획득 장치들이 필요하고, 각 생체정보들을 종류에 따라 처리하기 위한 별도의 알고리즘들과 이들을 혼합하기 위한 방법들이 지원되어야 하는 추가 비용

† 비 회 원 : 아주대학교 정보통신연구소 연구원
 maniac@ajou.ac.kr
 ** 정 회 원 : (주)알폰스테크 연구원
 diferens@nownuri.net
 *** 학생회원 : 아주대학교 정보및컴퓨터공학과
 riera@ajou.ac.kr
 **** 종신회원 : 아주대학교 정보및컴퓨터공학부 교수
 dykim@ajou.ac.kr
 논문접수 : 2004년 2월 10일
 심사완료 : 2004년 4월 6일

이 필요하며, 한 가지 생체 정보에서 두 가지 이상의 정보를 이용한 방법(한 사람으로부터 두 손가락의 지문을 사용하거나, 양쪽 눈의 홍채 정보를 사용하는 경우)을 사용한다고 해도 매 인증시마다 사용자에게 필요한 수 만큼의 입력을 요구하는 불편함을 가져온다. 이러한 추가비용과 불편함을 해소하기 위해 한 가지 지문 정보를 두 가지 다른 방법으로 표현하여 정합을 수행하고 그 결과를 혼합해 인증 성능을 향상시키는 방법을 Jain 등이 제안하였다.

Jain 등은 한 개의 지문정보에 대해 특징점 기반 방법과 필터뱅크 기반 방법으로 얻은 각각의 정합도(matching score)를 혼합해 한 가지 방법만을 사용하는 것보다 더 높은 성능을 보일 수 있는 정합도 혼합 방법을 제안하고, 실험을 통해 각각의 방법보다 혼합형 지문 정합 방법이 더 높은 성능을 보임을 입증하였다[1]. 또한 정석재 등은 필터뱅크 기반 방법에서 특징벡터를 추출할 때, 특징점이 존재하는 영역에 가중치를 부여하는 방식으로 필터뱅크 기반 방법의 성능을 향상시키는 방법을 제안하였다[2]. 그러나 Jain 등이 제안한 정합도 혼합 방법은 두 가지 방법의 결과로 얻은 각 정합도에 가중치를 부여해 더함으로써 정합 정도를 결정하기 때문에 두 시스템의 특성이 상쇄되는 결과를 가져온다. 그리고 정석재 등이 제안한 특징점의 유무에 따라 가중치를 부여해 특징벡터를 추출하는 방법은 필터뱅크 기반 방법의 단점인 중심점(core)을 찾지 못하는 경우 인증을 수행할 수 없다는 문제점을 갖는다.

본 논문에서는 특징점의 유무를 통해 가중치를 부여하는 대신 특징점마다 특징점을 포함하는 일정 영역에 가보필터(Gabor filter)를 적용해 특징벡터를 구하는 혼합형 지문정합 방법을 제안한다. 이 방법은 지문의 중심점을 기준으로 지문 이미지의 정렬을 수행하지 않고, 특징점들을 기준으로 정렬을 수행하므로 중심점을 찾지 못하는 지문에 대해서도 처리가 가능하게 된다. 그러나 이와 같이 구현한 혼합형 지문 인증 방법의 성능은 실제로 특징점 기반방법 보다 낮은 결과를 보인다. 이에 본 논문에서는 제안된 혼합형 지문정합 방법과 특징점 기반 지문정합 방법의 정합도를 적용적으로 선택해 최종 정합도를 결정하는 적응적 정합도 혼합 방법을 제안한다. 이때 정합도의 혼합 방식은 Jain 등이 제안한 방법과는 달리 두 가지 정합방법의 FAR(False Acceptance Rate)과 FRR(False Reject Rate)을 먼저 분석한 후 각 시스템의 성능을 살필 수 있도록 정합도를 선택하도록 한다. 제안한 방법에 따라 시스템을 구현하고 NIST Special Database 14에 포함된 1000쌍에 대해 Jain들이 제안한 정합도 혼합 방법과 비교하고 결과 분석을 통해 제안한 방법이 어느 정도 성능 향상을 가져

오는지 살펴보기로 한다.

본 논문의 구성은 다음과 같다. 2장에서는 Jain 등이 제안한 필터뱅크 기반 지문정합 방법에서 특징벡터 추출 방법에 대해 살펴본다. 그리고 정석재 등이 제안한 특징점의 유무에 따라 가중치를 부여해 특징벡터를 추출하는 방법과 Jain 등이 제안한 정합도 혼합 방법에 대해 설명한다. 3장에서는 본 논문에서 제안한 새로운 혼합형 지문정합 방법에 대해 설명하고, 두 가지 지문정합 방법의 결과로 얻은 정합도를 특성에 따라 혼합하는 적응적 정합도 혼합방법에 대해 설명한다. 4장에서는 제안된 방법의 성능을 평가하기 위해 사용된 지문 영상, 지문인식과정 등의 실험 조건에 대해 기술하고, 기존 방법들과 제안된 방법들에 따라 얻은 실험결과를 살펴보고 분석해 본다. 마지막으로 5장에서는 결론과 향후 연구 내용에 대해 기술한다.

2. 관련 연구

2.1 필터뱅크 기반 지문정합 방법

Jain 등은 기존 지문 정합 방법에서 가장 많이 사용되는 특징점을 사용하지 않고, 홍채 인식에서 사용되는 가보 필터(Gabor filter)를 이용한 필터뱅크 기반 지문 정합 방법을 제안 하였다[3]. 식 (1)은 Jain 등이 지문 영상에 적용한 등방성 가보필터(Even-symmetric Gabor Filter)를 보여준다[4].

$$G(x, y; f, \theta) = \exp\left[-\frac{1}{2}\left[\frac{x'^2}{\delta_x^2} + \frac{y'^2}{\delta_y^2}\right]\right] \cos(2\pi fx') \quad (1)$$

$$x' = x \sin \theta + y \cos \theta$$

$$y' = x \cos \theta - y \sin \theta$$

필터뱅크 기반방법에서 8가지 방향의 가보필터를 적용해 영역별로 특징값들을 계산하고 이를 Finger Code 라고 하는 특징벡터로 구성해 지문의 특성을 나타낸다. 그림 1은 이와 같은 방법으로 얻은 특징벡터를 보여준다.

이때 지문의 중심점(core)을 구하는 방법은 [4]에서

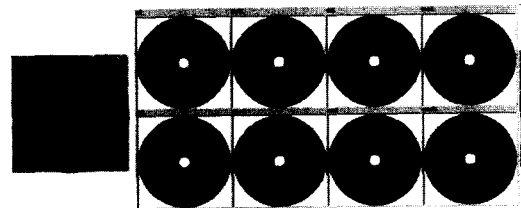


그림 1 중심점을 기준으로 추출한 영역에 대해 8가지 방향의 가보필터링을 수행한 후 얻은 특징 벡터 (실제 중심원의 중심은 지문의 중심점에서 20픽셀 아래 점을 이용한다.)

제한한 방법을 사용하고, Finger Code의 회전 오차를 줄이기 위해 11.25도 회전한 영상에 대해 같은 방법으로 특징벡터를 추가로 추출한다.

2.2 혼합형 지문 인식 방법

정식제 등은 필터뱅크 기반방법이 융선들의 방향에 많은 영향을 받기 때문에 서로 다른 지문이라 하더라도 같은 종류의 지문인 경우 높은 정합도를 보일 수 있다는 문제점을 지적하였다. 그들은 이러한 문제점을 극복하기 위해 특징벡터를 추출할 때, 해당 영역에 특징점이 존재하는가 여부에 따라 식 (2)와 같이 가중치를 부여하였다[2].

$$V_{ij} = \begin{cases} \frac{1}{n_i} (\sum_n |F_{ij}(x, y) - P_{ij}|) \times \alpha, & \text{if } 1 \leq N_{ij} \leq M_e \text{ or } 1 \leq N_{ij} \leq M_b \\ \frac{1}{n_i} (\sum_n |F_{ij}(x, y) - P_{ij}|) \times (1 - \alpha), & \text{else} \end{cases} \quad (2)$$

정식제 등이 제안한 방법은 기존 필터뱅크 기반 방법에 비해 높은 성능향상을 가져온다. 그리고 이들은 Finger Code의 회전 오차 고려에 소요되는 시간을 제거하기 위한 빠른 지문정렬 방법을 제안하였다. 그러나 이 방법들은 기존 필터뱅크 기반방법과 마찬가지로 지문의 중심점을 찾는 것에 의존하므로 지문의 중심점이 존재하지 않거나 획득한 지문영상의 품질이 좋지 않아 중심점을 찾지 못하는 영상에 대해서는 정합에 실패할 수 있다는 문제점을 갖는다.

이러한 문제점의 해결방법으로 Jain 등은 지문에서 추출한 특징점을 기반으로 지문영상에 대해 먼저 정렬을 수행하고 Finger Code를 추출하는 방법을 사용하였다[1]. 이 방법으로 중심점이 존재하지 않을 경우에도 필터뱅크 기반방법이 적용 가능하게 된다. 이때, 특징점을 이용해 지문영상을 정렬하는 방법은 Hough 변환을 이용하는 방법[5]이나 특징점 중 특히 끝점에 연결된 융선들의 정합을 통해 지문을 정렬하는 방법[6]을 사용할 수 있다.

또한 Jain 등은 [1]에서 기존에 제안했던 필터뱅크 기반 지문정합 방법과 특징점 기반 지문정합 방법을 혼합하는 혼합형 지문정합 방법을 제안하고, 두 정합방법을 혼합하기 위해 식 (3)과 같이 정합도를 계산한다.

$$S = \alpha S_M + (1 - \alpha) S_R \quad (3)$$

이때, $\alpha \in [0, 1]$ 이고 S_M 은 특징점 기반 방법으로 얻은 정합도이고 S_R 은 필터뱅크 기반 방법으로 얻은 정합도를 의미한다. 그들은 실험에서 α 에 0.5의 값을 사용하였다. 이와 같은 혼합 방법은 기존에 한 사람으로부터 두 가지 이상의 생체 정보를 사용하는 방법과는 달리 한 가지 정보로부터 두 가지 다른 정합방법을 적용한다는 점에서 사용자에게 한 가지 정보만을 요구하고, 두

가지 정합방법의 결과를 혼합해 더 높은 성능을 보일 수 있다는 두 가지 장점을 취할 수 있었다. 그러나 이 방법은 단순히 두 가지 방법의 정합결과를 더하는 방법을 취해 두 정합 방법의 장점들을 상쇄시키는 결과를 보일 수 있다는 점과 α 값을 실험을 통해 얻어야 한다는 단점을 가지고 있다.

3. 제안된 혼합형 지문 인식 시스템

본 논문에서는 특징점 기반 지문정합 방법과 필터뱅크 기반 지문정합 방법을 특징벡터 추출과정에서 혼합하는 방법을 제안한다. 그리고 이 방법을 통해 얻은 정합도와 특징점 기반 지문정합 방법을 통해 얻은 정합도를 두 방법의 특성에 따라 혼합하는 적응적 정합도 혼합 방법을 제안한다. 본 장에서는 이 두 가지 방법에 대해 자세히 설명하기로 한다.

3.1 제안된 혼합형 지문 인식 방법

제안된 혼합형 지문 인식 방법에서는 필터뱅크 기반 지문정합 방법에서 지문의 정해진 영역에 대해 특징벡터를 추출하는 방식과 달리 각 특징점들을 중심으로 하는 사각형 영역을 얻고, 이 영역들에 대해 가보필터를 적용하고, 특징벡터를 추출한다. 그림 2는 특징점을 중심으로 얻은 32x32의 사각형 영역에 가보필터를 적용한 결과를 보여준다.

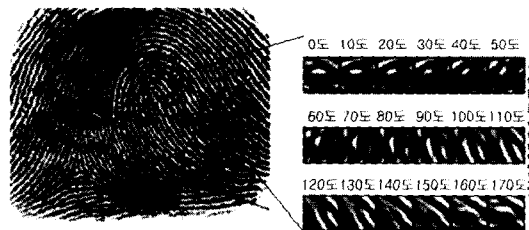


그림 2 한 특징점을 중심으로 32x32크기의 영역에 대해 가보필터링을 수행한 결과

기존 필터뱅크 기반 방법에서 8가지 방향을 적용한 것과 달리 18가지 방향에 대해 적용한다. 각도별로 얻은 가보 필터링 결과에 대해 식 (4)를 이용해 특징값들을 계산한다.

$$\frac{1}{n_i} \sum_n |F_i(x, y) - P_i| \quad (4)$$

식 (4)에서 i 는 18가지 방향 중 한 방향을 의미하고, n_i 는 i 방향의 가보필터를 적용한 영역의 화소수이며, $F_i(x, y)$ 는 i 방향의 가보필터를 적용한 결과 영상의 (x, y) 위치의 화소값이고, P_i 는 필터링 결과 영상의 평균 화소값을 의미한다.

그림 2에서 보는 바와 같이 한 특징점에 대해 18개의 필터링 결과 영상을 얻고 각각에 대해 특징값을 구하게 되며 특징점의 수에 따라 이 값들을 얻는다. 지문에서 추출한 모든 특징점들에 대해 이와 같이 값들을 얻고 이를 특징 벡터로 구성한다. 그러나 같은 지문이라 하더라도 지문영상의 획득시 주위 환경에 따라 추출되는 특징점의 수가 달라지므로 특징점들에 의존해 얻은 특징 벡터를 구성하는 값들의 개수 또한 달라진다. 그림 2의 예로 사용된 지문영상은 92개의 특징점을 가지므로 $92 \times 18 = 1656$ 개의 특징값들이 구해진다. 그러나 특징 벡터를 구성하는 값들의 수가 달라지면, 벡터의 거리를 계산하는데 문제가 되므로 이를 해결하기 위해 특징 벡터를 구성하는 값들의 수를 일치 시키는 방법이 필요하게 된다. 본 논문에서는 이것을 다음과 같은 방법으로 해결하였다.

- (1) 등록영상과 검증영상에 대해 특징점 기반 방법[5]에 따라 지문정합을 수행한다.
- (2) 등록영상의 특징벡터를 다음과 같은 방법으로 추출한다.
 - ① 등록영상을 32×32 크기의 블록으로 나눈다.
 - ② 각 블록에 포함되는 특징점들 중 한 개의 대표 특징점을 선정한다. 한 블록내에 두 개 이상의 특징점이 존재하는 경우 검증영상에서 정합을 이루는 특징점이 존재하는 것을 선택한다. 검증영상과 정합을 이루는 특징점이 존재하지 않는다면, 위쪽의 것을 선정하고 같은 행이라면 좌측의 것을 우선하여 선정한다.
 - ③ 각 블록의 대표 특징점들에 대해 그림 2와 같은 방법으로 그 블록에 대한 특징값들을 식 (4)에 따라 구한다.
 - ④ 추출한 전체 특징값들을 등록영상의 특징벡터로 한다.
- (3) 검증영상의 특징벡터를 다음과 같은 방법으로 추출한다.
 - ① 등록영상에서 특징벡터를 추출한 블록의 대표 특징점과 정합을 이루는 검증영상의 대상 특징점을 찾는다.
 - ② 대상 특징점을 찾은 경우 대상 특징점을 기준으로 그림 2와 같은 방법에 따라 대상 특징점의 특징값들을 구한다.
 - ③ 대상 특징점이 존재하지 않는 경우 특징값들은 모든 값이 0인 값으로 한다.
 - ④ 추출한 전체 특징값들을 검증영상의 특징벡터로 한다.

위 단계와 같은 방법으로 추출한 등록영상과 검증영상의 특징벡터들은 등록영상의 대표 특징점이 존재하는

블록들에 대해서만 특징값들을 추출하고, 서로 대응되는 영역들에 대한 특징값들을 포함하므로 등록영상과 검증영상의 특징벡터 간 거리를 계산할 수 있다. 특징벡터 간 거리는 유클리드 거리를 등록영상의 대표 특징점을 포함한 블록의 수로 나눈 값을 사용하였다. 이와 같은 값을 나눈 이유는 특징점이 적게 추출되는 지문영상이 적은 거리를 갖는 문제점을 피하기 위해서이다. 두 지문영상에서 추출된 특징벡터간 거리가 가까운 것은 지문의 유사성이 높고, 거리가 먼 것은 유사성이 낮음을 의미한다.

3.2 적응적 정합도 혼합 방법

Jain 등은 두 가지 지문정합 방법을 혼합하기 위해 두 정합 방법을 통해 얻은 정합도에 식 (3)과 같이 가중치를 부여해 최종 정합도로 사용한다. 이와 같이 지문정합 방법의 특성과 관계없이 같은 가중치를 부여하는 것은 각 지문정합 방법의 특성을 상쇄시키는 작용을 한다. 본 논문에서는 이러한 단점을 해결하기 위해 두 가지 방법을 통해 얻은 정합도를 구할 때, Jain 등이 제안한 방법과 달리 먼저 Jain이 제안한 방법에 따라 ERR (Equal eRror Rate)을 계산하고, ERR을 기준으로 식 (5)과 같이 결정한다.

$$S = \begin{cases} S_L & \text{if } S_j \leq ERR, \\ S_R & \text{otherwise} \end{cases} \quad (5)$$

S_L 은 FAR의 기술기가 더 큰 정합방법으로 얻은 정합도이고, S_R 은 FRR의 기술기가 더 큰 정합방법으로 얻은 정합도이다. 그리고 S_j 는 식 (3)에 의해 얻은 정합도이며, ERR_j 는 식 (3)의 결과로 얻은 정합도의 분포를 통해 얻은 ERR 위치의 정합도를 의미한다. 이와 같이 두 방법을 통해 얻은 정합도에 가중치를 부여해 더하는 방법 대신 각 방법에서 좋은 특성을 유지하도록 값을 적응적으로 선택해 각 방법의 특성이 최종 정합도에 영향을 미칠 수 있도록 할 수 있다.

4. 실험 및 검토

실험은 동일한 지문 데이터에 대해 기존 필터뱅크 기반 지문정합 방법(이하 방법1)[3], 특징점 기반 지문정합 방법(이하 방법2)[5], 그리고 3.1절에서 제안된 혼합형 지문정합 방법(이하 방법3)을 각각 구현하여 그 성능을 비교하는 것으로 수행된다. 또한 Jain 등이 제안한 정합도 혼합 방법[1]으로 방법2와 방법3의 정합도를 혼합하여 얻은(이하 방법4) 결과와 3.2절의 적응적 정합도 혼합 방법(이하 방법5)에 따라 방법2와 방법3의 정합도를 혼합하여 얻은 결과를 비교 분석해 본다. 먼저 실험을 위해 구현된 지문 인식 시스템과 실험에 사용된 지문 데이터에 대해 설명하고, 결과를 검토 및 분석해 보기로 한다.

4.1 실험 지문 영상(NIST Special Database 14)

미국 정부 산하의 NIST(National Institute of Standards and Technology)에서는 자동화된 지문 분류 시스템과 지문 인식 시스템을 평가하기 위한 지문 영상을 제공하고 있다. 본 논문에서 실험에 사용된 NIST Special Database 14(이하 NIST-14)는 미국 NCIC(National Crime Information Center)로부터 제공받은 일치하는 지문의 두 가지 영상을 각각 카드에 찍은 후, 스캐닝을 통해 832×768, 500dpi 해상도의 디지털 영상으로 저장되어 있다. 디지털 영상의 크기와 해상도는 FBI(Federal Bureau of Investigation)에서 일반적으로 사용하는 지문 영상에 상응한다[7]. 총 27,000 쌍의 지문으로 구성된 NIST-14 데이터베이스에서 F0000001.WSQ~F0001000.WSQ는 등록영상으로 하고, 이와 쌍으로 된 S0000001.WSQ~S0001000.WSQ는 검증영상으로 하여 총 1000쌍을 실험에 사용한다.

4.2 실험을 위해 구현된 지문 인식 시스템

본 논문에서 제안된 방법의 성능을 평가하기 위해 실험을 위해 구현된 지문 인식 시스템의 전체 과정은 그림 3과 같다.

실험에 사용된 NIST-14 지문 영상들은 지문 채취시

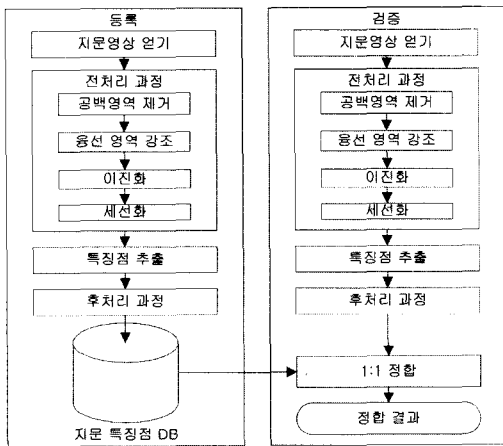


그림 3 구현된 지문인식 시스템의 순서도

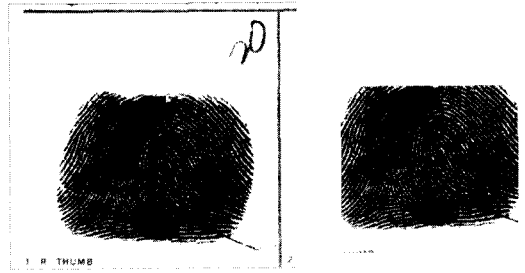


그림 4 원영상과 공백 영역 제거 영상

사용한 카드에 표기된 글씨나 지문 이외의 많은 공백 영역을 포함하고 있다. 이 영역들을 [8]에서 제시된 방법에 따라 제거하고, 512×480 크기의 지문 영상을 획득한다. 그림 4는 원영상과 불필요한 영역을 제거한 지문 영상을 보여준다.

공백영역을 제거한 512×480 크기의 영상을 16×16의 작은 영역으로 나누고 각 영역내에 포함된 융선의 방향을 구한다[9]. 각 블록에 대해 블록의 방향에 해당하는 식 (1)의 가보필터를 적용하면 융선 영역을 강조한 영상을 얻을 수 있다. 크기가 11×11인 가보필터로 융선 강조 영상을 얻고 블록별로 이진화[10]를 수행한 후 뼈대 영상을 구한다[11,12]. 그림 5는 공백 영역을 제거한 영상으로부터 구한 융선 영역 강조 영상, 이진화 영상, 그리고 뼈대 영상을 보여준다.

전처리 과정을 통해 얻은 융선들의 뼈대 영상으로부터 각 점에 대해 특징점 여부를 조사하고, 끝점(end point)과 분기점(bifurcation point)을 각각 구한다[13]. 이때 끝점의 방향은 끝점과 끝점으로부터 융선을 따라 10픽셀 추적한 지점을 직선으로 연결해 얻은 기울기로 정하고, 분기점의 경우 한 융선에서 두 융선으로 갈라질 때 분기점으로부터 두 갈라지는 융선들의 방향을 끝점과 같은 방식으로 구한 뒤, 갈라진 두 융선들의 방향에 대한 중간방향으로 정한다[14].

뼈대 영상으로부터 얻은 특징점들에는 잘못된 특징점들이 많이 포함되어 있다. 이러한 특징점들에 대해 각각 [13]에서 제안한 방법과 같이 검증을 통해 올바른 특징

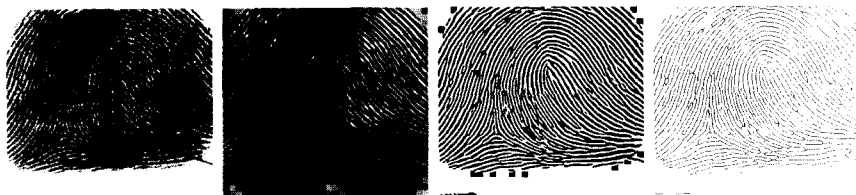


그림 5 전처리 과정에 따른 영상들. 왼쪽부터 원 영상, 융선 영역 강조 영상, 이진화 영상, 뼈대 영상

점들만을 추출하는 후처리 과정을 거친다. 그림 6은 전처리 과정과 특징점 추출과정을 통해 얻은 특징점들과 후처리 과정을 통해 검증된 특징점들을 보여준다. 후처리 과정 전에 272(끝점: 169, 분기점: 103)개인 특징점들의 수는 후처리 과정 후 92(끝점: 60, 분기점: 32)개로 감소됨을 알 수 있다.

실험에 사용된 모든 지문 영상은 등록영상과 검증영상의 쌍으로 구성되며, 두 영상의 정합도를 계산하기 위해 두 영상들로부터 각각 특징점들을 추출한 후, 등록영상과 검증영상으로부터 얻은 특징점들에 대해 [5]에서 제안한 방법으로 위치 정렬을 수행한다. 정렬이 수행된 후 최종적으로 등록영상의 각 특징점들에 대해 부합하는 점이 검증영상에서 존재하는가의 여부를 [5]의 방법

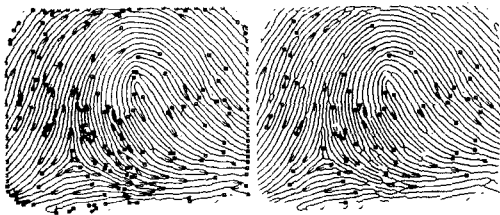


그림 6 뼈대 영상에서 얻은 특징점과 후처리 과정 후 특징점

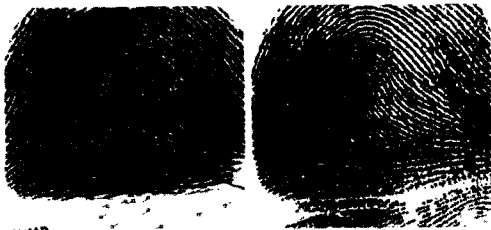


그림 7 등록영상과 검증영상에서 얻은 특징점 정합에 따라 부합하는 결과 영상

에 따라 확인한다. 그림 7은 이 과정에 따라 부합하는 특징점들을 찾은 결과를 보여준다.

4.3 실험 결과 및 분석

4.2절에 제시한 과정들을 Visual C++.NET으로 구현하고, Intel Pentium IV 2.4GHz의 CPU와 512M RAM으로 구성된 PC환경에서 진행하였다.

방법1, 방법2, 그리고 방법3의 실험에 사용된 2000개의 영상들은 화질의 상태와 상관없이 모든 영상들을 사용하였다. 이는 주관적인 방법으로 영상의 상태를 판단해 제외시키는 것이 시스템마다 다르게 영향을 받을 수 있다고 판단되었기 때문이다. 그러나 방법1의 경우 지문의 중심점 추출의 실패는 정합 결과의 실패를 가져오므로 이러한 경우에 대해서는 정당한 지문정합의 경우 가장 낮은 점수를 부여하고, 부당한 지문정합의 경우 가장 높은 점수를 부여해 정합 실패 영상의 비율을 적용하였다. 중심점을 찾는데 실패한 영상의 수는 2000개의 영상 중 등록영상 60개 + 검증영상 71개 = 131개로 전체 영상의 6.55%에 해당된다. 이 경우 총 비교횟수는 873,260번이고, 887회의 정당한 지문정합이 일어나고, 872,373회의 부당한 지문정합이 일어났다. 따라서 정당한 지문정합의 11.3%와 부당한 지문정합의 12.675%는 정합에 실패한 것이 된다. 방법2와 방법3은 모든 영상에 대해 성공적인 정합이 수행되었고, 총 비교횟수는 1,000,000번이고, 1000회의 정당한 지문정합이 일어나고, 999,000회의 부당한 지문정합이 일어났다.

세 가지 방법에 대해 [0, 100]의 값으로 정합도를 정규화 하였다. 그리고 방법1과 방법3은 거리가 가까운 것이 정합도가 높은 것이 되고, 방법2는 일치하는 특징점의 수가 클수록 높은 정합도를 나타내므로, 방법1과 방법3에 대해서는 100에서 정규화 한 값을 빼 최종 정합도를 결정하였다. 그림 8은 방법1에 의해 얻은 Genuine-Imposter 분포와 FAR-FRR을 보여준다. 100에서 Imposter가 높은 것과 0에서 Genuine이 높은 것은 중심점

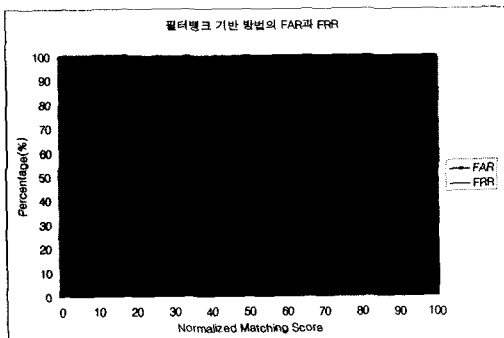
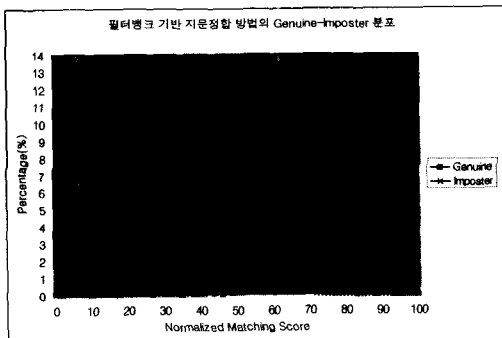


그림 8 필터뱅크 기반 지문정합 방법의 결과

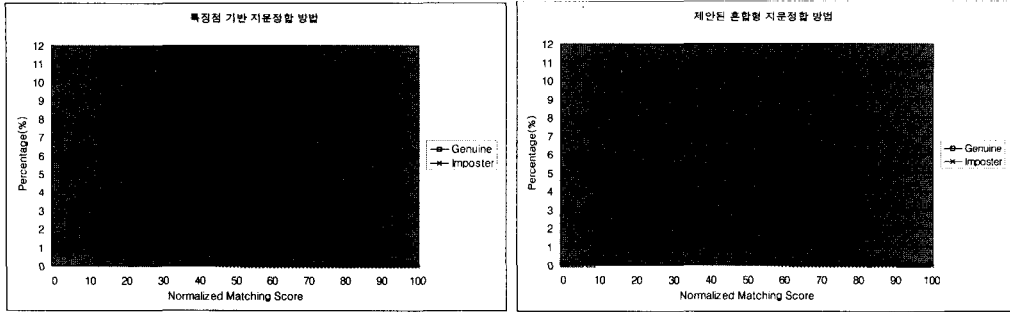


그림 9 특징점 기반 지문정합 방법과 제안된 혼합형 지문정합 방법의 Genuine-Imposter 분포

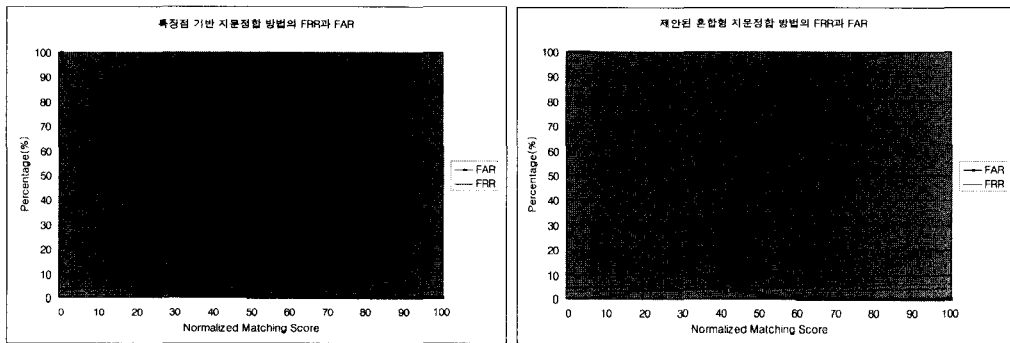


그림 10 특징점 기반 지문정합 방법과 제안된 혼합형 지문정합 방법의 FAR과 FRR

추출에 실패한 지문영상 때문이다.

그림 9는 방법2와 방법3의 Genuine-Imposter 분포를 보여주고, 그림 10은 방법2와 방법3의 FAR-FRR을 보여준다.

그림 8과 그림 10을 비교해 볼 때, ERR에서 제안된 방법3은 방법1보다 높고, 방법2에 비해 떨어지는 성능을 보였다. 표 1은 세 가지 방법에 대한 ERR 부근에서의 FAR과 FRR의 변화를 보여준다.

그러나 다시 방법2와 방법3에 대해 3.2에서 제안한 방법5를 통해 얻은 결과는 각각의 방법보다 우수한 성능을 보였다. 그림 11은 Jain 등이 제안한 방법4와 본

논문에서 제안한 방법5의 FAR과 FRR을 보여준다. 그림에서 보느냐와 같이 제안한 방법5가 Jain 등이 제안한 방법4보다 더 높은 성능을 보임을 알 수 있다.

표 2는 방법2와 방법3의 정합도를 각각 Jain 등이 제안한 방법4와 본 논문에서 제안한 방법5를 통해 얻은 최종 결과에서 ERR 주변의 FAR과 FRR을 보여준다.

표 2에서 제안된 혼합 방법은 Jain 등이 사용한 혼합 방법에 비해 ERR에서 약 1%의 성능 향상을 보여준다. 특히 Jain의 방법에서 FRR이 19.1%일 때, FAR이 20.98인데 비해, 제안한 방법은 같은 FRR에서 FAR이 17.873%로 동일한 FRR에서 약 3%의 성능향상이 이루어

표 1 세 가지 지문정합 방법에 따른 ERR 주변에서의 FAR과 FRR

(단위: %)

Normalized Matching Score		59	60	61	62	63
필터뱅크 기반 방법	FAR	27.863	25.559	23.408	21.476	19.706
	FRR	18.7	20.4	22.2	24.4	26.4
Normalized Matching Score		18	19	20	21	22
특징점 기반 방법	FAR	23.506	21.507	19.665	17.991	16.38
	FRR	18.1	18.7	19.7	19.8	20.4
Normalized Matching Score		35	36	37	38	39
제안된 혼합형 방법	FAR	26.377	23.042	19.977	17.196	14.697
	FRR	19.8	21.1	22.2	24.2	25.5

표 2 Jain의 정합도 혼합방법과 제안된 혼합방법에 대한 ERR 주변의 FAR과 FRR

Normalized Matching Score		26	27	28	29	30
Jain의 정합도 혼합방법	FAR	28.138	24.396	20.98	17.872	15.097
	FRR	16	17.8	19.1	20.5	21.8
Normalized Matching Score		22	23	24	25	26
제안한 혼합 방법	FAR	20.445	19.377	16.721	18.306	18.074
	FRR	17.9	18.2	18.5	18.8	18.8

(단위: %)

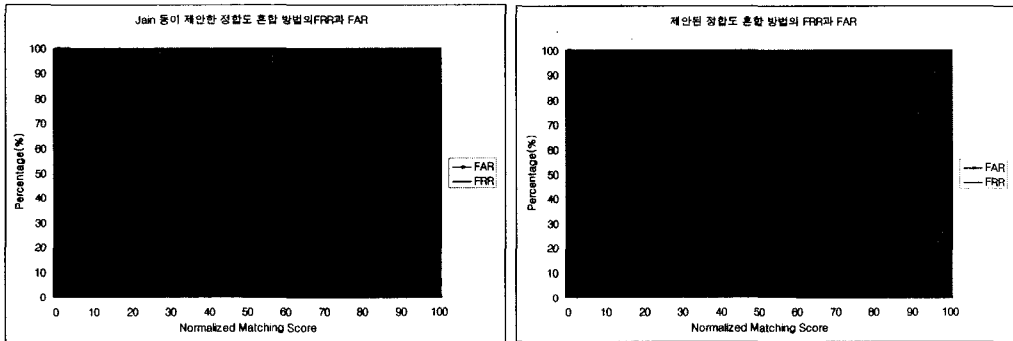


그림 11 Jain 등이 제안한 정합도 혼합 방법과 본 논문에서 제안한 혼합 방법의 FAR과 FRR

어짐을 확인할 수 있었다.

Jain 등이 제안한 방법4와 같이 지문정합 방법의 특성과 관계없이 같은 가중치를 부여하는 것은 각 지문정합 방법의 특성을 상쇄시키는 결과를 가져올 수 있다. 그림 9에서 방법2와 방법3의 Genuine-Imposter 정합도의 분포를 보면, 방법2는 부당한 정합의 경우 정합도의 분포가 왼편(낮은 정합도)으로 많이 치우쳐 있는 반면, 방법3은 정당한 정합에 대한 정합도의 분포가 오른쪽으로 치우쳐 있고 정규분포에 더 가까운 형태를 취하고 있음을 볼 수 있다. 따라서 정합도를 혼합할 때, 각 정합 방법의 특성을 반영하도록 하면, 더 좋은 결과를 얻을 수 있다. 그림 10에서 보는바와 같이 FAR의 기울기는 방법2가 더 크고 FRR의 기울기는 방법3이 더 크다는 것을 알 수 있다. 따라서 정합도를 혼합할 때, 두 정합 방법의 특성에 따라 3.2에서 제안된 식 (5)와 같이 수행할 경우 가중치를 부여해 정합도를 혼합하는 방법보다 더 높은 성능을 얻을 수 있다.

5. 결론

본 논문에서는 Jain 등이 제안한 필터뱅크 기반 지문정합 방법과 특징점 기반 지문정합 방법을 특징값 추출 과정에서 혼합하는 방법을 제안하였다. 이 방법은 [1]에서 각 지문정합의 결과로 얻은 정합도를 단순 혼합하는 방법과는 달리 특징점과 특징벡터의 성질을 함께 반영할 수 있는 특징값들로서 지문간의 고유성을 나타내고

자 하였다. 그러나 이 방법으로 구현된 지문인식 시스템의 성능은 필터뱅크 기반 방법에 비해 높은 성능의 ERR을 보이는 반면, 특징점 기반 방법보다는 낮은 성능의 ERR을 보였다. 이에 특징점 기반 방법과 제안된 혼합 방법의 정합도를 다시 혼합하는 방식을 취해 최종 시스템의 성능을 향상시킬 수 있었다. 특히, 두 정합 방법의 결과 정합도를 혼합할 때, [1]에서 제안한 방법과는 달리 두 시스템의 FAR과 FRR의 분포를 분석해 두 시스템간의 결과 정합도를 선택적으로 적용하는 방법을 취하는 적용적 정합도 혼합 방법을 제안하였다. 실험을 통해 살펴본 결과, 기존 정합도 혼합 방법에 비해 ERR에서 약 1%의 성능향상을 얻을 수 있었고, 동일한 FRR에서 FAR이 약 3% 향상됨을 확인하였다.

본 논문에서 제안한 방법들은 기존 필터뱅크 기반 방법이 지문의 중심점에 민감하다는 단점을 보완하고, 한 개의 지문을 이용한 혼합형 지문 정합방법을 향상시키는 데 적절히 응용될 수 있다. 현재 제안된 방법이 지문의 품질에 따라 어느 정도의 성능 향상을 보이는가를 평가하는 실험을 진행하고 있으며, 이는 온라인 지문인식시스템에서 실제로 적용하여 동일한 조건에서 혼합형 지문정합 방법이 기존 방법에 비해 많은 장점을 가질 수 있음을 증명할 수 있을 것이다. 그러나 이 방법은 두 가지 방법을 모두 한 시스템에 구현하므로 많은 시간을 소요하는 단점을 가진다. 이러한 방법을 해결하기 위해 지문 인식 시스템을 병렬로 구성해 처리 속도를 단축할 수 있는 방법에 관한 연구도 진행될 것이다.

참고 문헌

[1] Arun Ross, Anil Jain, James Reisman, "A hybrid fingerprint matcher," Pattern Recognition 36. pp.1661-1673, The Journal of the Pattern Recognition Society, 2003.

[2] 정석재, 김동윤, "필터뱅크 기반 지문정합에서 빠른 지문 정렬 방법 및 가중치를 부여한 특징 벡터 추출 방법", 한국정보과학회 논문지. SA 31(1), pp.71-81, 2004. 1.

[3] A. K. Jain, S. Prabhakar, L. Hong and S. Pankanti "Filterbank-based Fingerprint Matching," IEEE Transactions on Image Processing, Vol. 9, No.5, pp. 846-859, May 2000.

[4] A. K. Jain, S. Prabhakar, L. Hong, "A Multi-channel Approach to Fingerprint Classification," IEEE Transactions on PAMI, Vol.21, No.4, pp.348-359, April 1999.

[5] Nalini K. Ratha, Kalle Karu, Shaoyun Chen, Anil K. Jain, "A Real-Time Matching System for Large Fingerprint Databases," IEEE Transactions on PAMI, Vol.18, No.4, pp.799-813, 1996.

[6] Jain, A., Lin, H., Bolle, R., "On-line fingerprint verification," IEEE Transactions on PAMI, Vol.19, No.4, pp.302-313, 1997.

[7] M.D. Garris, C.I. Watson, "NIST Fingerprint Image Software," CD-ROM and documentation, 2001.

[8] C. I. Watson, G. T. Candela, P. J. Grother, "Comparison of FFT Fingerprint Filtering Methods for Neural Network Classification," Technical Report NISTIR 5493 September 1994.

[9] Salil Prabhakar, "Fingerprint Classification and Matching Using a Filterbank," A dissertation for the degree of doctor of philosophy, Computer Science & Engineering, Michigan State Univ., 2002.

[10] Otsu, N., "A Threshold Selection Method from Grey-level Histograms," Transactions on Systems, Man Cybernetics. Vol.9. 1:377-393, 1979.

[11] J. R. Parker, "Algorithms for Image Processing and Computer Vision," Wiley Computer Publishing, 1997.

[12] Zhang S., K. S. Fu, "A Thinning Algorithm for Discrete Binary Images," Proceedings of the International Conference on Computers and Applications. Beijing, China., 879-886, 1984.

[13] Alessandro Farina, Zsolt M. Kovacs-Vajna, Alberto Leone, "Fingerprint minutiae extraction from skeletonized binary images," Pattern Recognition 32. pp.877-889, The Journal of the Pattern Recognition Society, 1999.

[14] Yuliang He, Jie Tian, Xiping Luo, Tanghui Zhang, "Image enhancement and minutiae matching in fingerprint verification," Pattern Recognition Letters 24, 1349-1360, 2003.

정석재

정보과학회논문지 : 소프트웨어 및 응용
제 31 권 제 1 호 참조



박상현

2002년 2월 아주대학교 정보통신대학 정보 및 컴퓨터공학과 학사. 2004년 2월 아주대학교 정보통신전문대학원 정보통신공학과 석사. 2004년 3월~현재 (주)알폰스테크 연구원. 관심분야는 컴퓨터 비전, 영상처리, 영상압축, 멀티미디어



문성림

1999년 2월 아주대학교 정보통신대학 정보 및 컴퓨터공학과 학사. 1999년 3월~2002년 2월 한국품질환경인정협회(Korea Accreditation Board) 근무. 2002년 3월~2004년 2월 아주대학교 정보통신전문대학 정보통신공학과 석사. 2003년 9월~현재 영국 Middlesex University 석사(졸업 예정). 2004년 3월~현재 아주대학교 정보통신대학 정보 및 컴퓨터공학과 박사과정. 관심분야는 컴퓨터비전, 영상처리, 지문인식, 로봇비전

김동윤

정보과학회논문지 : 소프트웨어 및 응용
제 31 권 제 1 호 참조