

포인트 패턴 기반의 고속 지문인식 알고리즘 설계

나호준*, 김창수*

*부경대학교 정보보호학과

Design of High Speed Fingerprint Identification Algorithm based on Point Pattern

Ho-Jun Na*, Chang-Soo Kim

Department of Information Security, Pukyong National Univ.

요약

특징점(minutiae)에 기반한 정합 방식은 간결하고 계산이 쉬워서 가장 널리 사용되는 반면 동일한 지문의 두 입력 사이에서 발생할 수 있는 여러 가지 왜곡들을 고려해야 한다. 본 논문에서는 이러한 왜곡들을 줄이고 포인트 패턴을 이용하여 고속 지문정합 알고리즘을 설계하였으며 1:N정합에 적합한 알고리즘이다. 특징점 추출 알고리즘에서 구한 특징점에 직교좌표의 정보를 조합한 특징점들 중에서 정합을 시도하려는 두 개 특징점 데이터 집합에 대하여 지역 정합을 시도하여 지역 정합된 특징점의 비율이 임계값을 넘어가면 전역 정합을 시도하여 정합 점수를 결정하고 가장 높은 정합수를 얻은 지문 데이터를 결정한다.

I. 서론

오늘날 지문인식은 그 사용의 편리성과 비교적 높은 보안성에 의하여 가장 널리 연구되는 생체인식 기술 중에 하나이며, 최근에는 지문 영상을 획득하는 입력 부분에서부터 본인 여부를 결정하는 최종 정합까지, 대규모 사용자를 위하여 고 인식의 신뢰성을 얻기 위한 다양한 연구가 진행되고 있다.

지문의 분류와 정합의 정도는 지문영상에 존재하는 특징점(minutiae)의 수와 방향성의 흐름 패턴을 이용한 특이점(singular point) 정보에 의존한다. 특징점은 분기점(bifurcation), 단점(ridge ending)을 말하며 이들의 갯수와 방향성에 의해 지문영상의 특징이 정해지며, 이 점들은 정합을 위한 등록점으로 사용되어지며, 특이점은 중심점(core)과 삼각주(delta)를 말하며 이것 또한 지문 검증을 위한 중요한 정보로 이용된다.[1]

특징점(minutiae)에 기반한 정합 방식은 간결하고 계산이 쉬워서 가장 널리 사용되는 반면 동일한 지문의 두 입력 사이에서 발생할 수 있는 여러 가지 왜곡들을 고려해야 한다.

센서 위에 불균등한 압력으로 인한 비선형성 뒤틀림(distortion)과 센서 위에 대는 손가락의 위치 차이로 인한 평행이동(translation) 및 회전이동(rotation) 그리고 추출 알고리즘의 오류로 인한 특징점의 상실과 의사(spurious)특징점 등이 이런 왜곡이다.

본 논문에서는 위에서 나열한 왜곡들을 최소한으로 줄이고 포인트 패턴 정합방식을 이용하여 1:N정합에 적합한 지문 인식 알고리즘을 설계하였다.

II. 연구 동향

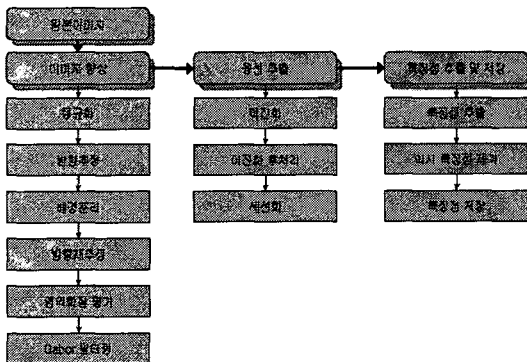
지문에 대한 보다 체계적이고 과학적인 연구는 1800년대 중반부터 시작되었는데, Edward Henry는 1880년대에 전체적인 융선 모양을 기준으로 지문을 5종류(궁상문, 돌기궁상문, 좌계상문, 우계상문, 외상문)로 분류한“Henry System”을 발표하였고[2], 이는 오늘날까지도 일부 지문인식에서 대용량 데이터베이스에서의 검색시간 단축등을 목적으로 사용되고 있다.

지문인식 기술은 사용하는 특징에 따라 특징점 기반방식, 영상 기반 방식, 그리고 주파수 영역에서 특징추출이나 필터링을 통한 특징추출 후 인식하는 방식으로 나눌 수 있다. 최근에는 인식 성능을 높이기 위해 특징점과 그 이외의 특징을 조합하는 방식[3] 및 여러 인식 알고리즘의 결과를 조합하여 최종 인식을 하는 연구[4][5]와 지문의 압력에 의한 왜곡을 보상하여 인식하는 연구[6] 등이 진행되고 있다.

특징점 기반 방식에는 각 특징점 주변의 다른 특징점들의 조합으로 지역적 구조(Local Structure)를 형성하여 등록 영상과 입력 영상간의 기준점을 찾고 그 기준점을 중심으로 영상들을 정렬하여 정합하는 방식이 주류이다 [7][8]. 두 영상을 정렬 시 모든 특징점을 이용하는 것은 시간 소모가 많으므로 지문의 중심점 주변의 특징점이나 영상을 이용하여 시간을 단축시키기도 한다[9]. 특징점을 구성하는 정보로는 특징점의 좌표와 형태 그리고 방향 등이 있는데 이외에 특징점 주변의 융선 방향이나 융선 자체를 이용한 방법도 있다[10].

2.1 지문 특징점 추출 시스템

지문 특징점 추출과정은, 입력으로 들어온 원본 그레이(gray-level) 지문 이미지를 이미지 향상, 융선 추출, 특징점 추출 단계를 거쳐, 최종적으로 지문 특징점(minutiae)을 추출해 낸다. 이 지문 특징점 추출과정은 기본적으로[11]의 접근방법을 따르고 있다. [그림 1]은 지문 특징점 추출과정을 개략적으로 보여주고 있다.



[그림 1] 지문 특징점 추출과정
이미지 향상 단계에서는 정규화를 거친 후

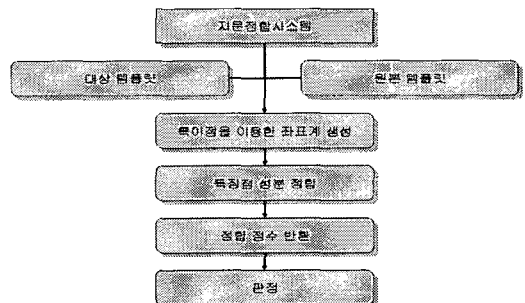
일정한 크기의 블럭에 대하여 방향을 구한 후 배경을 분리해 지문 영역을 추출한다. 먼저 배경을 분리해야 이후 단계에서 계산 부하를 줄일 수 있다. 이 때, 주위 블럭과 비교해서 방향이 크게 어긋나는 지역은 해상도를 높여서 다시 방향을 계산한다. 다음으로 지문 내의 각각의 블럭에 대하여 화질평가를 수행한다. 마지막으로 앞에서 구한 블럭의 방향을 이용하여 게이버 필터링(gabor filtering)을 수행하여 이미지 향상을 꾀한다

융선추출 단계에서는 지역 평균과 분산을 이용하여 적응 이진화(adaptive thresholding)를 수행한 후 융선 내의 구멍(hole)과 외부의 얼룩(speckle)을 제거한다. 이후 이진화 이미지에 세션화 알고리즘을 적용하여 1 픽셀(pixel) 두개의 골격(skeleton)을 뽑아낸다.

특징점 추출 단계에서는 특징점을 추출한 후 후처리(post-processing)을 거쳐서 의사특징점을 제거하고 최종적으로 특징점을 추출한다.

2.2 지문 정합 시스템

지문정합 시스템에서는 지문 특징점 추출 시스템에서 얻어진 원본 템플릿과 정합하려는 대상 템플릿을 이용하여 두 지문간의 유사도를 측정하여 판정한다. 우선 두 템플릿의 특이점을 이용하여 직교좌표계를 생성하고 각 좌표계에 생성된 특징점의 성분을 지역정합을 실시하고 일정한 임계값 이상이면 전역정합을 실시하여[12] 각 지문정보 성분에 가중치 값을 부여하여 계산한 후 정합되면 같은 지문으로 판정한다. 아래 [그림 2]는 지문정합시스템의 단계를 보여주고 있다.



[그림 2] 지문정합시스템 정합단계

III. 고속 지문 정합 알고리즘 설계

본 논문에서 제안하는 방식은 1:N 방식의 지문인식시스템에 적합한 알고리즘으로 방법은 포인트 패턴 매칭을 기본으로 하여 특징점의 속성인 위치 정보, 방향 정보, 특징점의 종류를 사용하여 구성된다. 포인트 패턴 매칭에서는 각각의 특징점의 좌표가 아래 (식 3.1)과 같은 식을 통하여 다른 특징점의 좌표로 매핑되는 것을 찾아야한다.

$$T \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \end{pmatrix} + s \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (\text{식 3.1})$$

즉 한 점이 다른 템플릿의 다른 특징점으로 이동하는 것은 Scale과, 회전된 각도 그리고 X축과 Y축의 방향으로의 이동인 t_x , t_y 등 네가지 변수가 필요하다. 따라서 정합 과정에서는 정합된 결과를 추출해내기 위해서 기준점을 가지고 위의 네 가지 변수를 추출해 낸 후 전체 특징점에 적용하여 정합되는지 여부를 판단하여야 한다.

3.1 지역 정합(Local matching)

정합을 시도하려는 특징점 데이터 Set A와 B에 대하여 A의 한 특징점 p, B의 한 특징점 q, 지역 정합의 시도횟수를 결정하는 k2, k3, 그리고 정합 확률의 임계값 ρ 가 입력으로 받는다.

이때 p와 q의 이웃 중 가장 먼 것들 중 k2개, q의 이웃 중 역시 가장 먼 것들 k3개를 조합하여 총 k2*k3개의 페어를 만들수 있고, 이때 선택된 p의 이웃 특징점을 a, q의 이웃 특징점을 b라 하면 이 네개의 특징점의 좌표를 기준으로 A에서 B로 변환되는 페러미터를 추출할 수 있다. 추출하는 방법은 (식 3.2)와 같다.

$$s = \frac{|\overrightarrow{qb}|}{|\overrightarrow{pa}|},$$

$$\theta = \text{angle from } \overrightarrow{pa} \text{ to } \overrightarrow{qb}, \quad (\text{식 3.2})$$

$$t_x = q_x - p_x s \cos(\theta) + p_y s \sin(\theta),$$

$$t_y = q_y - p_x s \sin(\theta) + p_y s \cos(\theta).$$

그러나 p와 a특징점의 각도를 θ (rotation)만큼 회전시켰을 때 q와 b의 특징점의 각도를 각각 비교하여 두 점간의 각도의 차이가 하나라

도 90°를 벗어나는 경우, 혹은 s(Scale)의 값이 0.8보다 작거나 1.2보다 더 큰 경우는 더 이상 진행하지 않는다. 이는 지역 정합이 될 가능성이 적기 때문이며 이를 통해 수행시간을 단축할 수 있다.

p의 모든 이웃 점에 대하여 T를 통해 변환한 점이 q의 이웃 점과 정합되는지를 확인하며 정합되는 두 특징점은 배열 L1과 L2의 같은 위치에 저장되게 된다. 위의 과정을 P의 K개의 모든 이웃에 대하여 반복한 후 정합된 비율이 ρ 보다 크게 되면, 즉 K*p보다 더 큰 개수만큼 L1과 L2에 각각 특징점이 저장되면 L1과 L2를 가지고 전역 정합을 시도한다.

3.2 전역 정합

우선 지역 정합에서 주어진 지역정합의 결과 L1과 L2리스트를 가지고 A의 모든 특징점에 대해서 특징점의 좌표와 각도를 가지고 정합되는 B의 특징점의 리스트를 작성한다. 이때 각각의 정합되는 페어에 대하여 거리와, 특징점의 종류, 그리고 각도를 가지고 가중치를 주어 모든 조건이 다 일치할 경우 1점이 주어지도록 한다. 이때의 각각의 가중치는 [표 1]과 같다.

[표 1] 정합 점수 가중치

각도의 차이(δ)	특징점의 종류	거리
$\delta < 33.75^\circ : 1$	같은 경우 : 1	$\left(1 - \frac{2}{3} \times \frac{\text{현재거리}^2}{\text{기준거리}^2}\right)$
$33.75^\circ \leq \delta < 67.5^\circ : 0.7$	다른 경우 : 0.8	
$67.5^\circ \leq \delta < 90^\circ : 0.4$		
$\delta \geq 90^\circ : 0.1$		

완벽히 일치할 경우 1점이 더해지게 되며, 최악의 경우에는 0.1*0.8*0.33점이 더해지게 된다. 이로서 유사 특징점으로 인한 잘못된 정합의 경우를 상당히 감소시킬 수 있다. 위의 과정을 반복하여, 얻어진 점수를 총합하여 반환하게 된다. 따라서 모든 점이 정확히 정합될 경우의 점수는 정합시킨 특징점의 개수와 같게 된다.

정합 점수가 가장 높은 지문 데이터를 결정하게 된다.

IV. 결론

본 논문에서는 포인트 패턴을 이용하여 기존의 정합 알고리즘 보다 빠른 지문정합 알고리즘을 설계하였으며 1:N정합에 적합한 알고리즘이다. 가장 많이 사용되는 특징점 추출 알고리즘은 전처리 과정을 거쳐 후보 특징점을 추출하고 후처리 과정에서 의사 특징점을 제거하는 알고리즘이다. 본 논문에서 설계한 정합 알고리즘은 특징점 추출 알고리즘에서 구한 특징점에 직교좌표의 정보를 조합한 특징점들 중에서 정합을 시도하려는 특징점 데이터 Set A와 B에 대하여 지역 정합을 시도하여 지역 정합된 특징점의 비율이 임계값을 넘어가면 전역 정합을 시도하여 정합 점수를 결정하고 가장 높은 정합수를 얻은 지문 데이터를 결정한다.

[참고문헌]

- [1] Leong Chung Ern, Chazali Sulong "Fingerprint classification approaches :an overview", IEEE ISSPA, 6th International Symposium on Signal Processing and its Applications, Vol. 1, pp.347-350, Aug. 2001
- [2] Henry, E.R., "Classification and Uses of Fingerprints", Routledge, London, 1900
- [3] A. K. Jain, A. Ross, and S. Prabhakar, "Fingerprint Matching Using Minutiae and Texture Features", Proc. International Conference on Image Processing (ICIP), Vol.3, pp282-285, Oct. 2001
- [4] A. J. Willis, L. Myers "A cost-effective fingerprint recognition system for use with low-quality prints and damaged fingertips", Pattern Recognition, Vol.34, No.2, pp255-270, Feb. 2001
- [5] Asker M. Bazen, Sabih H. Gerez "Fingerprint matching by thin-plate spline modelling of elastic deformations", Pattern Recognition, Vol.36, No.8, pp1859-1867, Aug. 2003
- [6] Vivek A. Sujana, Michael P. Mulqueen, "Fingerprint identification using space invariant transforms", Pattern Recognition Letters, Vol.23, No.5, pp609-619, Mar. 2002
- [7] Ratha N.K., Bolle R.M., Pandit V.D., and Vaish V., "Robust fingerprint authentication using local structural similarity", Applications of Computer Vision, 2000, Fifth IEEE Workshop on, pp29 -34, 2000
- [8] Xudong Jiang, Wei-Yun Yau, "Fingerprint minutiae matching based on the local and global structures", Pattern Recognition, 2000. Proceedings. 15th International Conference on, Vol.2, pp1038 -1041, 2000
- [9] Weiwei Zhang, Yangsheng Wang, "Core-Based Structure Matching Algorithm of Fingerprint Verification", Proceedings 16th International Conference on Pattern Recognition, Vol.1, pp70-74, Aug. 2002
- [10] Tico, M., Kuosmanen, P., "Fingerprint matching using an orientation-based minutia descriptor", Pattern Analysis and Machine Intelligence, IEEE Transactions on, Vol.25 No.8, pp1009-1014, Aug. 2003
- [11] L. Hong, Y. Wan, and A. K. Jain, "Fingerprint Image Enhancement Algorithm and Performance Evaluation", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.20, No.8, pp777-789, Aug. 1998
- [12] Van Wamelen P.B., Lib Z., Iyengar S.S., "A fast expected time algorithm for the 2-D point pattern matching problem", Pattern Recognition, Vol.37, No.8, pp1699-1711, Aug. 2004.