

적응적 거리를 이용한 지문 정합 방법

이 동 재(李東宰), 김 선 주(金善柱), 이 상 준(李尙俊), 김 재 희(金在熹)
 연세대학교 전기·전자공학과
 전화 : (02) 2123-4537 / 팩스 : (02) 312 - 4584

Fingerprint Minutia Matching Using Adaptive Distance

Dongjae Lee, Seonjoo Kim, SangJun Lee, Jaihie Kim
 School of Electrical and Electronic Engineering, Yonsei University
 E-mail : poseldon@seraph.yonsei.ac.kr, poseldon@nownuri.net

Abstract

We propose a new fingerprint minutia matching algorithm which matches the fingerprint minutiae by using adaptive distance. In general, fingerprint is deformed by pressure and orientation when a user press his fingerprint to sensor. These nonlinear deformations change the distance between minutiae and reduce verification rate. We define the adaptive distance using ridge frequency. Adaptive distance normalizes the distance between minutiae and compensates for nonlinear deformation. Our algorithm can distinguish two different fingerprints better and is more robust. Experimental results show that the performance of the proposed algorithm is superior to using Euclidean distance.

I. 서론

지문은 융선(ridge)과 골(valley)로 구성되어 있고, 융선과 골의 모양이 이루는 형태에 따라 끝점(end-point)과 분기점(bifurcation)을 형성하며, 이를 특징점(minutiae)이라 한다. 특징점은 그림 1과 같은 좋은 상태의 지문은 약 40~100개 정도의 특징점을 포함하고 있다[1][2]. 이러한 특징점들의 위치는 동일인이 입력하더라도 입력되는 시간, 센서에 손가락을 누르는 압력 그리고 손가락이 눌리는 방향 등에 따라 지문의 모양이 변형되고, 특징점들의 위치와 형태도 변형되어

진다. 그러므로, 입력되는 두 지문이 동일인인지 아닌지를 판별하는 문제는 매우 어려워진다.

이러한 문제점들을 해결하기 위하여 A. K. Jain 등은 특징점을 포함한 융선을 이용하여 등록된 특징과 입력된 특징을 일치시키고, 일치된 특징점들을 정합하기 위하여 adaptive elastic 정합 알고리즘을 사용하는 alignment-based 정합방법을 제안하였다[3][4]. 그러나, 이 방법은 성능은 우수하나 각각의 특징점을 위한 융선을 추가적으로 저장하고 있어야 하므로 큰 저장공간이 필요한 단점을 가지고 있다. 또 A. K. Hrechak과 J. A. Mchugh 등은 특징점들 사이의 관계를 이용하여 지역적 구조(local feature)를 사용한 정합 방법을 제안하였다[5]. 이 방법은 처리시간과 회전되거나 부분적으로 입력된 지문의 경우에도 강한 장점을 가지고 있다. 이러한 장점 때문에 좀더 향상된 방법이 A. Wahab과 X. Jiang 등에 의해 제안되었다[6][7]. 그러나, 이 논문들에서 사용된 지역적인 구조들은 서로 다른 손가락에서도 비슷한 구조를 가질 수 있고, 같은 손가락에서도 손의 상처나 오물 등에 의해서 다른 형태의 구조로 형성될 수 있기 때문에 정확도가 떨어질 수밖에 없는 단점을 가지고 있다.

본 논문에서는 지문의 변형정도를 융선의 빈도수(ridge frequency)를 이용한 적응적 거리를 정의하여 정합하는 방법을 제안한다. 2장에서는 두 지문을 일치(alignment) 시키는 방법을 소개하고, 3장에서는 일치된 두 지문사이의 특징점을 적응적 거리를 이용한 정합 알고리즘을 소개한다.



(a) 끝점 (b) 분기점
 그림 1. 지문의 특징점

II. 두 지문의 일치(Alignment) 방법

두 지문을 일치시키는 방법은 [6][7]에 사용된 방법을 변형하여 사용하였다.

지문으로부터 검출된 특징점 P_k 는 다음과 같은 특

징 벡터로 구성되어 진다.

$$FV_k = (x_k \ y_k \ \phi_k \ t_k)^T \quad (1)$$

여기서 $(x_k \ y_k)$ 는 특징점의 좌표, ϕ_k 는 특징점의 방향(orientation), t_k 는 특징점의 형태(끝점 혹은 분기점)를 나타낸다. 특징 벡터를 이용하여 일치할 하기 위해서 지역적인 구조를 형성하여야 하는데 지역적인 구조(Local Feature) LF_k 는 하나의 특징점(k)을 중심으로 이웃하는 m개의 특징점(1,2...m)을 이용하여 그림 2와 같이 구성된다.

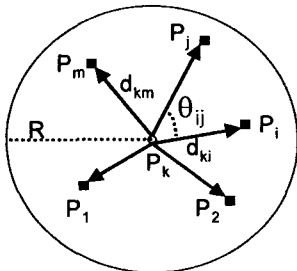


그림 2 지역적 구조(Local Feature)

여기서 d_{ki} 는 상대적인 거리, θ_{ij} 는 상대적인 각도 (relative angle), ϕ_{ki} 는 특징점의 상대적인 방향 (relative orientation)를 나타내며, 중심 특징점 P_k 와 m-nearest neighbor인 특징점 P_i 사이에서 다음과 같이 계산되어 진다.

$$d_{ki} = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2} \quad (2)$$

$$\theta_{ij} = \tan^{-1}\left(\frac{y_k - y_i}{x_k - x_i}\right) - \tan^{-1}\left(\frac{y_k - y_j}{x_k - x_j}\right) \quad (3)$$

$$\phi_{ki} = \phi_k - \phi_i \quad (4)$$

특징점 P_k 에서 지역적인 구조를 형성할 경우 다음과 같은 형태가 얻어진다.

$$LF_k = (d_{k1} \dots d_{km}, \theta_{12} \dots \theta_{im}, \phi_{k1} \dots \phi_{km}, t_k t_1 \dots t_m)^T \quad (5)$$

여기서 P_1 은 특징점 k에 가장 가까운 특징점이고, P_2 는 두 번째로 가까운 특징점인 경우이다.

등록된 지문 영상의 지역적인 구조 LF_k^R , 입력된 지문 영상의 지역적인 구조 LF_k^I 라고 가정하면, 각 지역적 구조들의 유사도는 다음과 같이 정의된다.

$$SIM(i, j) = \frac{W_D \cdot D(i, j) + W_\theta \cdot \Theta(i, j) + W_\phi \cdot \Phi(i, j) + W_T \cdot T(i, j)}{W_D + W_\theta + W_\phi + W_T} \quad (6)$$

여기서 $D(i, j)$ 는 거리에 대한 차이, $\Theta(i, j)$ 는 특징

점의 상대적인 방향 차이를 그리고 $\Phi(i, j)$ 는 상대적인 각도 차이를 각각 나타낸다.

$$D(i, j) = \begin{cases} \frac{T_D - |LF_i^I - LF_j^R|}{T_D}, & \text{if } |LF_i^I - LF_j^R| < T_D \\ 0, & \text{Others} \end{cases} \quad (7)$$

$$\Theta(i, j) = \begin{cases} \frac{T_\theta - |LF_i^I - LF_j^R|}{T_\theta}, & \text{if } |LF_i^I - LF_j^R| < T_\theta \\ 0, & \text{Others} \end{cases} \quad (8)$$

$$\Phi(i, j) = \begin{cases} \frac{T_\phi - |LF_i^I - LF_j^R|}{T_\phi}, & \text{if } |LF_i^I - LF_j^R| < T_\phi \\ 0, & \text{Others} \end{cases} \quad (9)$$

$$T(i, j) = \begin{cases} 1, & \text{if } LF_i^I = LF_j^R \\ 0, & \text{others} \end{cases} \quad (10)$$

등록된 지문과 입력된 지문을 비교하여 유사도가 가장 높은 지역적인 구조의 중심 특징점을 일치기준점으로 설정한다.

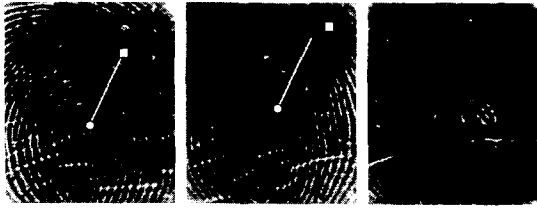
등록 영상에서의 중심 특징점 P_k^R 과 입력 영상에서의 중심 특징점 P_j^I 를 중심으로 하는 지역 구조가 가장 유사도가 높을 경우 일치 벡터 AV_{IR} 은 다음식으로 표현된다.

$$AV_{IR} = (X = x_i^I - x_j^R, Y = y_i^I - y_j^R, \Theta = \theta_i^I - \theta_j^R) \quad (11)$$

여기서 (X, Y) 는 평행이동(translation)에 대한 매개변수이고, Θ 는 회전(rotation)에 대한 매개변수이다.

III. 적응적 거리를 이용한 정합

2 절에서 구한 일치 매개변수를 이용하여 입력된 영상과 등록된 영상을 일치시킨 후, 두 개의 영역이 중복된 부분에서만 특징점을 정합하게 된다. 이 때 등록된 영상의 특징점을 대상으로 일정 영역 안에 동일한 특성(특징점의 형태, 거리, 특징점의 방향)을 가진 입력 영상의 특징점이 존재하는지를 판단하게 되는데, 기존의 논문들에서는 여기에 사용되는 거리를 유클리디안 거리(Euclidean distance) 혹은 융선의 개수(Ridge count)를 사용한다[6][7]. 그러나, 지문은 입력될 때 손가락이 센서에 눌리는 압력, 눌리는 힘의 방향에 따라 입력되는 융선사이의 거리가 틀려지기 때문에 같은 특징점 사이의 거리라 하더라도 입력 조건에 따라 그림 3의 (a),(b)와 같이 그 거리는 변하게 되는 문제점이 있다. 또한, 융선의 개수를 사용하는 경우에도 품질이 좋은 지문영상에서는 문제가 없지만 그림 3의 (c)처럼 부분적으로 너무 진하거나 너무 흐려서 지문의 융선이 제대로 형성되지 않는 경우 정확한 융선의 개수를 알기 힘든 문제점이 있다. 본 논문에서는 이러한 문제점을 극복하기 위하여 지문 융선의 빈도수(ridge frequency)를 이용하여 거리에 가중치를 부여함으로써 지문의 입력 상태에 따른 변형이나 부분적으로 진하거나 흐리게 입력되는 곳에서 발생하는 거리에 대한 문제점을 해결하였다.



(a) (b) (c)
 그림 3 (a),(b)는같은 특징점 사이의 거리가 지문의 변형 때문에 거리차가 나는 경우
 (c)는 영상의 품질이 좋지 않아 융선의 개수를 세기 힘든 경우

3.1 지문 융선의 빈도수(ridge frequency) 구하기

지문 융선의 빈도수를 구하는 방법은 Hong 등이 제안한 방법을 사용한다[1].(그림 4)

- 1) 입력 영상 G를 $w \times w$ 크기의 블록으로 나눈다.
- 2) 각각의 블록 중심에서 융선의 흐름에 수직인 방향으로 $l \times w$ 크기의 방향원도우를 설정한다.
- 3) 각각의 블록에서 융선과 골의 x-signature를 구한다.

$$X[k] = \frac{1}{w} \sum_{d=0}^{w-1} G(u, v) \quad k=0, 1, \dots, l-1 \quad (12)$$

$$u = i + (d - \frac{w}{2}) \cos \alpha(i, j) + (k - \frac{l}{2}) \sin \alpha(i, j) \quad (13)$$

$$v = j + (d - \frac{w}{2}) \sin \alpha(i, j) + (\frac{l}{2} - k) \cos \alpha(i, j) \quad (14)$$

- 4) x-signature에서 두 개의 연속적인 첨점(peak) 사이의 평균 거리에 대한 역수로 융선의 빈도수 $\Omega(i, j)$ 를 구한다.

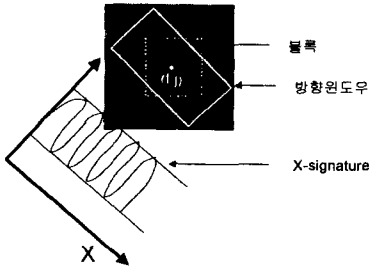


그림 4 방향원도우와 x-signature

3.2 적응적 거리

정합시 등록영상의 특징점과 정합되는 특징점을 입력 영상에서 찾기위해서 등록영상에서 일치기준점으로 부터 특징점까지의 거리와 방향을 이용하여 입력영상에 이에 해당하는 한계영역(tolerance box)을 설정하여 그 안에 같은 특성을 갖는 특징점이 존재하는가로 정합을 결정한다. 이 때, 일치 기준점으로부터 특징점까지의 거리는 가까운 경우 지문의 변형에 큰 영향을 받지 않지만, 거리가 멀어질수록 지문의 변형에 큰 영향을 받

게 되어 에러율을 높이는 요인으로 작용한다. 이를 극복하기 위해 Luo 등은 거리에 따라 한계영역의 크기를 조정하였으나, 이는 거리가 멀어질수록 한계영역이 커져서 원하는 특징점 이외의 다른 특징점들이 정합될 수 있는 문제점이 있다[8].

본 논문에서는 빈도수를 이용한 두 특징점 사이의 적응적 거리를 이용하여 지문 변형에 따른 영향을 감소시켜 정합율을 높이는 알고리즘을 제안한다. 적응적 거리 D_{adapt} 은 다음과 같이 정의된다.

$$D_{adapt} = \frac{1}{\Omega_{norm}} \sum_{i=1}^N D_i \cdot \Omega_i \quad (15)$$

여기서 N은 두 특징점을 잇는 직선이 지나는 블록의 개수이고, D_i 는 i 번째 블록에서 직선의 판통거리, Ω_i 는 i 번째 블록의 빈도수, Ω_{norm} 은 정규화 상수를 나타낸다.

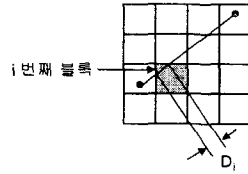


그림 5 적응적거리의 계산

이와 같이 두 특징점 사이의 거리를 D_{adapt} 으로 정의하여 입력 상태에 따라 지문이 변형되어 생겼던 특징점 사이의 거리 변화에 대한 오류를 최소화하도록 하였다.

3.3 정합

등록 영상에서 M개의 특징점 집합을 R, 입력 영상에서 N개의 특징점 집합을 I 라고 하고, 각 특징점들의 좌표를 일치기준점을 중심으로 한 극좌표계로 변환하면 다음 식과 같다.

$$R = \{(r_1, \theta_1, \phi_1), \dots, (r_M, \theta_M, \phi_M)\} \quad (16)$$

$$I = \{(r_1, \theta_1, \phi_1), \dots, (r_N, \theta_N, \phi_N)\} \quad (17)$$

각각의 등록 영상의 특징점을 중심으로 한계영역 ($\Delta r, \Delta \theta$)을 설정하여 그 안에 입력 영상의 특징점이 존재할 경우 두 특징점의 특성(특징점방향 ϕ , 형태 t)을 비교하여 정합점수(matching score)를 결정한다.

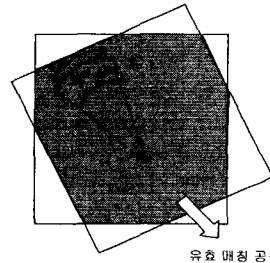


그림 6 유효정합공간과 한계영역의 설정

유효정합 공간 내에 위치한 모든 특징점들에 대해서 위의 과정을 한 다음 전체 정합점수를 이용하여 두 영상의 정합여부를 최종적으로 결정한다.

IV. 실험 결과

본 실험은 30명의 지문 각 10개씩, 총 300개의 지문 영상에 대해서 실시하였다. 지문 영상은 광학방식 지문센서로 취득하였다(영상 크기 : 248*292, 실험 환경 : Pentium III - 450Mhz).

제안한 정합방법의 검증 성능은 다음 표4-1과 같다. 여기서, reject_num은 reject된 영상의 수를 나타내고 correct_num은 올바르게 검증된 영상의 수를 나타내며 false_num은 잘못 검증된 영상의 수를 나타낸다고 하면, verification rate와 reject rate는 다음의 식으로 계산되어 진다.

$$verification\ rate = \frac{correct\ num}{correct\ num + false\ num} \times 100\% \quad (18)$$

$$reject\ rate = \frac{reject\ num}{total\ image\ num} \times 100\% \quad (19)$$

표 4-1 Reject rate에 따른 Verification rate의 변화

| Verification Rate | Reject Rate |
|-------------------|-------------|
| 99.95% | 14.0% |
| 99.71% | 10.6% |
| 99.52% | 7.3% |
| 99.38% | 4.0% |

본 논문에서 제안한 적응적 거리를 사용했을 때의 검증률과 유클리디안 거리를 사용했을 때의 검증률을 비교하기 위하여 FAR(False Acceptance Rate)를 고정시킨 상태에서 검증률을 비교하는 ROC(Receiver Operating Curves)를 이용하였다. ROC 결과는 다음 그림 7과 같이 제안한 알고리즘에 의한 정합방법이 유클리디안 거리를 적용했을 때보다 검증 성능이 향상되었음을 보여준다.

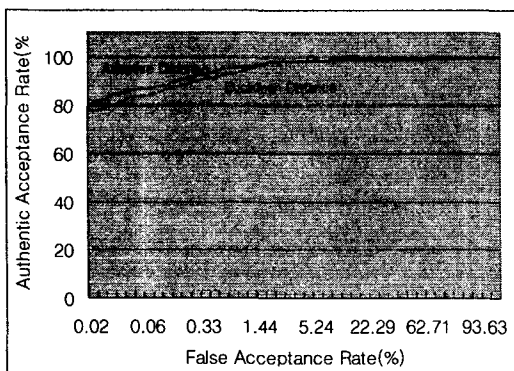


그림 7 Receiver Operating Curves

지문 100개를 무작위로 추출해서 정합한 평균 처리 시간은 유클리디안 거리를 사용했을 때 0.10초이고, 제안한 알고리즘은 0.11초로서 처리시간은 약 0.01초의 차이가 있었다.

V. 결론 및 향후 연구 계획

본 논문에서는 지문의 변형에 따른 특징점 사이의 거리변화에서 생기는 오차를 줄이기 위해 적응적 거리를 이용한 새로운 정합 알고리즘을 제안하였다. 실험 결과, 새로 제안한 방법은 유클리디안 거리를 사용했을 때의 성능보다 좋아짐을 확인할 수 있었고, 처리 시간에서도 비슷하여 전체적인 성능이 향상됐음을 알 수 있다. 실제적으로 지문의 변형에는 특징점 사이의 거리 변화뿐만 아니라 일치 중심점으로부터의 각도도 변하게 된다. 그러므로, 향후에는 지문의 변형에 따른 거리와 일치 중심점으로부터의 각도를 동시에 고려하여 오류를 줄이는 정합 방법에 대한 연구가 필요하다.

참고문헌

- [1] L. Hong, Y. Wan, and A. Jain, "Fingerprint Images Enhancement Algorithm and Performance Evaluation," IEEE Trans. on PAMI Vol.20, No.8, pp 777-789, 1998
- [2] A.K. Jain, R. Bolle and S. Pankanti, "BIOMETRICS: Personal Identification in Networked Society," Kluwer Academic Publishers Group, 1999
- [3] A.K. Jain, L. Hong and R. Bolle, "On-Line Fingerprint Verification," IEEE Trans. PAMI vol. 19, no. 4, pp. 302-314, 1997
- [4] A. K. Jain, L. Hong, S. pankanti, and R.Bolle, "An Identity-Authentication System Using Fingerprints," Proc. IEEE, vol.85, no. 9, pp. 1465-1388, Sept. 1997
- [5] A. K. Hrechak and J. A. Mchugh, "Automated fingerprint recognition using structural matching," Pattern Recognition, vol. 23, no. 8, pp. 893-904, 1990
- [6] A. Wahab, S. H. Chin and E. C. Tan, "Novel approach to automated fingerprint recognition," IEE Proc.-Vis. Image Signal Process., Vol. 145, no. 3, pp. 160-166, June 1998
- [7] X. Jiang, W. Y. Yau, "Fingerprint Minutiae Matching Based on the Local And Global Structures," IEEE 15th International Conference on Pattern Recognition, Vol. 2, pp. 1042-1045, Sep. 2000
- [8] X. Luo, J. Tian, Y. Wu, "A Minutia Matching Algorithm in Fingerprint Verification," IEEE 15th International Conference on Pattern Recognition, Vol. 4, pp. 833-836, Sep. 2000