

윤곽 특징과 LAT(Local Affine Transformation)를 이용한 필기체 숫자 인식

우경행 · 최원호
전자공학과

<요 약>

본 논문에서는 오프 라인 필기체 숫자 패턴에 대하여 LAT(local affine transformation)와 윤곽 특징을 이용한 패턴 정합에 관하여 연구하였다. LAT는 다른 정합 방법들보다 다소 시간이 많이 소요되지만 패턴의 크기 변화, 회전, 일그러짐, 이동 등의 변형을 흡수할 수 있고, 구조적 특징과의 결합이 용이하다. 가우시안 윈도우 함수의 분포를 줄이면서 LAT를 반복하여 수행함으로써 알고리즘이 수렴하게 되며, 최적 정합을 이루게 된다.

입력 패턴과 마스크 패턴의 윤곽 특징을 비교하여, 유사도가 큰 마스크 패턴들을 선별하여 LAT 알고리즘에 적용시켜서 LAT를 수행하는데 소요되는 시간을 줄였으며, 인식의 신뢰도를 향상시키기 위하여 LAT에 구조적 특징인 교차수와 주축을 LAT의 목적함수의 가중치로 첨가하였다.

Handwritten Numeric Character Recognition using the Profile Features and the Local Affine Transformation

Kyoung Haeng Woo · Won Ho Choi
Dept. of Electronic Engineering

<Abstract>

An off-line pattern matching method of handwritten numerals using the local affine transformation(LAT) and structural features is proposed in this paper. Even though the computation time of the LAT is somewhat longer than the other matching methods, the LAT is known as a suitable method to recognize handwritten characters

since it is invariant in scale, rotation, shearing and translation, and it is easily coupled to structural features, such as crossing counts, principal angles, etc. The experiment shows that the corresponding mask pattern is converged to its input pattern as the iterations of the LAT are continued.

A preclassification is added with the profile features in order to decrease the computing time. The principal angles and crossing counts are combined to the LAT as the weight of the objective function in order to improve the performances.

The algorithm is applied to the recognition of handwritten numeral images. The average recognition rate is 85.5%.

I. 서 론

문자 인식은 패턴 인식의 일종으로 사람이 문자를 읽는 과정을 컴퓨터에서 구현하려는 노력에서 시작되었다. 오늘날 정보의 양은 엄청나게 방대해졌으며, 컴퓨터의 이용도 나날이 증대되어 정보의 전산화가 필연적이 되었다. 뿐만 아니라 문서량 또한 나날이 증가하고 있는 추세다. 따라서 문서 자동 입력이 절실하게 되었다.

문자 인식은 문자 패턴 정보를 얻는 방법에 따라서 온라인 문자 인식과 오프라인 문자 인식으로 분류된다. 온라인 문자 인식은 사람이 전자펜등으로 쓴 문자를 인식하는 것으로, 획, 필순 등의 정보를 쉽게 얻을 수 있다. 반면 오프 라인 문자 인식은 스캐너나 카메라로 받아들인 문자 영상을 인식하는 것으로, 획, 필순 같은 사전 지식을 얻을 수 없으며 골격선 추출과 같은 전처리를 과정을 거쳐 문자 정보를 얻을 수 있다.

오프 라인 문자 인식은 인쇄체와 필기체 두 부분으로 이루어져 있다. 인쇄체는 기계로 찍은 문자로 일반적으로 일정한 형태를 가지므로 문자 변형이 적다. 그리고 인쇄체 인식에 대한 연구가 활발히 이루어져 현재 상품화되어 시판되고 있는 경우도 있다. 반면 필기체는 크기 변화, 일그러짐, 획의 굵김 등의 변형이 많고, 문자간의 구별에도 어려움이 많아서 연구가 아직은 초보적인 단계에 머물러 있다.

기존의 필기체 문자 인식에서는 원형 정합 방법, 확률 통계적 방법, 구조적 방법, 신경망을 이용한 방법 등을 사용하고 있다. 최근에는 신경망과 더불어 은닉 마르코프 모델이 오프 라인 필기체 문자 인식에 사용되고 있다[1].

필기체 문자를 인식하는 한 방법으로 패턴 정합이 있다. 패턴 정합은 입력 패턴을 미리 설정된 여러 개의 마스크 패턴들에 각각 정합 시켜서 최적의 정합을 얻은 마스크 패턴을 찾아서 입력 패턴을 인식하는 방법이다. 패턴 정합은 대표적으로 이완 정합(relaxation matching)[2]과 동적 프로그램 정합(dynamic programming matching)[3] 등이 있다. 동적 프로그램 정합은 문자 패턴의 구조적 특징과 결합하기 위해서는 복잡한 절차를 거쳐야 한다.

다른 방법들보다 다소 시간이 많이 소요되지만 Wakahara가 제안한 LAT (local affine transformation)를 이용한 정합 방법[4][5]은 구조적인 정보를 첨가하는 것이 용이하다. 또한 이 방법은 문자의 크기 변화, 회전, 일그러짐, 이동 등의 문자 변형을 충분히 수용할 수 있다.

본 논문에서는 LAT (local affine transformation) 정합 방법을 사용하여 숫자를 인식하였으며, 인식의 신뢰도를 높이기 위하여 구조적 특징을 LAT 정합 조건식의 가중치로 적용하였다. LAT 정합에 소요되는 시간을 줄이고, 입력과 유사한 패턴을 선별하기 위하여 패턴 윤곽의 두께 정보를 이용하였다. 입력과 유사한 윤곽 분포를 가지는 마스크 패턴들

을 미리 선별하여, 선별된 마스크 패턴들에 대해서만 입력 패턴과의 정합을 수행하였다.

시험 패턴은 20명이 한 숫자에 대해서 10번씩 쓴 2000개의 필기체 숫자 패턴을 사용하였다. 시뮬레이션은 DIGITAL DEC 3000 workstation으로 수행하였다.

II장에서는 LAT를 이용한 패턴정합 방법을 설명하고, III장에서는 구조적 특징과 윤곽 특징을 LAT에 부가하는 방법을 제시하였다. IV장에서는 시험 패턴에 제시한 방법을 적용하여 시뮬레이션 시킨 실험결과를 나타내었으며, 마지막으로 V장에서 결론을 맺었다.

II. LAT (Local Affine Transformation)를 이용한 패턴 정합 방법

2.1 LAT (local affine transformation)

M개의 흑화소로 구성되어 있는 마스크 패턴(mask pattern)을 R이라 하고 다음과 같이 나타낸다.

$$R = (R_1, R_2, \dots, R_i, \dots, R_M) \quad (2-1)$$

여기서, R_i 는 마스크 패턴의 i 번째 흑화소의 위치 벡터이다.

같은 방법으로, N개의 흑화소로 구성되어 있는 입력 패턴(input pattern)을 S라하고 다음과 같이 나타낸다.

$$S = (S_1, S_2, \dots, S_k, \dots, S_N) \quad (2-2)$$

여기서, S_k 는 입력 패턴의 k 번째 흑화소의 위치 벡터이다.

마스크 패턴 R과 입력 패턴 S에서 흑화소의 위치 이동은 각 부분 영역에서 선형적으로 변한다. 따라서 각 부분 영역에서 위치 이동은 affine transformation으로 나타낼 수 있다. 즉 입력 패턴의 각 부분 영역에서의 왜곡을 affine transformation으로 표현할 수 있다고 가정한다. 만약 이러한 가정이 타당하다면 두 패턴의 각각의 부분 영역에서 최적 정합을 이루는 local affine transformation (LAT)을 결정함으로써 패턴 정합을 성공적으로 수행할 수 있고, 또한 두 패턴에 적당한 윈도우 함수를 적용함으로써 LAT의 범위를 제한할 수 있다. 이것이 LAT를 이용한 패턴 정합의 기본 개념이다[4][5].

LAT를 이용한 단일 정합 과정은 다음의 5단계로 이루어져 있다.

◆ 단계 1

마스크 패턴의 i 번째 흑화소 R_i 에 대한 affine transformation은 다음과 같이 정의한다.

$$R'_i = A_i R_i + b_i \quad (1 \leq i \leq M) \quad (2-3)$$

여기서, A_i 는 회전, 크기 변화, 밀림 등을 나타내는 2×2 행렬이고, b_i 는 위치 변화를 나타내는 벡터이다.

$$A_i = (a_{pq}), \quad p, q = 1, 2$$

$$b_i = (b_x, b_y)^T \quad (2-4)$$

◆ 단계 2

변형의 부분적인 연속성을 보장하기 위하여 마스크 패턴 R_i 의 이웃 화소 R_j 에 같은 A_i 와 b_i 를 적용한다.

$$R_j = A_i R_j + b_i \quad (j \neq i) \quad (2-5)$$

여기서, A_i 는 회전, 크기 변화, 밀림 등을 나타내는 2×2 행렬이고, b_i 는 위치 변화를 나타내는 벡터이다.

◆ 단계 3

마스크 패턴 R_i 에 대해서 최적의 A_i 와 b_i 를 정하기 위해서 목적 함수를 도입한다. 목적 함수는

$$\begin{aligned} \psi_i &= \sum_j \omega_{ij} \sum_k \rho_{jk} \|R_j - S_k\|^2 \\ &= \sum_j \omega_{ij} \sum_k \rho_{jk} \|A_i R_j + b_i - S_k\|^2 \end{aligned} \quad (2-6)$$

이다. 여기서 $\|\cdot\|$ 은 Euclidean norm을 나타낸다. 목적함수가 최소가 될 때의 A_i 와 b_i 를 구하면 최적치가 된다. ω_{ij} 는 마스크 패턴의 화소 R_i 에 대해서 이웃 화소 R_j 가 목적 함수 ψ_i 에 미치는 영향을 조절하기 위한 가중치이다. ρ_{jk} 는 마스크 패턴의 화소 R_j 에 대해서 입력 패턴의 화소 S_k 가 목적 함수 ψ_i 에 미치는 영향을 조절하기 위한 가중치이다. 이 가중치들은 두 화소의 거리가 멀어질수록 줄어들어야 하며, 이를 만족하는 함수로 가우시안 윈도우 함수(Gaussian window function)를 적용한다. 가중치 ω_{ij} 와 ρ_{jk} 는

$$\begin{aligned} \omega_{ij} &= \exp(-\|R_i - R_j\|^2 / \theta_s^2) \\ \rho_{jk} &= \exp(-\|R_j - S_k\|^2 / \theta_d^2) \end{aligned} \quad (2-7)$$

이다. 여기서, θ_s 와 θ_d 는 가우시안 윈도우 함수의 분포를 나타내는 윈도우 매개 변수(window parameter)이다. θ_s 와 θ_d 는 다음 식과 같은 관계를 가지고 있다.

$$\theta_d = \alpha \cdot \theta_s, \quad 0.0 < \alpha < 1.0 \quad (2-8)$$

여기서, α 는 비례 상수로 0과 1사이의 실수 값이다.

◆ 단계 4

목적 함수는 A_i 와 b_i 의 각각의 원소에 대해서 편미분한 값이 0이 될 때 최소 값이 된다.

$$\begin{aligned} \frac{\partial \Psi_i}{\partial a_{pq}} &= 0, \quad (p, q = 1, 2) \\ \frac{\partial \Psi_i}{\partial b_i} &= 0 \end{aligned} \tag{2-9}$$

식(2-9)의 해를 구함으로써 목적 함수 값이 최소일 때의 A_i 와 b_i 를 얻을 수 있다. 이 값이 최적의 A_i 와 b_i 값이다.

◆ 단계 5

단계 4에서 구해진 A_i 와 b_i 를 마스크 패턴에 적용함으로써 마스크 패턴을 변형시킨다.

$$R'_i = A_i R_i + b_i \quad (1 \leq i \leq M) \tag{2-10}$$

LAT의 한 예를 그림 1에 나타내었다. 그림 1에서 \oplus 은 마스크 패턴의 화소, \circ 은 입력 패턴의 화소, \bullet 은 일치되는 화소를 나타낸다.

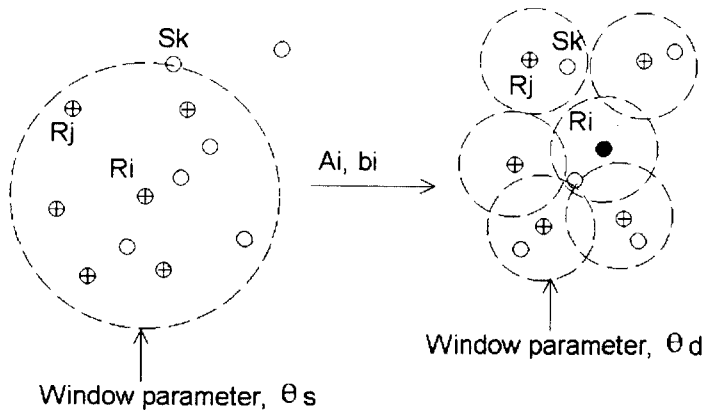


그림 1. LAT의 예
Fig 1. An example of the LAT

2.2 반복 LAT 과정

패턴 정합의 최종 목적은 입력 패턴과 이에 대응하는 마스크 패턴을 점-대-점 대응이 이루어지도록 하는 것이다. 최적의 정합을 만들기 위해서 윈도우 매개변수의 크기를 줄여서 궁극적으로 점-대-점 대응이 되도록 한다. 또한 알고리즘을 반복 수행하면 점차 알고리즘이 수렴하는 성질을 가진다.

윈도우 매개변수의 크기를 줄이면서 LAT를 반복 수행하여 입력 패턴과 마스크 패턴이 최적의 정합을 이루도록 유도하고, 알고리즘이 점차 수렴하도록 만든다. 반복 알고리즘에

서 윈도우 매개변수의 크기는 다음과 같이 줄인다.

$$\begin{aligned}\theta_s(t) &= \theta_s(0)/\sqrt{t} \\ \theta_d(t) &= \alpha \cdot \theta_s(t), \quad \alpha \in (0.0, 1.0)\end{aligned}\tag{2-11}$$

여기서, t 는 반복 횟수로 $t = 1, 2, 3, \dots$ 이고, $\theta_s(0)$ 는 초기 매개변수 값이다. 각 반복 단계의 목적함수에 식(2-11)의 매개변수를 적용시킨다.

t 번째 반복 이후 마스크 패턴과 입력 패턴 사이의 유사도를 측정하기 위하여 비유사도(dissimilarity) $D(t)$ 를 측정한다. 비유사도 $D(t)$ 는

$$D(t) = \frac{1}{2} \left\{ \frac{1}{M} \sum_i \min_k \|R_i(t) - S_k\|^2 + \frac{1}{N} \sum_k \min_i \|R_i(t) - S_k\|^2 \right\}\tag{2-12}$$

이다. 반복횟수가 증가할수록 $D(t)$ 는 점점 감소한다. 즉, 반복횟수가 증가할수록 마스크 패턴이 점차로 입력 패턴으로 변형되어 간다.

III. 구조적 특징과 윤곽(profile) 특징

3.1 구조적 특징

입력과 마스크 패턴에 구조적인 정보를 활용하여 가중치를 부여함으로써 입력과 마스크 패턴 사이의 유사도가 커지도록 해준다. 본 논문에서는 교차수(crossing count)와 주축(principal angle)을 구조적 특징으로 사용하였다.

형태학적 특징인 교차수는 각 흑화소의 인접 화소들을 반시계 방향으로 회전하면서 구한다. 임의의 흑화소 p 의 교차수를 구하기 위한 3×3 마스크를 그림 2에 나타내었다. 교차수는 각 흑화소의 연결성을 나타내 준다. 흑화소 p 의 교차수 $\beta(p)$ 는 다음과 같이 구한다.

$$\beta(p) = \sum_i |e_{i+1} - e_i|\tag{3-1}$$

여기서 e_i 는 0과 1 둘 중 하나의 값을 갖는다. $\beta(p)$ 가 0이면 고립점이고, $\beta(p)$ 가 2면 끝점이다. $\beta(p)$ 가 4이면 연결점이고, 6 또는 8값을 가지면 교차점을 나타낸다. 따라서 교차수가 패턴의 형태학적 특징을 나타냄을 알 수 있다.

e ₃	e ₂	e ₁
e ₄	p	e ₀
e ₅	e ₆	e ₇

그림 2. 교차수를 위한 3 × 3 마스크
 Fig 2. A 3 × 3 mask for the crossing count

각 흑화소 p에서의 주축(principal angle)은 그림 3에 나타낸 4 방향 중 하나로 나타낸다. 즉 수평(H), 오른쪽 대각(RD), 수직(V), 왼쪽 대각(LD)으로 나타낸다. 이때, 주축은 $\Gamma(p)$ 로 나타내고, $\Gamma(p)$ 는 수평, 오른쪽 대각, 수직, 왼쪽 대각에 대해서 각각 0, 1, 2, 3의 값을 갖는다.

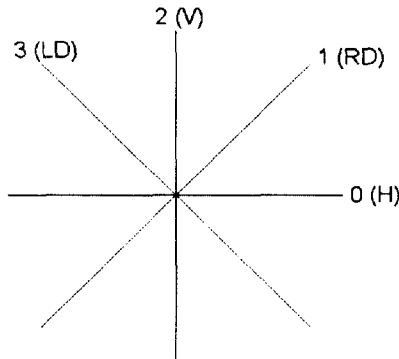


그림 3. 주축의 4방향 코드
 Fig 3. The four-directional code to detect principal angles

주축은 골격선을 추출한 패턴에 대해서 고유 벡터 선 집합(eigenvector line fitting)[6] 방법을 이용하여 구한다. 본 논문에서는 5 × 5 이웃 화소에 대해서 수행하였다.

교차수 $\beta(p)$ 와 주축 $\Gamma(p)$ 를 LAT의 목적함수에 첨가하여 유사도에 대한 가중치를 준다.

$$\Psi_i = \sum_j \omega_{ij} \sum_k \rho_{jk} \xi_{jk} \zeta_{jk} \| A_i R_j + b_i - S_k \|^2 \tag{3-2}$$

$$\zeta_{jk} = 1, \quad \text{for } \alpha(R_j) = \alpha(S_k)$$

$$= 0, \quad \text{for } \alpha(R_j) \neq \alpha(S_k)$$

ξ_{jk} : 주축에 대한 유사도의 측도

$\Gamma(R_j)$ 와 $\Gamma(S_k)$ 의 차가 0이면 ξ_{jk} 는 1 값을 갖고, $\Gamma(R_j)$ 와 $\Gamma(S_k)$ 의 차가 1이면 ξ_{jk} 는 0.5 값을 갖게 되며, 그 차가 2 이상이면 ξ_{jk} 는 0 값을 갖는다. R_j 와 S_k 의 교차수가 같을 경우, 즉 형태학적인 구조가 같은 경우의 화소에 대해서만 목적함수를 구하며, 입력과 마스크 패턴의 주축이 같은 경우에 큰 가중치를 준다.

3.2 윤곽 특징 (Profile Features)

필기체 숫자는 다양한 형태를 지니고 있다. 여러 형태를 하나의 마스크가 모두 수용할 수는 없으며 여러 개의 마스크 패턴이 필요하다. 그러나 마스크의 갯수가 많아지면 그만큼 계산량도 늘어나게 된다. 이러한 문제를 해결하기 위하여 LAT를 이용한 패턴 정합을 수행하기 전에 마스크 패턴들을 입력 패턴의 형태와 비교하여 60% 정도의 마스크 패턴을 먼저 선정한다. 선정된 마스크 패턴에 대해서만 LAT 알고리즘을 수행하여 인식하도록 한다.

사전에 마스크 패턴을 선정하기 위하여 본 논문에서는 윤곽 특징[7][10]을 이용하였다. 입력 패턴과 마스크 패턴들의 윤곽 모양을 고려하여 유사한 패턴들을 추출한다.

각 마스크 패턴들에 대해서 입력 패턴의 수평 방향으로 투영한 값의 분포와 비슷한 분포를 가지는 패턴들을 마스크 패턴 갯수의 60% 정도를 추출한다. 수평 방향으로의 투영을 그림 4에 나타내었다. 즉, x_1 의 분포와 x_2 의 분포를 비교하여 입력과 마스크 패턴의 형태 유사도를 측정한다.

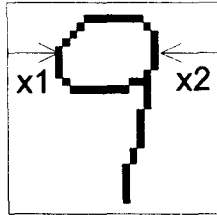


그림 4. 수평 방향의 투영

Fig 4. The left and right profile to detect thickness

IV. 실험 및 결과

4.1 실험방법

필기체 숫자를 인식하기 위하여 LAT를 이용하여 마스크 패턴을 변형시켜 입력과 정합시키는 방법을 사용하였다. 인식의 신뢰도를 향상시키기 위하여 구조적인 특징인 교차수와 주축을 최적 이동 범위의 조건식의 가중치로 사용하였다. 수행 시간을 줄이고 불필요한 계산을 줄이기 위해서 모든 마스크 패턴을 사용하지 않고, 윤곽 특징을 이용하여 마스크와 입력 패턴의 윤곽, 즉 두 패턴의 두께 정보를 비교하여 입력과 비슷한 형태를 가진 마스크 패턴을 60%정도 미리 선별하여 사용하였다.

입력 패턴은 20명이 아라비아 숫자를 5번씩 쓴 1000자의 필기체 숫자를 사용하였다. 입력 패턴은 부록에 나타내었으며 패턴들을 살펴보면 다양한 크기의 여러 형태를 가지고 있음을 알 수 있다. 먼저 크기가 다른 패턴들을 일정한 비율로 축소하여 필기체 숫자를 16 × 16 화소로 만든다. 16 × 16 크기의 입력들에 대하여 세션화[8][9]를 수행하고 8연결을 형성하도록 전처리 과정을 거쳐 숫자들의 골격선을 추출한다. 세션화 과정을 거친 필기체 숫자 패턴의 예를 그림 5에 나타내었다.

입력 패턴의 각 화소의 빈도 수를 토대로 0과 1은 한 개씩, 2부터 9까지는 2개씩, 18개의 마스크 패턴을 만들어 사용하였다. 그림 6에 사용한 마스크 숫자 패턴을 나타내었다.

프로그램은 C-언어로 작성하고, DIGITAL DEC 3000 Workstation에서 시뮬레이션 하였다.

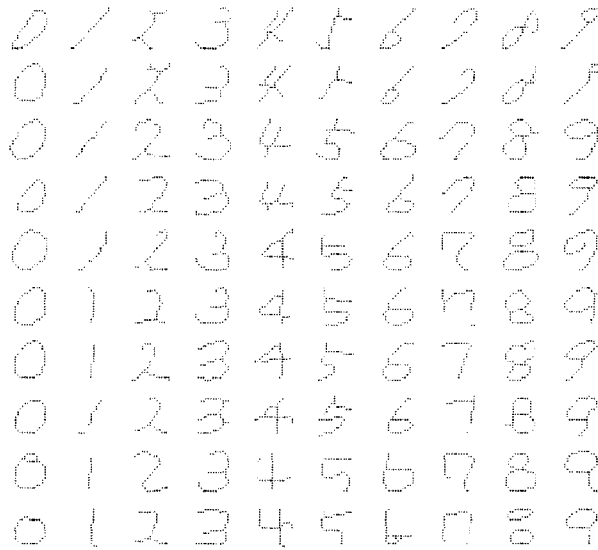


그림 5. 정규화 후 세션화한 필기체 숫자의 예
 Fig. 5 Examples of normalized and thinned handwritten numerals

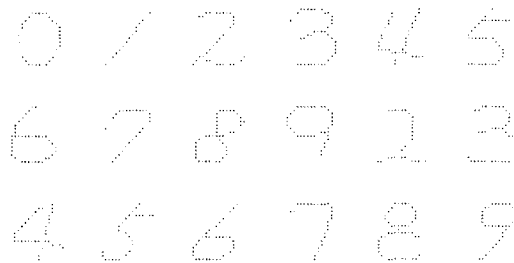


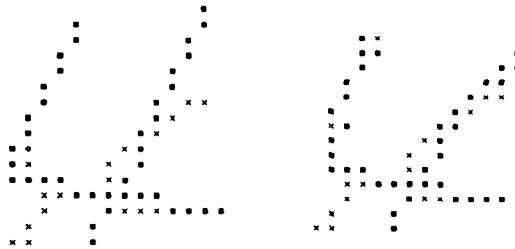
그림 6. 마스크 패턴
 Fig 6. Mask patterns

입력 패턴에 대해서 선택된 마스크 패턴들을 LAT를 이용하여 변형시키고 입력과 마스크 패턴의 유사도를 측정한다. 입력 패턴과 마스크 패턴이 비유사도가 일정 값보다 작은 경우 서로 정합이 이루어졌다고 보고 마스크의 변형을 중지한다. 입력과 마스크 패턴의 정합이 이루어지지 않은 경우, 즉 비유사도가 큰 값을 가지는 경우는 윈도우 매개 변수의 값을 줄여서 LAT를 반복 수행하여 마스크 패턴을 변형시킨다.

만약 비유사도를 측정하였을 때 첫 번째 최소치가 두 번째 최소치와 근소한 차이가 난다면 불량 패턴으로 분류한다. 제대로 인식된 패턴, 불량 패턴, 다른 패턴으로 대치되어 오인식된 패턴의 세 부류로 분류된다.

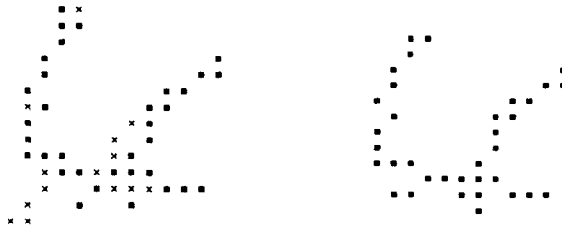
4.2 실험 결과

필기체 숫자 패턴에 대하여 구조적 정보를 가중치로 첨가하여 LAT를 반복 수행하여 마스크 패턴을 변형시켜 입력 패턴과 정합시켰다. 실험에서 윈도우 파라미터 θ_s 는 5로 하고, α 는 0.3으로 적용하였다. 따라서 θ_d 는 3 값을 가진다. 그림 8과 9는 숫자 4와 6에 대하여 앞에 제시한 값을 가지고 LAT 알고리즘을 반복 수행한 결과이다. 그림 7, 8에서 검은 점은 마스크 패턴의 화소를 나타내고, 'x' 표시된 점은 입력 패턴의 화소를 나타낸다. 그림 7의 (a)와 (b)를 비교하여 볼 때 마스크 패턴이 점점 입력 패턴으로 변형되어 가는 것을 알 수 있다. 그리고 반복 횟수가 증가할수록 유사도가 증가함을 알 수 있다.



(a)

(b)

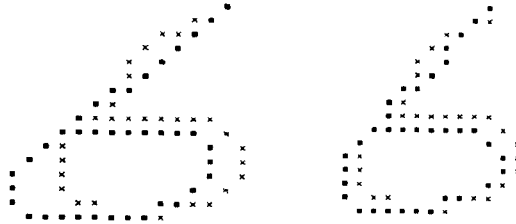


(c)

(d)

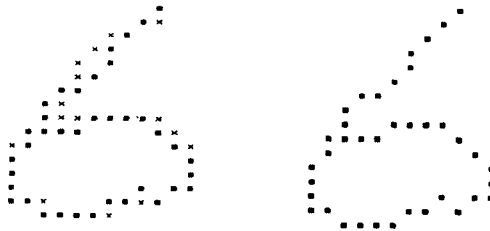
그림 7. 숫자 4에 대한 LAT 결과 : (a) 반복이 없는 경우
 (b) 반복 횟수 1 (c) 반복 횟수 4 (d) 반복 횟수 10

Fig 7. The LAT results of the numeral 4 : (a) No iteration,
 (b) # of iterations=1, (c) # of iteration=4 (d) # of iteration=10.



(a)

(b)



(c)

(d)

그림 8. 숫자 6에 대한 LAT 결과 : (a) 반복이 없는 경우
 (b) 반복 횟수 1 (c) 반복 횟수 4 (d) 반복 횟수 10

Fig 8. The LAT result of the numeral 3 : (a) No iteration,
 (b) # of iteration=1 (c) # of iteration=4 (d) # of iteration=10.

표1은 1000개의 입력 패턴에 대하여 윈도우 매개 변수 θ_s 를 5, α 를 0.3으로 하여 LAT를 한번 수행한 후의 인식 결과이다. 정인식률이 84.3%이고 오인식률이 10.3%이고, 비슷한 유사도를 가져서 제거되는 비율이 6.6%이다.

표1. 인식결과 ($\theta_s = 5, \alpha = 0.3$)Table 1. The result of recognition ($\theta_s = 5, \alpha = 0.3$)

숫자	정인식	오인식	제거
0	98	1	1
1	100	0	0
2	85	11	4
3	89	5	6
4	75	18	7
5	69	22	9
6	79	8	13
7	75	19	6
8	81	9	10
9	82	8	10

표2는 1000개의 입력 패턴에 대하여 윈도우 매개 변수 θ_s 를 8, α 를 0.3으로 하여 LAT를 한번 반복한 후의 인식 결과이다. 정인식률이 84.9%이고 오인식률이 12.0%이고, 비슷한 유사도를 가져서 제거되는 비율이 3.1%이다.

표2. 인식결과 ($\theta_s = 8, \alpha = 0.3$)Table 2. The result of recognition ($\theta_s = 8, \alpha = 0.3$)

숫자	정인식	오인식	제거
0	100	0	0
1	100	0	0
2	87	13	0
3	87	12	1
4	79	17	4
5	74	23	3
6	84	12	4
7	80	17	3
8	77	16	7
9	81	10	9

표1과 표2를 비교하여 보면 윈도우 매개 변수의 값에 따라 조금씩 다른 인식 결과를 나타내고 있음을 알 수 있다. 매개 변수 값을 어떻게 선택하느냐에 따라서 각각 다른 인식률을 나타낸다. 현재로서는 실험 결과를 토대로 선택하여야 한다. 표1과 표2의 결과로 볼 때 θ_s 는 5로 하고 α 는 0.3으로 하는 경우가 정인식률이 낮은 반면 오인식률이 적게 나타나고 있다.

숫자별로 비교해 보면 0과 1은 거의 100%의 인식률을 보여준 반면 4, 5, 7은 대체로 낮은 인식률을 보여주고 있다. 4의 경우는 0으로 오인식되는 경우가 있었고, 5는 6으로 오인식되는 경우가 발생하였다. 7은 9 또는 2로 오인식되는 경우가 발생하였다. 숫자의 과도한 변형을 막지 못한 때문으로 생각된다.

V. 결 론

본 논문은 다양한 형태를 지닌 필기체 숫자의 변형을 흡수하여 인식할 수 있는 local affine transformation을 토대로 인식의 신뢰도를 향상시키기 위하여 구조적인 특징인 교차수와 주축을 LAT 목적 함수의 가중치로 적용하여 필기체 숫자 인식을 수행하였다. 알고리즘 수행 과정에서 마스크 패턴을 입력 패턴의 형태로 변형시켰다. 즉 마스크 패턴의 각 화소들을 이웃화소들의 분포를 고려하여 입력 패턴과의 가장 잘 정합될 수 있는 지점으로 이동시켰다. 모든 마스크 패턴을 입력 패턴에 대해 최소-제곱 조건을 통해 변형시켜서 입력 패턴과의 거리가 최소가 되는 마스크 패턴을 입력 패턴의 부류로 인식하였다.

필기체 숫자는 다양한 형태를 지니고 있어서 한가지의 마스크로 모두 수용할 수는 없다. 그래서 여러 개의 마스크를 사용하여 필기체 숫자의 변형을 최대한 수용하도록 하였다. 본 논문에서는 0과 1을 제외한 숫자들은 2 종류의 마스크 패턴을 사용하였다. 마스크의 갯수가 많아지면 그만큼 계산량도 늘어나게 된다. 다양한 형태의 숫자들을 수용하고 계산량도 줄이기 위하여 마스크 패턴들을 입력 패턴과의 형태를 비교하여 60% 정도의 마스크를 먼저 선정하고, 선택된 마스크 패턴에 대해서만 LAT 알고리즘을 수행하여 입력 패턴을 인식하였다.

윈도우 매개 변수의 값에 따라 서로 다른 인식률을 보였다. 따라서 적절한 윈도우 매개 변수 설정이 필요하며, 윈도우 매개 변수를 입력 패턴에 따라서 자동으로 변화시켜줄 수 있는 알고리즘이 필요하리라 본다. 입력 패턴들을 어느 정도 비슷한 크기로 되어 있는 것을 사용한다면 보다 나은 결과를 얻을 수 있으리라 본다.

M개의 흑화소를 가지는 마스크 패턴과 N개의 흑화소를 가지는 입력 패턴에 대해서 LAT를 한번 수행하는데 $M \times (M-1) \times N$ 번의 계산량이 필요하다. 이러한 계산을 병렬로 처리할 수 있는 하드웨어로 구현한다면 실시간 처리가 가능하리라고 본다. 실시간 처리가 가능해지면 우편 번호 인식, 금융 기관의 매출 전표 인식 등 사무 자동화에 응용할 수 있으리라 본다.

참 고 문 헌

1. S. Mori, C. Y. Suen and K. Yamamoto, "Historical Review of OCR Research and Development," Proceedings of the IEEE, Vol.80, No.7, pp. 1029-1058, July 1992.
2. S. Ranada and A. Rosenfeld, "Point Pattern Matching by Relaxation," Pattern Recognition, Vol.12, pp. 269-275, 1980.
3. A. A. Amiri, T. E. Weymouth and R. C. Jain, "Using Dynamic Programming for Solving Variational Problems in Vision," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.12, No.9, pp. 855-867, 1990.
4. T. Wakahara, "Dot Image Matching Using Local Affine Transformation," in Proc. 10th Int. Conf. Pattern Recongnit., pp. 837-841, 1990.
5. T. Wakahara, "Shape Matching Using LAT and its Application to Handprinted Numeral Recognition," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.16, No.6, pp. 618-629, Jun 1994.
6. R. O. Duda and P. E. Hart, Pattern Classification and Scene Analysis New York: Wiley, 1973.
7. S. Mori, K. Yamamoto and M. Yasuda, "Research on Machine Recognition of Handprinted Characters," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.6, No.4, pp. 386-405, July 1984.
8. R. C. Gonzalez and R. E. Woods, Digital Image Processing, Addison Wesley, 1992.
9. N. J. Naccache and R. Shinghal, "SPTA: A Proposed Algorithm for Thinning Binary Patterns," IEEE Trans. Systems, Man, and Cybernetics, Vol. 14, no. 3, pp. 409-418, 1984.
10. F. Kimura and M. Shridhar, "Handwritten Numeral Recognition Based on Multiple Algorithms," Pattern Reckon. Vol.24, No.10, pp. 969-983, 1991.