

# PCA를 이용한 온라인 문자인식 기법

유재만<sup>†</sup>, 한정훈<sup>\*\*</sup>, 김우생<sup>\*\*\*</sup>

## 요 약

온라인 문자 인식 기술은 PDA, 태블릿 PC 등 많은 새로운 응용에서 사용되고 있으나, 인식 기술은 아직 이러한 첨단 도구들을 자연스럽게 이용하기에는 못 미치는 실정이다. 또한 최근 많이 사용되는 은닉 마르코프 모델(HMM)은 입력패턴을 전체 표준패턴과 비교함으로써 많은 기억장소와 계산량을 필요로 하는 단점을 지니고 있다. 따라서 본 논문에서는 더욱 효율적으로 온라인 문자 인식을 가능하게 하는 방법을 제안한다. 본 연구에서는 전처리 단계를 거쳐 학습 데이터와 인식 데이터의 체인코드를 생성하고, 인식 단계에서 입력 데이터에 주성분 분석(PCA)기법을 적용하여 데이터의 차원을 줄여 문자를 인식한다. 제안하는 방법의 타당성은 실험을 통해서 검증한다.

## Online Character Recognition Technique Using PCA

Jaeman Yoo<sup>†</sup>, Jeonghoon Han<sup>\*\*</sup>, Woosaeng Kim<sup>\*\*\*</sup>

## ABSTRACT

Online character recognition techniques have been applied in many new fields of PDA, Tablet PC etc. But the recognition techniques can not use such high technologies naturally yet. Hidden Markov Model (HMM) that is much used recently requires high memory space and complex computational tasks because of comparing the input data with entire standard patterns. In this paper we propose a method to recognize the online characters more efficiently. At first we create chain-codes of learning data and recognition data in preprocessing phase, and then we compress dimensions of data using Principal Component Analysis (PCA) and recognize a character compressed data in recognition phrase. Validity of proposed method is verified by experiment results.

**Key words:** Online Character Recognition(온라인 문자인식), PCA(PCA)

## 1. 서 론

전자 펜은 휴대성과 편의성의 장점을 지녀 PDA, palm PC, pocket PC, 스마트 폰, 태블릿 PC 등의 많은 응용에 사용될 수 있기에 높은 인식률의 온라인 문자 인식 방법이 더욱 필요하다[1]. 또한 많은 응용들이 소형화, 휴대화 되는 추세에서 인식률과 함께 기억장소와 계산량에서 효율적인 방법이 필요하다.

온라인 문자인식은 필기 데이터의 획득시 필기의 시간적, 공간적인 동적 정보를 얻을 수 있으며 획수, 획순, 각 획에 대한 필기 방향과 각 획 내에서의 필기 속도 등의 정보를 얻어 이를 인식 알고리즘에 적용한다. 이런 동적 정보는 인식 알고리즘에 유용하기도 하지만 오히려 인식 알고리즘을 복잡하게 만들기도 하므로 적절한 동적 정보를 얻고 그에 맞는 알고리즘을 선택하는 것이 중요하다. 온라인 문자 인식의 장

※ 교신저자(Corresponding Author) : 한정훈, 주소 : 서울특별시 노원구 월계동 447-1번지(139-701), 전화 : 02)940-5217, FAX : 02)909-0998, E-mail : sopure@paranr.com  
접수일 : 2005년 2월 23일, 완료일 : 2005년 11월 22일

<sup>†</sup> 정희원, 광운대학교 컴퓨터과학과  
(E-mail : sopure@paran.com)

<sup>\*\*</sup> 정희원, 광운대학교 컴퓨터과학과 공학 석사  
(E-mail : semyo03@daritechs.com)

<sup>\*\*\*</sup> 정희원, 광운대학교 컴퓨터 공학부 교수  
(E-mail : kwsrain@gmail.com)

※ 본 논문은 2005년도 광운대학교 교내 학술연구비 지원에 의해 연구되었음

점은 필기자와 컴퓨터가 서로 상호 대화가 가능하다는 장점을 가지고 있다[2].

온라인 필기를 인식하는 방법은 크게 전처리, 정합의 두 단계로 나누어진다. 전처리 단계는 입력 데이터 내의 잡음을 제거하고, 입력 데이터를 특징 추출에 적합한 형태로 변환하여 입력 데이터의 변형으로 인한 오류를 최소화 하며, 전처리를 거친 데이터로부터 펜의 이동 자취나 위치 관계 등을 이용하여 문자에 포함된 정보를 유지할 수 있는 특징들을 추출한다. 정합 단계는 입력 문자에서 추출한 특징과 각 모델의 특징을 비교하여 가장 유사한 모델로 문자를 인식하는 단계이다[2]. 이러한 정합 연구들에 쓰인 방법들 중에서 은닉 마르코프 모델은 음성 인식 분야 뿐만 아니라 문자 인식에서도 성공적으로 적용되고 있다[3].

온라인 문자 인식 단계에서의 최종적인 목표는 확순과 확수에 무관하게 흘러 쓴 문자 인식 알고리즘을 개발하는 것인데, 동일한 문자일지라도 필기자 고유의 특성에 따라 확순 및 확순의 변형이 다양하게 나타나게 되므로 이런 변형을 흡수할 수 있는 알고리즘을 개발하는 것은 쉬운 일이 아니다[1]. 특히 이런 연구 과정에서 직면하는 문제점은 필기자의 개인차에 의해 나타나는 모든 가능한 패턴의 변형을 흡수할 수 있는 특징의 파악 및 구조 해석이 어렵다는 점이다. 따라서 이러한 개인차에 의해 나타나는 변형을 극복하기 위한 방안으로 각 문자 패턴에 대하여 필기자에 따른 다양한 필기 패턴들을 모델로 생성하거나, 은닉 마르코프 모델(HMM)처럼 탄력 정합(elastic matching)과 동적 프로그래밍 정합(dynamic programming matching)을 이용하여 입력 패턴에 대해 모든 표준 패턴을 정합하는 방법론이 사용되고 있다[2]. 이 방법론은 표준 패턴과의 정합에 요구하는 시간이 표준 패턴의 수에 비례하게 되므로 문자 집합이 많은 경우에는 실용적으로 이용하기에 단점이 있다. 또한, 위에 제시된 여러 방법들은 많은 기억장소와 계산량을 필요로 하는 단점을 지니고 있다[2].

본 논문에서는 온라인 필기를 인식할 때, 전처리 단계를 통해 입력 데이터를 특징 추출에 적합한 8방향 체인코드열의 상태로 변환한 후, 정합 단계에서 주성분 분석 기법(Principal Component Analysis)[4]을 적용하여 데이터의 차원을 줄여 인식하는 방법을 제안함으로써 입력 패턴과 모든 표준 패턴의 정합

시 계산량을 줄이는 효율적인 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 연구 배경에 대하여 소개하고, 3장에서는 전처리 단계를 통해 각 문자의 특징을 나타내는 체인코드들을 얻고 이를 이용하여 eigenvector를 생성하는 방법과 생성된 eigenvector를 이용하여 입력 데이터의 차원을 줄이는 방법과 문자를 인식하는 방법에 대해서 설명한다. 4장에서는 실험 및 결과를 보이며, 마지막으로 5장에서는 결론 및 추가 연구에 대하여 언급한다.

## 2. 연구 배경

본 장에서는 현재까지 문자 인식에 관한 연구들이 어떻게 이루어져 왔는가에 관한 흐름에 대해서 살펴보고자 한다. 문자 인식 시스템은 입력 필기 데이터와 그 시스템이 가진 정보, 즉 유한개의 모델을 적절한 기준에 따라 비교하고, 판단을 내리는 구조를 가지고 있다. 문자 인식은 전처리-정합 단계로 이루어져 있는데, 전처리는 특징 추출이 보다 정확하게 이루어 질 수 있도록, 원래의 데이터를 수정하는 것이고, 정합은 전처리를 거친 데이터의 특징을 추출하는 과정이다. 정합단계에서 추출된 특징을 이용하여 문자 인식에 사용하게 된다.

문자 인식시에는 수많은 변형을 갖는 각각의 문자들을 인식하여야 한다. 이러한 변형을 크게 두 가지로 나누면, 동일한 필기자의 필기 상태에 따라 달라지는 글씨의 동적 변형과 각각의 필기자 간에 나타나는 차이인 정적 변형이 있다[2]. 이러한 문자들을 인식하기 위한 여러 가지 방법이 제안되어 왔다. 그 중에서 최근 많이 사용되는 확률통계적 인식 방법인 은닉 마르코프 모델을 포함한 여러 방법에 대해 알아본다.

확률통계적 인식 방법은 인식의 결정 함수값이 통계에 근거한 확률로 주어지는 것이다. 이러한 방법은 주어진 패턴이 본질적으로 많은 변형을 가질 경우 이러한 변형을 통계적으로 수용하고자 하는 방법이다. 필기는 점과 점의 연결로 이루어진다고 볼 수 있다. 점들은 무작위로 배치되는 것이 아니라 펜의 움직임 즉 방향의 제약을 받도록 배치된다. 따라서 적절한 통계 모델을 생성한다면, 확률적인 예측이 가능하게 된다. 여기서 사용되는 통계 모델은 여러 가지

확률적 요소를 이용하여, 패턴의 특징을 표현하는 방법이다. 확률통계적 인식 방법으로서 가장 널리 알려져 있고, 문자, 음성 인식 분야에서 성공적으로 적용되고 있는 것이 은닉 마르코프 모델(HMM)이다. 은닉 마르코프 모델은 시간에 따른 순차적인 정보를 모델링하는데 뛰어난 장점을 갖고 있어 기존의 동적 프로그래밍 정합을 이용한 방법 보다 더 좋은 결과를 보여주고 있다[3].

곡선 정합 방법은 신호처리에서 주로 사용하는 방법으로 필기 인식의 경우 흘림체의 인식에 주로 사용된다. 입력된 미지의 획은 이미 저장되어 있는 획들의 모델들과 비교된 후에 가장 근사한 것이 선택된다. 획을 이루는 각 점들의 체인 코드를 사용하거나 [5], 획의 각 점들에 대한 Fourier계수를 구하여 일대일 대응을 조사하는 방법인데, 실제로 이러한 일대일 대응은 찾을 수 없기 때문에 탄력 정합 방법을 사용하기도 한다[6].

신경망을 이용한 방법은 인간의 두뇌 조직을 모델링한 것으로, 문자 인식을 위하여 단순한 기능을 수행하는 처리기들이 대규모 상호 연결된 병렬 분산 네트워크를 이용하는 방법이다. 신경망을 이용한 문자 인식은 학습에 의해 새로운 패턴에 적용할 수 있으며, 패턴의 국부적인 변형 및 잡영에 민감하지 않다는 장점을 갖는 반면, 패턴의 크기가 큰 경우 학습에 걸리는 시간이 매우 길어지며, 인식 후보 대상의 개수가 많은 경우 성능이 저하된다는 단점을 갖는다[2].

위의 방법들은 개인차에 의해 나타나는 변형을 극복하기 위해 각 문자 패턴에 대하여 필기자의 다양한 필기 패턴들을 모델로 생성하고, 입력패턴에 대해 모든 표준패턴과 정합하게 되므로 많은 기억장소와 계산량을 필요로 한다는 문제점을 가지고 있다. 따라서 본 논문에서는 전처리 단계를 통해 입력 데이터를 적절하게 처리한 후, 인식 단계에서 주성분 분석 기법(Principal Component Analysis)을 적용하여[7] 데이터의 차원을 줄여 인식하는 방법을 제안함으로써 문자 인식시 계산량을 줄이는 효율적인 방법을 제안한다.

### 3. PCA를 이용한 문자 인식 시스템

본 장에서는 PCA를 이용한 온라인 문자 인식 시스템의 구조와 사용된 인식 기술에 대해서 설명한다.

전체 시스템의 구조는 그림 1과 같다. 전체 시스템은 크게 학습 단계와 인식 단계로 나누어 지고, 각 단계는 전처리와 정합 과정을 거치게 된다. 전처리 과정에서는 타블릿과 같은 온라인 문자 입력기를 통해 입력 받은 학습을 위한 문자 데이터와 인식을 위한 문자 데이터내의 잡영을 제거하고, 특징 추출에 적합한 형태로 변환하여 안정되고 일관성 있는 특징을 체인코드 형태로 추출할 수 있게 한다. 정합과정에서는 구해진 특징들로 구성된 학습 데이터들의 집합에 주성분 분석 기법(Principal Component Analysis)를 적용하여 기저벡터인 eigenvector를 구한 후, eigenvector를 이용하여 학습 데이터와 인식 데이터의 유사도 측정에 사용될 각 성분(weighting)값들을 구한다. 위의 과정을 거쳐서 나온 성분값은 학습 단계에서 문자별로 군집화를 하게 되고, 이때 같은 문자끼리는 특징 벡터 공간 상에서 유사한 위치를 점하게 되기 때문에 각 문자를 대표하는 평균 성분값을 구할 수 있게 된다. 인식 단계에서 인식 문자의 성분값을 구하여 각 문자의 평균 성분값과의 거리차를 구해 문자를 인식하게 된다.

#### 3.1 전처리 및 코드화 과정

문자의 인식 데이터나 학습 데이터는 가장 먼저 전처리 과정을 거친다. 본 논문에서 사용한 전처리 과정은 잡영 제거 과정과 정규화 과정이다[2]. 전자

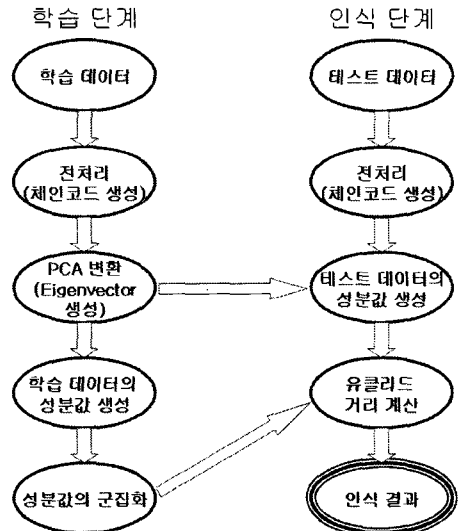


그림 1. 시스템의 구조

펜으로 타블릿 위에 쓴 글씨는 일반적으로 전자 기계적인 결합이나, 부정확한 손의 움직임 때문에 발생하는 잡음을 포함한다. 이러한 잡음을 가능한 줄이기 위해 빼침제거, 평활화, 여과를 통해 잡음을 제거하였다[2][8]. 또한 다양한 크기로 입력된 필기를 인식에 알맞은 일정한 크기로 바꾸어 주고, 문자의 쓰여진 위치의 변형에 무관하게 특징을 추출하기 위해서 정규화 하였다[8].

전처리 과정을 거친 입력에 대하여 코드화 과정을 적용한다. 코드화 과정은 점열을 차례로 이은 일련의 선분을 유한한 크기의 코드북에 의한 코드열로 변환하는 과정이다[8]. 본 논문에서는 8방향 체인코드를 사용한다. 이렇게 생성된 체인코드는 학습데이터나, 인식 데이터로 사용할 수 있게 된다.

### 3.2 학습데이터에 주성분 분석을 적용하여 고유벡터(eigenvector)를 생성

타블릿을 통해 입력된 문자의 특징은 전처리 단계를 거쳐 체인코드의 형태로 구해지게 된다. 이때 각 문자에 대한 체인코드들은 하나의 집합을 이루게 되고 이 집합들을 묶어 학습을 위한 훈련데이터 집합을 구성하게 된다. 체인코드들은 입력 순서대로 세로 방향으로 나열되어 하나의 행렬 형태의 훈련데이터 집합으로 구성된다.

기저벡터인 고유벡터를 구하기 위하여 각 문자의 체인코드의 길이가  $N$ 이고, 훈련데이터 집합에 있는 체인코드들의 개수가  $M$ 개일 때, 각각의 문자를  $N$ 개의 성분을 가지는 1차원 열벡터들로 표현하면  $N \times M$  행렬로 구성할 수 있다.

다음 식 (1)에서  $\Gamma_n$ 은 훈련데이터 집합안에 있는 각각의 문자를 나타낸다.

$$\Gamma_n, n=1, \dots, M \tag{1}$$

그 다음 훈련데이터 집합 평균 벡터( $\Psi$ )을 다음의 식 (2)로 구한다.

$$\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n \tag{2}$$

이때 벡터  $\phi_n (= \Gamma_n - \Psi)$ 라고 정의 하고 이를 이용해서 공분산 행렬을 다음 식 (3)과 같이 만든다.

$$C = \frac{1}{M} \sum_{n=1}^M \phi_n \phi_n^T \tag{3}$$

이 공분산 행렬에서 데이터의 분포를 가장 잘 표현하는  $N$ 개의 정규 직교 고유벡터  $e_1, e_2, \dots, e_N$ 와 이와 관련된 고유값  $\lambda_1 > \lambda_2 > \dots > \lambda_N$ 를 구한다. 관련된 고유값들은 데이터간의 분산을 특징 짓는데 있어서의 그 값의 유용성에 따라 크기 순으로 정렬한다. 고유벡터들을 고유값의 크기 순으로 재배열한다. 이렇게 얻어진 고유벡터들은 앞쪽에 있을수록 큰 고유값과 관련이 있게 되므로 앞쪽에 있을수록 중요하다.

### 3.3 각 문자의 성분(weighting)값을 군집화하여 평균 성분값 생성

학습기에서는 생성된 고유벡터를 이용하여 훈련 데이터 집합에 있는 문자들을 재구성 할 수 있다. 즉 훈련데이터 집합에 있는 문자들을 앞서 구한 일련의 고유벡터로 표현하기 위해, 고유벡터 공간으로 투사시켜 그 성분값을 구한다. 이때 우선 처리할 데이터의 양을 줄이기 위해 큰 고유치들에 대응하는  $M'$  ( $M' < N$ )개의 고유벡터를 사용하게 된다. 훈련데이터 집합내에 있는 문자들의 고유벡터 공간상에서의 성분값은 다음 식 (4)을 통하여 구할 수 있다.

$$\omega_k = e_k^T (\Gamma - \Psi) \text{ for } k = 1, K, M' \tag{4}$$

이를 통해 훈련데이터 집합에 있는 모든 문자에 대한 성분값을 얻을 수 있다. 만약 훈련데이터가 10개의 숫자를 각각 10번씩 훈련한 경우라면 전체 훈련데이터의 개수는 100개가 된다. 구한 각 문자에 대한 성분값들은 그림 2에서와 같이 유사한 것끼리 유사한 위치에 있게 된다. 즉 같은 문자들은 유사한 위치에 있게 된다. 그러므로 같은 문자에 대해서는 평균 성분값을 구해 그 문자를 대표하는 표준 성분값으로 사용하게 된다.

### 3.4 문자 인식

인식기에서는 새로운 테스트 문자가 입력되면, 이

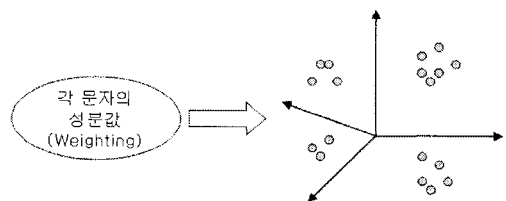


그림 2. 각 문자들의 성분값의 분포

문자는 전처리 단계를 거쳐 특징을 나타내는 체인코드를 생성한 후 고유벡터 공간으로 투사시켜 그 성분값을 구한다. 이렇게 얻어진 성분값을 이용하여 입력된 문자를 표현하는데 있어서 각각의 고유벡터의 공헌도를 의미하는 벡터  $\Omega$ 를 다음 식 (5)와 같이 구한다.

$$\Omega^T = [\omega_1, \omega_2, \dots, \omega_M] \quad (5)$$

그림 2에서와 같이 문자의 성분값들은 벡터 공간 상에서 유사한 문자는 유사한 위치에 있다. 그러므로 식 5와 같이 값이 구해지면 학습기를 통해서 구한 각 문자의 고유벡터 공간에서의 평균 성분벡터( $\Omega_k$ )와 비교해서 유클리디안 거리(Euclidian distance) ( $\epsilon_k = \sqrt{\sum |\Omega - \Omega_k|^2}$ )가 최소가 되는 문자가 입력 문자와 유사한 문자로 인식이 된다.

## 4. 구현 및 결과

### 4.1 구현 환경

본 논문에서 구현된 시스템은 Intel Pentium IV 2.8GHz, 1GB 메모리를 가지고 운영체제는 Microsoft Windows XP Professional 환경에서 구현되었으며, 문자데이터로부터 전처리 단계를 거쳐 체인코드를 생성하기 위해서 Wacom tablet과 Visual Studio .Net 2003의 Visual C#과 Microsoft Tablet PC SDK API version 1.5[9]로 구현하였다.

각 문자 훈련을 위한 사용자 인터페이스는 그림 3과 같이 2\*5크기의 박스를 만들어 '0'부터 '9'까지의 숫자를 타블릿에 전자펜을 이용하여 필기하면 해당 문자는 각 문자 파일에 체인코드 열들이 저장된다. 또한 인식을 위한 데이터 역시 학습 데이터를 만드는 과정과 마찬가지로, 타블릿을 이용하여, 각 문자 파일로 저장되며 이러한 데이터들은 학습데이터로 재사용이 가능하다.

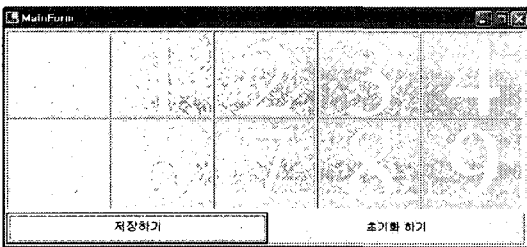


그림 3. 문자인식 인터페이스

### 4.2 구현 결과 및 분석

제안하는 방식의 전체적인 성능을 조사하기 위해서 먼저 PCA를 적용하여 eigenvector를 생성하였을 때, 몇 개의 주성분 차원을 선택하는 것이 필요한지를 조사하였다. 주성분 차원수를 구하는 실험을 간단하게 하기 위해서 '0'부터 '9'까지의 각각의 문자를 10명이 5번씩 필기하도록 하여, 각 문자당 평균적으로 50개 정도의 해당 문자를 가지고 있는 훈련데이터를 이용하였으며 주성분 차원을 위한 인식데이터는 '0'부터 '9'까지의 문자를 각 문자당 20개씩 인식하도록 하였다. 그런 다음 주성분 차원수를 변화시켜가면서 '0'부터 '9'까지의 문자의 인식률을 살펴보았다. 그림 4에서와 같이 차원의 수가 커질수록 문자의 인식률이 높아지다가 어느 정도 되면 수렴됨을 볼 수 있으며 차원수가 5만 되어도 데이터를 충분히 표현할 수 있음을 알 수 있었다. 따라서 본 논문에서는 실험을 통해 알아낸 수치인 5를 주성분 차원수로 고정하여 실험을 하였다.

문자 훈련은 숫자 '0'부터 '9'까지 30명의 데이터를 사용하였고, 각 사람은 각각의 문자를 10번 필기하도록 해, 각 모델은 평균적으로 300개 정도의 해당 문자를 가지고 훈련되었다. 또한 체인 코드의 길이는 20으로 하였다.

인식 데이터는 훈련에 참가하지 않은 10명이 '0'부터 '9'까지의 문자를 4개씩 40개를 인식시키도록 하였다. 사용될 고유벡터의 수는 그림 4에서 구한 차원수인 5를 이용하였다. 차원수를 5로 하였을 때의 각 문자의 표준 성분값은 그림 5와 같다.

훈련시 얻은 데이터를 이용하여 각 문자의 표준 패턴을 구할 수 있고, 이를 기준으로 하여 인식 데이터와 각 문자의 표준 패턴의 유클리드 거리를 이용하여 거리가 가장 짧은 문자가 인식이 되게 시험을 하

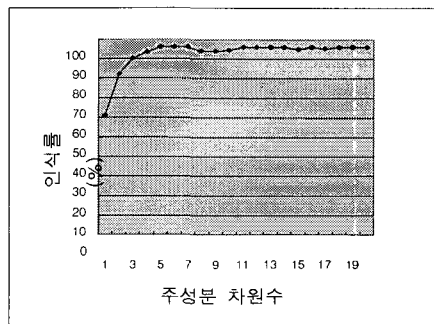


그림 4. 주성분 차원수에 따른 문자의 인식률

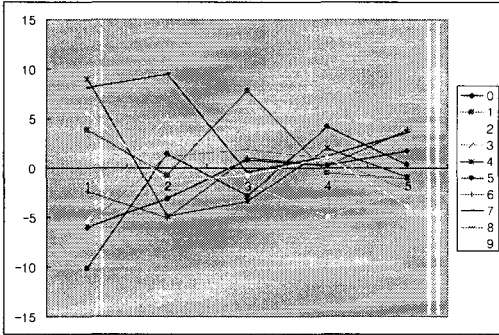


그림 5. 차원수가 5인 각 문자의 표준 성분값

표 1. 차원수 5일 때의 문자 인식 결과

문자	패턴수	인식	오인식	문자	패턴수	인식	오인식
0	40	36	4	1	40	40	0
2	40	36	4	3	40	36	4
4	40	40	0	5	40	39	1
6	40	40	0	7	40	40	0
8	40	38	2	9	40	39	1

였다. 인식 결과는 표 1과 같다. 이때, 0은 6과 8로, 2는 7로, 3은 9로 오인식이 일어났으며, 전체 인식률은 96%를 보였다.

오인식의 원인을 알기 위하여 오인식이 발생한 문자인 '0'과 '6' 그리고 '8', '3'과 '9'의 표준 성분값을 그림 6에서 비교해 보았다. 그림에서 알 수 있듯이 2개 문자의 평균 성분값이 매우 가까움을 알 수 있었다. 따라서 이들 사이에서는 오인식이 날 수 있는 가능성이 다른 문자들 보다 높음을 알 수 있다. 추후에는 이들 사이의 관계를 더욱 명확히 할 수 있는 추가적인 구분과정과 함께 문자의 특성을 좀더 잘 표현해 줄 수 있는 전처리 방법이 필요하다.

제안기법과 HMM 기법과의 비교 분석을 하였다. 이때 사용된 학습데이터와 인식데이터는 표 1의 데이터를 사용하였다. 그 결과는 표 2와 같다.

표 2에서 보이는 것처럼 제안 기법을 사용한 경우 문자당 인식률에서 적게나마 인식률이 떨어지는 반면 문자당 평균 인식 시간이 기존의 모델들을 사용한 경우보다 더욱 유리함을 알 수 있다. 제안기법은 전체 패턴을 비교하여 계산하는 HMM과는 다르게 주 성분 차원수를 줄여서 나온 성분값만 가지고 인식과정을 거치기 때문에 계산량을 적게 차지하게 되므로 인식시간이 덜 걸린다는 것을 알 수 있다.

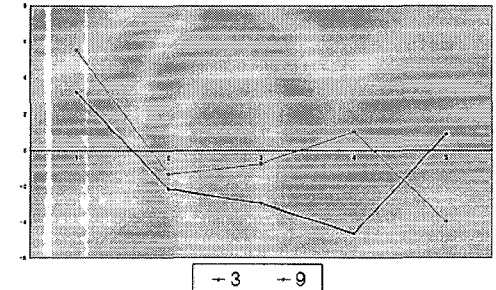
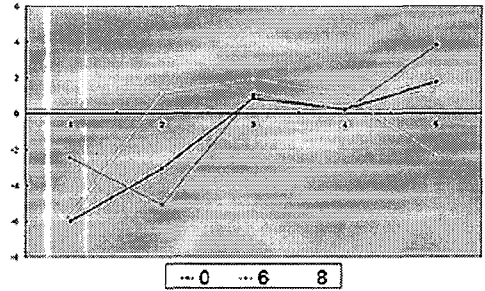


그림 6. 오인식 문자의 표준 주성분

표 2. 기존의 방법들과의 비교 분석

기법	총 문자수	오인식	인식률	문자당 평균 인식시간
PCA를 이용한 기법	400	16	96%	0.01초
HMM	400	10	97.5%	0.11초
폭선 정합 방법	400	18	95.5%	0.27초
신경망	400	10	97.5%	0.30초

## 5. 결 론

온라인 문자 인식시에는 동일한 문자일지라도 필기자 고유의 특성에 따라 획수 및 획순의 변형이 다양하게 나타나게 되므로 이런 변형을 흡수할 수 있는 알고리즘을 개발하는 것은 쉬운 일이 아니다. 또한 필기자의 개인차에 의해 나타나는 모든 가능한 패턴의 변형을 흡수할 수 있는 특징의 파악 및 구조 해석이 어렵기 때문에 각 문자 패턴에 대하여 필기자에 따른 다양한 필기 패턴들을 모델로 생성하거나, 입력 패턴에 대해 모든 표준 패턴을 정합하는 방법론이 사용되고 있으나, 이러한 방법들은 많은 기억장소와 계산량을 필요로 하는 단점을 지니고 있다.

본 논문에서는 기존의 온라인 문자 인식 기법들이 가지고 있는 많은 계산량과 메모리 사용을 줄이기 위한 방법을 제시하고자 하였다. 그를 위해 전처리

단계를 거친 각 문자의 특징을 나타내는 8방향 체인 코드로 이루어진 학습데이터들과 인식데이터에 PCA를 적용하여, 차원수를 줄임으로써 문자인식에 쓰이는 데이터의 양을 줄여 계산량을 줄이고자 하였고 따라서 인식 시간을 줄이고자 하였다. 또한, 각 문자에 대한 평균 성분값을 구함으로써 문자 인식시 모든 학습데이터를 검사하는 것이 아니라 각 문자를 대표하는 평균 성분값들만을 검사하게 함으로써 인식 효율을 높이고자 하였다.

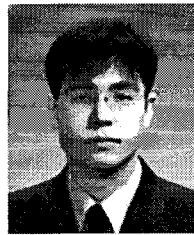
추후 연구 과제로는 숫자에 한정된 데이터를 영문 및 한글로도 확대하고, 이번 논문에서는 글자의 표기 시 한 방향으로만 작성된 문자 데이터만 사용하였으나 같은 글자이되 획의 순서가 다른 경우에 대해서도 연구를 할 예정이다. 또한 문자 인식시 '0'이 '6' 이나 '8'로 또는 '3'이 '9'로와 같이 혼동되는 몇 가지 부류에 대해서 추가적인 구분과정의 도입이 필요하며, 전처리 단계에서 각 문자의 특징을 최대한 수용할 수 있는 기술에 대한 연구가 필요하다.

참 고 문 헌

[1] 이성환, "온라인 필기 인식 기술의 현황," 한국 정보과학회지, 제 9권, 제 1호, pp. 54-63, 1991.  
 [2] 이성환, 문자인식-이론과 실제, 홍릉과학출판사, 서울, 1993.  
 [3] L. R. Labiner, "A Tutorial on HMM and Selected Applications in Speech Recognition," Proc. IEEE, Vol. 77, No. 2, pp. 257-289, 1989.  
 [4] M. KIRBY and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces," IEEE transactions on pattern analysis and machine intelligence, Vol. 12, No. 1, pp. 103-108, 1990.  
 [5] C.C.Tappert, "Cursive Script Recognition by Elastic Matching," IBM Journal of Research and Development, Vol. 26, No. 6, pp. 765-771, 1982.  
 [6] 이희동, 김태균, T.Agui, M.Nakajima, "확장된 DP매칭법에 의한 흘림체 한글 온라인 인식," 대한 전자공학회 논문지, 제 15권, 제 3호, pp. 29-37, 1989.  
 [7] Deepu. V, Madhvanath. S and Ramakrishnan.

A.G., "Principal Component Analysis for Online Handwritten Character Recognition," Proc. IEEE, Vol. 2, No. 2, pp. 327-330, 2004.

[8] 이재준, 은닉 마르코프 모델을 이용한 한/영 혼용 필기의 온라인 인식, 한국 과학 기술원 석사 학위 논문, 1994.  
 [9] Rob Jarrett and Philip Su, Building Tablet PC Applications, Microsoft Press, Washington, 2003.



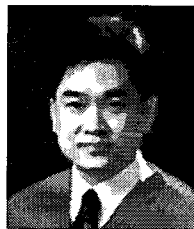
유 재 만

2003년 광운대학교 컴퓨터과학과 학사 졸업  
 2005년 광운대학교 컴퓨터과학과 공학 석사  
 2005~현재 (주) 다리테크 개발 연구소 연구원  
 관심분야: 멀티미디어 시스템, 영상/비디오 처리 및 패턴인식



한 정 훈

2004년 광운대학교 화학과 학사 졸업  
 2006년 광운대학교 컴퓨터과학과 공학 석사  
 관심분야: 멀티미디어, 데이터베이스, 영상 처리 및 패턴 인식



김 우 생

1985 서울대 수료 및 University Texas at Austin 전산학과 졸업  
 1987 University of Minnesota 전산학 이학 석사  
 1991 University of Minnesota 전산학 이학 박사  
 2001 University of California at Berkeley 교환 교수  
 1992~현재 광운대학교 컴퓨터 공학부 교수  
 관심분야: 멀티미디어 시스템, 데이터베이스, 영상/비디오 처리 및 패턴인식