

論文2000-37SP-11-5

# Bhattacharyya distance 기반 특징 추출 기법 (Feature Extraction Method Using the Bhattacharyya Distance)

崔義善\*, 李哲熙\*  
(Euisun Choi and Chulhee Lee)

## 요 약

Bhattacharyya distance는 패턴 분류 문제에 있어서 클래스간 분리도 측정의 수단으로 사용되어 왔으며 특징 추출 시 유용한 정보를 제공한다. 본 논문에서는 최근 발표된 Bhattacharyya distance를 이용한 에러 예측 기법을 이용하여 예측된 분류 에러가 최소가 되는 특징 벡터를 추출하는 방법에 대하여 제안한다. 제안한 특징 추출 기법은 최적화 알고리즘인 전체탐색 및 순차탐색 방법의 적용 시 분류 에러를 직접 구하지 않고 Bhattacharyya distance를 이용하여 분류 에러를 예측하므로 고차원 데이터의 경우 고속의 특징 추출이 가능하며, 에러 예측 성질을 이용하여 패턴 분류 시 필요한 최소 특징 벡터의 수를 예측할 수 있는 장점이 있다.

## Abstract

In pattern classification, the Bhattacharyya distance has been used as a class separability measure. Furthermore, it is recently reported that the Bhattacharyya distance can be used to estimate error of Gaussian ML classifier within 1-2% margin. In this paper, we propose a feature extraction method utilizing the Bhattacharyya distance. In the proposed method, we first predict the classification error with the error estimation equation based on the Bhattacharyya distance. Then we find the feature vector that minimizes the classification error using two search algorithms: sequential search and global search. Experimental results show that the proposed method compares favorably with conventional feature extraction methods. In addition, it is possible to determine how many feature vectors are needed for achieving the same classification accuracy as in the original space.

## I. 서 론

일반적으로 패턴 인식은 수집된 정보에 대하여 분류 과정에 필요한 정보를 추출하는 특징 추출(feature extraction) 단계와 추출된 특징을 이용하여 분류기를 통하여 인식을 수행하는 패턴 분류(pattern classification) 단계로 구분된다. 특징 추출은 고차원 데이터를 저차원 데이터로 변환하여 패턴 분류시 복잡도(complexity)를 감소시키며 분류기의 성능을 향상시

켜 효율적인 패턴 분류를 가능하게 한다. 따라서 특징 추출은 패턴 인식 문제에 있어서 중요한 연구 주제이며 다수의 연구자들에 의하여 폭넓게 연구되어 왔다<sup>[1-5]</sup>.

대부분의 기존 특징 추출 알고리즘들은 두 개의 클래스에 대하여 정의된 결정 기준 함수를 사용한다. 예를 들어, Foley-Sammon 방법은 다음 식과 같은 결정 기준 함수의 값을 최대로 하는 특징 벡터  $d$ 를 구한다<sup>[4]</sup>.

$$f(d) = \frac{d^T B d}{d^T \Sigma d}$$

여기서  $B = (M_1 - M_2)(M_1 - M_2)^T$ ,  $\Sigma = k\Sigma_1 + (1-k)\Sigma_2$  ( $0 \leq k \leq 1$ )이고,  $M_i, \Sigma_i$ 는 각각 클래스  $\omega_i$ 의 평균 벡터와 공분산 행렬이다. Foley-Sammon 방법은 클레

\* 正會員, 延世大學校 電氣·電子工學科

(Dept. Electrical and Electronic Eng., Yonsei University)

※ 본 연구는 과학재단 특장기초연구비 지원으로 이루어졌습니다.

接受日字:2000年2月25日, 수정완료일:2000年10月16日

스의 평균 벡터에 중점을 둔 기법으로 클래스들간의 평균벡터의 차이가 비교적 클 경우 효과적이다. 한편, 두 클래스의 결정 경계(decision boundary)에 기초한 결정 경계 특징 추출 방법(decision boundary feature extraction)은 다음 식과 같은 결정 기준함수를 사용한 다<sup>[5]</sup>.

$$\Sigma_{DBFM} = \frac{1}{K} \int_S N(X)N(X)^T \mu(X) dX$$

여기서  $N(X)$ 는 결정 경계에 수직한 벡터이며,  $\mu(X)$ 는 확률 밀도 함수,  $K = \int_S \mu(X) d(X)$ 이며  $S$ 는 결정 경계이다. 결정 경계 특징 추출 방법은 결정 경계에 수직한 방향의 특징 벡터를 구하는 방법으로 실제 알고리즘 적용 시 유효 결정 경계(effective decision boundary)  $S_E$ 를 사용하게 되며, 평균 벡터와 공분산의 차이를 모두 효율적으로 이용한다. 그러나 유효 결정 경계를 학습데이터(training data)에서 수치적으로 구하게 됨으로 처리시간과 정확도에 문제가 발생할 수 있다<sup>[5]</sup>.

본 논문에서는 두 클래스 패턴 분류 문제에 대하여 최근 발표된 Bhattacharyya distance를 이용한 에러 예측 기법<sup>[6,7]</sup>을 사용하여 분류 에러를 예측하고, 예측된 분류 에러를 최소화하는 특징 추출 기법을 제안한다. 분류 에러를 최소화하는 특징 벡터를 구하기 위하여 본 논문에서는 최적화 기법인 순차탐색 및 전체탐색 기법을 적용한다<sup>[8]</sup>. 제안한 특징 추출 방법은 에러 예측 기법을 이용하여 분류 에러를 예측하므로 고속의 특징 추출이 가능하며, 에러 예측 성질을 이용하여 패턴 분류시 필요한 최소 특징 벡터의 수를 예측할 수 있는 장점이 있다.

앞으로 본 논문의 구성은 제II장에서 선형 특징 추출의 개요와 Bhattacharyya distance를 이용한 에러 예측 기법을 소개하고 제III장에서 Bhattacharyya distance를 이용한 에러 예측 기법을 특징 추출에 적용하는 방법에 대하여 구체적으로 기술한다. 제IV장에서는 실험 결과 및 기존 알고리즘들과의 패턴 분류 시 특징 추출 성능을 비교 평가하고 마지막으로 제V장에서 결론을 맺는다.

## II. 연구 배경

### 1. 선형 특징 추출

일반적으로 선형 특징 추출은 식 (1)과 같이  $N$ 차원 유클리드 벡터 공간  $E^N$ 에서 고차원 데이터  $X$ 를 저차원 데이터  $Y$ 로 선형 변환하는 것으로 이해될 수 있으며, 다수의 특징 추출 알고리즘들이 선형 특징 추출로 분류된다<sup>[1]</sup>.

$$\begin{aligned} Y &= A^T X = [\phi_1, \phi_2, \dots, \phi_M]^T X \\ M_Y &= A^T M_X A \\ \Sigma_Y &= A^T \Sigma_X A \end{aligned} \quad (1)$$

식 (1)에서  $A$ 는  $N \times M (M < N)$  행렬이며  $\phi_i$ 는  $N \times 1$  특징 벡터이다. 또한  $M_Y$ 와  $\Sigma_Y$ 는 각각 행렬  $A$ 에 의하여 선형 변환된 평균 벡터와 공분산 행렬이다. 패턴 분류 시 특징 추출은 분류에 필요한 식별 정보(discriminatory information)만을 추출하여 효율적인 패턴 분류를 가능하게 한다.

### 2. Bhattacharyya distance를 이용한 에러 예측

Bhattacharyya distance는 두 클래스의 분리도(separability) 측정의 수단이 되며 Bayes 에러의 상위 한계와 하위 한계를 제공한다<sup>[3]</sup>.

정규 분포를 가정할 경우 클래스  $\omega_1, \omega_2$ 에 대하여 Bhattacharyya distance는 다음 식 (2)와 같이 정의된다.

$$\begin{aligned} \mu\left(\frac{1}{2}\right) &= \frac{1}{8} (M_2 - M_1)^T \left[ \frac{\Sigma_1 + \Sigma_2}{2} \right]^{-1} \\ & \quad (M_2 - M_1) + \frac{1}{2} \ln \frac{|\Sigma_1 + \Sigma_2|/2}{|\Sigma_1|^{1/2} |\Sigma_2|^{1/2}} \end{aligned} \quad (2)$$

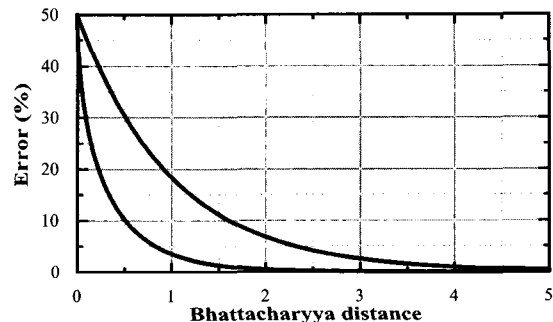


그림 1. Bhattacharyya distance를 이용한 분류 에러의 이론적 상위한계와 하위한계

$$\frac{1}{2} (1 - \sqrt{1 - e^{-2\mu(1/2)}}) \leq \epsilon \leq \frac{1}{2} e^{-\mu(1/2)}$$

Fig. 1. Theoretical upper and lower bounds of classification error.

$$\frac{1}{2} (1 - \sqrt{1 - e^{-2\mu(1/2)}}) \leq \epsilon \leq \frac{1}{2} e^{-\mu(1/2)}$$

두 클래스의 분포가 정규분포이고 동일한 선행 확률(prior probability)을 가질 경우 분류에러의 상위 한계와 하위 한계는 다음 식과 같으며 그림 1은 이를 그래프로 보여준다.

$$\frac{1}{2}(1 - \sqrt{1 - e^{-2\mu(1/2)}}) \leq \epsilon \leq \frac{1}{2} e^{-\mu(1/2)}$$

그러나 그림 1에서 볼 수 있듯이 상위 한계와 하위 한계의 간격이 비교적 크므로 특징 추출을 포함한 실제 문제에 있어서 유용한 정보를 제공하지 못한다.

최근 Bhattacharyya distance와 실제 Gaussian ML 분류기의 에러와의 관계를 실험적으로 조사하여 1-2%의 오차 한계에서 에러를 예측할 수 있는 에러 예측식이 제안되었다<sup>[7]</sup>. 제안된 에러 예측 기법에서는 먼저 Bhattacharyya distance와 분류 에러와의 관계를 유도하기 위하여 그림 2와 같이 정규 분포를 가지는 두 클래스  $N(M_1, \Sigma_1)$ ,  $N(M_2, \Sigma_2)$ 를 각각  $N(0, I)$ ,  $N(M, \Sigma)$ 으로 대각화(diagonalization)한다<sup>[3]</sup>. 그림 2는 이러한 대각화 과정을 보여준다. 여기서  $I$ 는 단위 행렬이고  $M = [m_1, m_2, \dots, m_N]^T$ 이며  $\Sigma$ 는  $\lambda_1, \lambda_2, \dots, \lambda_N$ 을 대각원소로 가지는 대각행렬이다. 대각화된 좌표계에서 Bhattacharyya distance는 다음의 식 (3)과 같이 나타낼 수 있다.

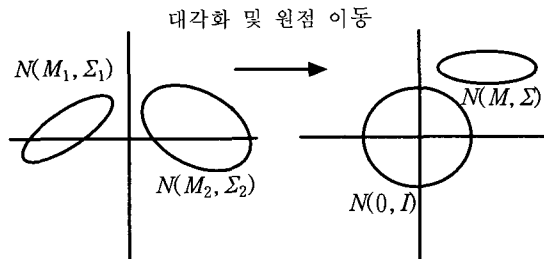


그림 2. 정규분포 클래스의 대각화(2차원 예)  
Fig. 2. Simultaneous diagonalization of two classes(2 dimension).

$$\mu\left(\frac{1}{2}\right) = \frac{1}{2} \sum_{i=1}^N \left\{ \frac{m_i^2}{2(1+\lambda_i)} + \ln\left(\frac{1+\lambda_i}{2}\right) - \frac{1}{2} \ln \lambda_i \right\} \quad (3)$$

Lee는 [7]에서  $m_1, m_2, \dots, m_N$ 과  $\lambda_1, \lambda_2, \dots, \lambda_N$ 이 이루는 공간(space)을 샘플링하여 약 1억 6천만 개의 클래스 조합을 생성하고 생성된 각 클래스 조합에 대하여 Bhattacharyya distance 및 실제 패턴 분류 시 분류 에러를 조사하여 다음의 식 (4)와 같은 에러 예측식을 제안하였다. 그림 3은 Bhattacharyya distance와 분류 에러와의 관계를 보여준다.

$$\begin{aligned} \epsilon = & 40.219 - 70.019 * \mu + 63.578 * \mu^2 - 32.766 * \mu^3 \\ & + 8.7172 * \mu^4 - 0.91875 * \mu^5 \quad (4) \end{aligned}$$

그림 3에서 볼 수 있듯이 분류 에러를 비교적 정확하게 예측할 수 있음을 볼 수 있다. 식 (4)의 에러 예측식은 클래스 데이터의 분포가 정규분포를 가질 경우 Gaussian ML 분류기의 에러를 1-2%의 오차 한계에서 예측할 수 있는 것으로 보고되었다<sup>[7]</sup>.

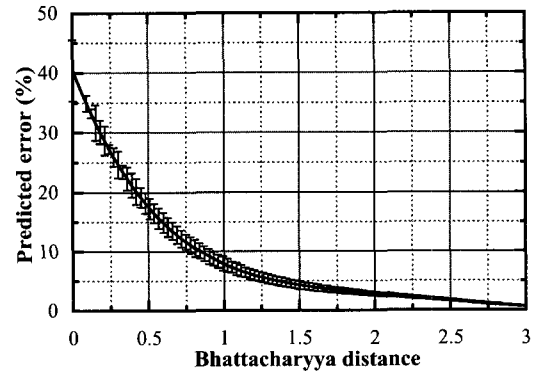


그림 3. Bhattacharyya distance와 에러와의 관계  
Fig. 3. Relationship between Bhattacharyya distance and classification error.

### III. Bhattacharyya distance를 이용한 특징 추출

본 논문에서는 Bhattacharyya distance를 이용하여 예측된 에러가 최소가 되는 특징 벡터를 추출하기 위하여 특징 추출을 위한 공간 탐색 알고리즘인 순차 탐색 및 전체탐색 방법을 적용하여 전체 특징 벡터 공간을 탐색한다<sup>[8]</sup>. 순차탐색 방법은 특징 벡터의 수를 증가시킬 경우 이전에 구해진 특징 벡터와 함께 분류 에러가 최소가 되는 특징 벡터를 추가로 구하며, 전체 탐색 방법은 특징 벡터의 수가 증가하는 경우 새로운 초기 특징 벡터 집합을 설정하여 이전에 구해진 특징 벡터에 제한을 받지 않는다.

#### 1. 순차 탐색

$N$ 차원 유클리드 벡터공간에서 직교 기저벡터 집합  $\Phi_N = \{\phi_1, \phi_2, \dots, \phi_N\}$ 을 고려한다. 여기서  $\phi_i$ 는  $N \times 1$  벡터이다. 순차탐색 방법은 먼저 특징 벡터 한 개를 추출하는 경우  $\phi_1$ 을 초기 특징 벡터로 가정하여 특징

을 추출하고 분류 에러를 예측한다. 에러 예측의 경우 식 (1)에서와 같이 벡터  $\phi_1$ 에 의하여 선형 변환된 평균 벡터와 공분산 행렬을 이용하여 식 (2)로부터 Bhattacharyya distance를 계산하고 식 (4)의 에러 예측식을 이용하여 에러를 예측한다. 즉

$$y_1 = \phi_1^T X$$

$$m_{1y} = M_Y = \phi_1^T M_X$$

$$\sigma_{1y} = \Sigma_Y = \phi_1^T X \phi_1$$

이며, 여기서  $m_{1y}$ 과  $\sigma_{1y}$ 는 식 (2)에서 Bhattacharyya distance를 구하기 위하여 사용된다.

그 다음 특징 벡터  $\phi_1$ 을 식 (5)와 같이 나머지 기저 벡터 방향으로 이동시킨다.

$$\phi_1^i = \phi_1 + \delta\phi_i \quad (i=2, \dots, M) \quad (5)$$

여기서  $\delta$ 는 양의 상수이며 그림 4는 위의 과정을 보여준다. 이동된 벡터  $\phi_1^i$ 을 특징 벡터로 사용하여 다음의 식 (3)과 같이  $\phi_i$  방향으로의 에러의 변화율  $r_i$ 를 계산한다.

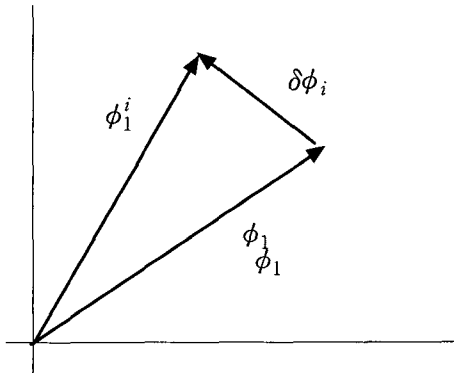


그림 4. 특징 벡터의 이동  
Fig. 4. Movement of the feature vector.

$$r_i = \frac{\Delta \epsilon}{\delta} = \frac{\epsilon(\phi_1^i) - \epsilon(\phi_1)}{\delta} \quad (6)$$

여기서  $\epsilon(\phi_1^i)$ 와  $\epsilon(\phi_1)$ 는 각각  $\phi_1^i$ 과  $\phi_1$ 을 특징 벡터로 사용하였을 때 에러 예측식으로 계산되는 값이다. 변화율  $r_i$ 를 나머지 벡터  $\phi_i$  ( $i=2, \dots, M$ )에 대하여 계산하고 다음의 식 (7)과 같이 특징 벡터  $\phi_1$ 을 갱신한다.

$$\phi_{1, updated} = \phi_1 + \alpha \sum_{i=2}^M (-r_i \phi_i) \quad (7)$$

여기서  $\alpha$ 는 양의 상수이며 특징 벡터  $\phi_{1, updated}$ 는 분류 에러를 감소시키는 방향으로 갱신된다. 이러한 과정을 식 (8)의 종료 조건을 만족할 때까지 반복하여 최종적으로 특징 벡터를 추출한다.

$$|\epsilon(\phi_{i, updated}^{(k+1)}) - \epsilon(\phi_{i, updated}^{(k)})| < \gamma \quad (8)$$

식 (8)에서  $k$ 는 반복 단계이며  $\gamma$ 는 문턱치(threshold)이다. 이와 같이 반복하는 경우 갱신된 특징 벡터  $\phi_{1, updated}$ 은 나머지 기저 벡터와 선형적으로 독립이 아니므로 매 반복 단계마다 Gram-Schmidt 방법을 적용하여 전체 기저 집합의 직교성을 유지한다<sup>[10]</sup>.

식 (5)에서 특징 벡터  $\phi_1$ 을 이동할 때 전체 벡터공간의 차원이  $N$ 차원이므로  $N-1$ 의 자유도(degree of freedom)가 존재하며, 추가로 특징 벡터를 구할 경우 이미 구해진 특징 벡터  $\phi_{1, updated}$ 과 함께 최소의 분류 에러를 얻을 수 있는 특징 벡터  $\phi_{i, updated}$ 를 위와 동일한 과정을 통해 구한다. 이 경우 자유도는 하나씩 감소하게 된다. 따라서 순차탐색은 다음에 설명할 전체탐색에 비하여 계산 속도가 빠른 장점이 있다.

## 2. 전체 탐색

앞서 기술한 순차탐색 방법은 두 개 이상의 특징 벡터를 구할 경우 이전에 구해진 특징 벡터를 그대로 사용하므로 이전에 구해진 특징 벡터가 최적 특징 벡터 중의 하나가 아닐 경우 결과적으로 최적의 특징 추출을 보장할 수 없다. 이러한 문제점을 해결하기 위하여 전체탐색 방법에서는 특징 벡터의 수를 증가시킬 경우 이미 구해진 특징 벡터에 제한 받지 않고 임의의 새로운 초기 특징 벡터집합  $\Phi_M = \{\phi_1, \phi_2, \dots, \phi_M\}$  ( $M < N$ )을 설정한다. 그리고 특징 벡터들 사이의 직교성을 유지하며 각각의 특징 벡터들을 나머지 벡터  $\phi_{M+1}, \phi_{M+2}, \dots, \phi_N$  방향으로 이동시키고 식 (4)의 에러 예측식을 이용하여 분류 에러의 변화율을 조사한다. 이 경우  $(N-M)$ 개의 새로운 특징 벡터 집합  $\Phi_{M(i)}$ 이 존재하게 된다.  $\Phi_{M(i)}$ 는 특징 벡터 집합  $\Phi_M$ 중에서 벡터  $\phi_j$ 를 나머지 벡터  $\phi_i$  방향으로 이동시켰을 때 생성되는 벡터 집합으로 다음과 같은  $(N-M)$ 개의 벡터 집합이 존재한다.

$$\{\phi_{M(j)}^{M+1}, \phi_{M(j)}^{M+2}, \dots, \phi_{M(j)}^N\} \quad (j=1, 2, \dots, M)$$

새롭게 구해진 특징 벡터 집합  $\phi_{M(j)}^i$ 을 이용하여 분류 에러를 예측하고 식 (9)와 같이 분류 에러의 변화율  $r_j^i$ 을 계산한다.

$$r_j^i = \frac{\Delta \epsilon}{\delta} = \frac{\epsilon(\phi_{M(j)}^i) - \epsilon(\phi_{M(j)})}{\delta} \quad (9)$$

이 과정을  $\phi_{M(j)}^i$  ( $j=1, 2, \dots, M$ )에 대하여 반복 수행하고 최종적으로 특징 벡터 집합  $\phi_M$ 을 식 (10)과 같이 갱신시킨다.

$$\phi_{M, updated} = \sum_{j=1}^M \left\{ \phi_j + \alpha \sum_{i=M+1}^N (-r_j^i \phi_i) \right\} \quad (10)$$

이러한 과정을 순차탐색 방법의 경우와 같이 분류 에러의 변화가 문턱 치보다 작을 때까지 반복하며 매 단계마다 Gram-Schmidt 방법을 적용하여  $N$ 차원 벡터공간에서의 직교 기저집합  $\phi_N$ 의 직교성을 유지한다. 예를 들어 3차원 벡터공간에서 두 개의 특징 벡터를 추출한다고 가정하면 다음 식과 같이 벡터집합  $\phi_{2(1)}^3$ ,  $\phi_{2(2)}^3$  각각에 대하여 식 (9)를 이용하여 분류 에러의 변화율을 계산하고 식 (10)을 이용하여 특징 벡터 집합  $\phi_{2, updated}$ 를 구한다.

$$\phi_{2(1)}^3 = \{\phi_1 + \alpha \phi_3, \phi_2\}$$

$$\phi_{2(2)}^3 = \{\phi_1, \phi_2 + \alpha \phi_3\}$$

$$\phi_{2, updated} = \{\phi_{1, updated}, \phi_{2, updated}\}$$

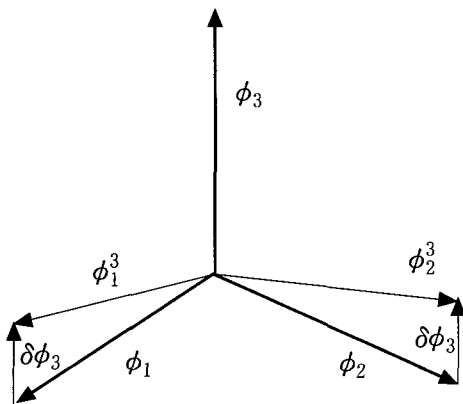


그림 5. 전체탐색 방법의 예 ( $N=3, M=2$ )  
Fig. 5. Illustration of the global search.

그림 5는 위의 과정을 보여준다. 순차탐색 알고리즘에 비하여 전체탐색 알고리즘의 장점은 초기 벡터집합의 설정이 자유롭다. 즉, 특징 수를 증가시킬 경우 이미 구해진 특징 벡터에 제한 받지 않고 초기 벡터 집합을 다르게 설정함으로써 지역극소에 도달할 가능성을 감소시킬 수 있다. 그러나 계산량은 순차탐색에 비하여 증가하는 단점이 있다.

### 3. 최소 특징 벡터의 수 예측

일반적으로 패턴 분류는 최소의 특징 벡터만을 사용하여 분류 에러를 최소화하는 것이 바람직하다. 따라서 특징 벡터를 추출하는 경우 필요한 특징 벡터의 개수에 대한 사전 지식은 효율적인 특징 추출을 가능하게 한다. 본 논문에서 제안한 특징 추출 방법은 Bhattacharyya distance를 이용하여 분류 에러를 예측하게 되므로 이러한 성질을 이용하여 주어진 문제에 대한 패턴 분류 시 요구되는 최소 특징 벡터의 수를 예측할 수 있는 장점이 있다. 즉 식 (1)의 특징 추출 과정을 수행하지 않은 원래 데이터의 평균 벡터  $M_X$ 와 공분산 행렬  $\Sigma_X$ 를 이용하여 식 (2)로부터 Bhattacharyya distance를 계산하고 식 (4)를 사용하여 분류 에러  $\epsilon(\phi_N)$ 을 예측한 후, 학습 단계 (training stage)에서 차원 증가에 따라 갱신되는 특징 벡터들을 이용하여 예측되는 분류 에러와의 관계가 다음과 같은 조건을 만족하는 경우 최소 특징 벡터의 수  $M_{min}$ 이 결정된다.

$$M_{min} = \min \left\{ M \mid \frac{1 - \epsilon(\phi_M)}{1 - \epsilon(\phi_N)} \geq t, \quad 0 < t \leq 1 \right\}$$

여기서  $t$ 는 1보다 작은 양의 상수이며, 본 논문에서는 0.99를 사용하였다.

## IV. 실험 결과

본 논문에서 제안된 특징 추출 기법의 성능을 평가하기 위하여 Gaussian ML 분류기를 이용하여 패턴 분류 시 분류 정확도(classification accuracy)를 측정하였다. 제안된 방법과 성능 비교를 위하여 기존의 특징 추출 알고리즘인 Foley-Sammon 방법 [4]와 결정 경계 특징 추출 방법 [5]를 사용하였다.

첫 번째 실험으로 본 논문에서 제안한 특징 추출 방법과 기존 특징 추출 알고리즘과의 패턴 분류 시 성

표 1. 정규 분포로 발생시킨 클래스 데이터의 정보

Table 1. Statistics of two dimensional Gaussian generated data.

	Case 1		Case 2	
	class 1	class 2	class 1	class 2
Mean vector	0.01 0	-0.01 0	-1 1	1 -1
Covariance matrix	4    0 0    4	4    0 0    1	4    3.6 3.6    4	4    3.6 3.6    4
No. sample (train)	1000 (300)		1000 (300)	

표 2. 2차원 정규 분포 데이터를 이용한 기존 알고리즘과의 성능 비교

Table 2. Performance comparison using two dimensional Gaussian generated data.

No. feature	전체탐색 (Bhattacharyya 에러 예측)		순차탐색 (Bhattacharyya 에러 예측)		Foley-Sammon method	결정 경계 특징 추출 방법
	평균	표준편차	평균	표준편차		
Case 1						
(%)	평균	표준편차	평균	표준편차		
1	64.195	0.799	64.044	0.834	57.3	64.4
2	65.4	-	65.4	-	65.4	65.4
Case 2						
(%)	평균	표준편차	평균	표준편차		
1	98.447	0.285	98.419	0.318	98.6	98.6
2	98.6	-	98.6	-	98.6	98.6

능을 비교하기 위하여 표 1과 같이 2차원 정규분포 데이터를 발생시켰다. 데이터의 샘플 수는 1000개이며 그 중 300개를 학습데이터로, 나머지를 시험데이터로 사용하였다. 표 1에서 case 1은 두 클래스의 평균벡터들의 차이가 적은 경우이며 case 2는 평균벡터들의 차이가 비교적 큰 반면 각 클래스의 공분산 행렬이 같은 경우로 상관 계수(correlation coefficient)는 0.9이다.

표 2는 case 1, 2에 대하여 제안한 방법과 기존 알고리즘인 Foley-Sammon 방법 및 결정 경계 특징 추출 방법과의 패턴 분류시 분류 정확도를 비교한 결과이다. 제안한 방법의 경우 임의의 초기 벡터를 이용하여 100회 반복 실험하였으며 표 2에 분류 정확도의 평균값과 표준 편차를 나타냈다.

표 2에서 case 1의 경우 특징 벡터 한 개를 추출할 때, 전체탐색 방법과 순차탐색 방법은 결정 경계 특징 추출 방법과 유사한 성능을 보이고 있는 반면, Foley-Sammon 방법은 57.3%의 분류 정확도로 성능이 떨어졌다. Foley-Sammon 특징 추출 방법은 클래스의 평균 벡터에 중심을 둔 기법으로 case 1과 같이 두 클래스의 평균 벡터의 차이가 적을 경우 일반적으

로 성능이 저하된다. 한편, case 2의 경우는 각 특징 추출 기법간의 성능 차이가 거의 없는 것으로 나타났다.

표 3. FSS 주요 사양

Table 3. Parameters of Field Spectrometer System.

No. bands	60 bands
Spectral Coverage	0.4 - 2.4 $\mu\text{m}$
Altitude	60 m
IFOV(ground)	25 m

다음 실험에서는 실제 원격 탐사된 고차원 데이터(FSS)의 통계치를 이용하여 정규 분포 데이터를 발생시키고 제안한 방법의 성능을 분석하였다. 표 3은 FSS 시스템의 주요 사양이며 표 4는 실험에 사용된 클래스 데이터의 정보를 보여준다<sup>[11]</sup>. 표 4에서 실험을 위해 데이터의 차원 수를 각각 8, 10, 12로 조정하고 각 클래스의 샘플 수는 1000개로 발생시켜 그 중 500개를 학습데이터로 사용하고 나머지는 시험 데이터로 사용하였다. 그림 6과 7은 차원 수가 8, 10인 클래스 데이터에 대하여 실험한 결과로 학습데이터와 시험 데이터 각각에 대하여 분류 정확도를 측정하고 성능을

비교하였다.

표 4. 원격 탐사된 클래스 데이터의 정보  
Table 4. Information of remotely sensed class data.

No. dimension	Class Species	Date	No. sample
8	SPRING WHEAT	77.09.20	129
	SPRING WHEAT	77.10.18	313
10	OATS	78.07.26	173
	OATS	78.09.21	182
12	WINTER WHEAT	77.09.20	292
	SPRING WHEAT	77.10.18	313

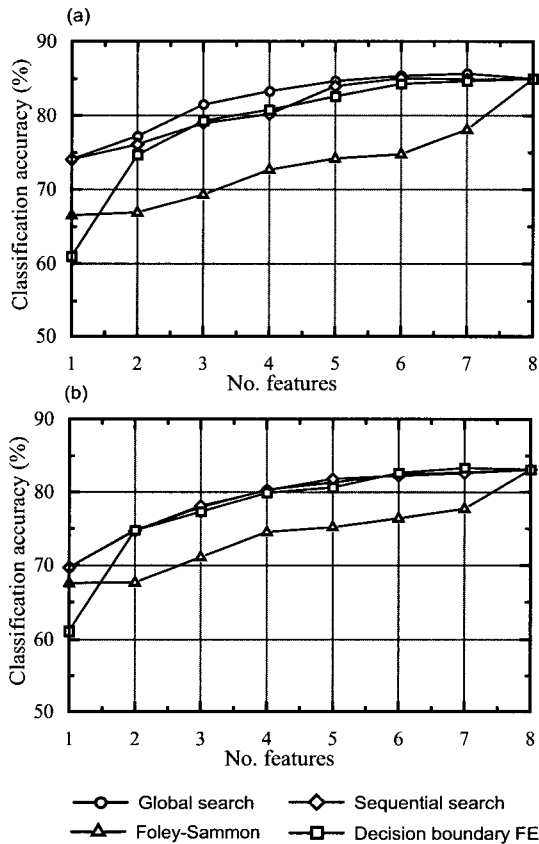


그림 6. 특징벡터 수의 증가에 따른 분류정확도 비교 (8 차원) (a) 훈련 데이터 (b) 시험 데이터 (정규 분포 데이터)

Fig. 6. Performance comparison.(8 dimension). (a) training data (b) test data (normal distribution)

그림 6, 7에서 볼 수 있듯이 본 논문에서 제안한 특징 추출 방법은 기존의 방법인 Foley-Sammon 방법과 결정 경계 특징 추출 방법과 비교하여 대체로 우수

한 성능을 보이고 있다. 특히, 그림 7에서 학습 데이터의 경우 제안한 특징 추출 방법은 결정 경계 특징 추출 방법과 유사한 성능을 보이고 있는 반면, Foley-Sammon 방법과 비교하여 최대 약 7% 이상의 성능 차이를 나타냈다. 또한 전체탐색 방법은 순차탐색 방법과 비교하여 데이터의 차원 수가 증가할수록 향상된 성능을 보였다.

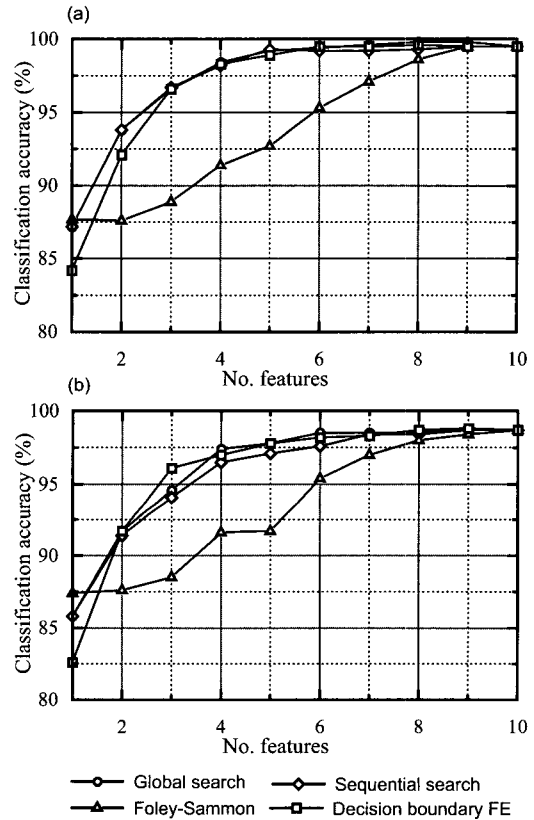


그림 7. 특징벡터 수의 증가에 따른 분류정확도 비교 (10차원) (a) 훈련 데이터 (b) 시험 데이터(정규 분포 데이터)

Fig. 7. Performance comparison.(10 dimension).(a) training data (b) test data(normal distribution)

한편, 제안한 방법은 Bhattacharyya distance를 이용한 에러 예측의 성질을 이용하여 특징 추출을 하지 않은 원래의 데이터를 그대로 패턴 분류에 사용하였을 경우 관측되는 분류 정확도와 동일한 수준의 분류 정확도를 얻을 수 있는 최소 특징 벡터의 수를 예측할 수 있다. 예를 들어 그림 8은 표 4의 12차원 데이터를 이용하여 실험한 결과로 전체탐색 방법을 이용하여 특징 추출을 하는 경우 특징 벡터의 수가 증가할

때, 학습 과정(training stage)에서 갱신되는 특징 벡터들을 이용하여 예측되는 분류 정확도를 도기한 그림이다. 그림 8에서 볼 수 있듯이 예측 정확도와 실제 정확도가 거의 일치함을 볼 수 있으며 예측 정확도를 사용하여 이 문제에 몇 개의 특징이 필요한지를 예측할 수 있다. 즉 모든 특징을 사용하여 얻을 수 있는 정확도의 99%를 원한다면 7개의 특징만을 사용하면 된다.

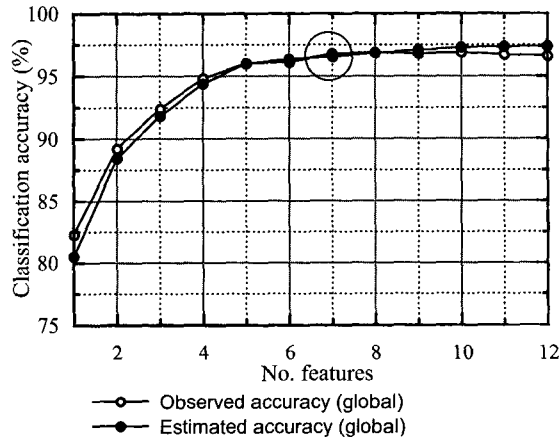


그림 8. 특징 추출 훈련 단계에서의 특징 벡터 수의 증가에 따른 예측 분류 정확도  
Fig. 8. Estimated classification accuracy of training data.

마지막 실험으로 표 5의 실제 원격 탐사 데이터에 대하여 실험하였다. 이 경우 실제 데이터를 그대로 실험에 사용하였으며 샘플 수를 고려하여 학습 데이터와 시험 데이터 모두에 전체 샘플을 적용하였다. 그림 9와 10은 각각 차원 수가 8, 10인 클래스 데이터에 대하여 실험한 결과이다. 그림 9에서 제안한 방법은 기존의 특징 추출 방법에 비하여 비교적 우수한 성능을 보이고 있다. 그러나 그림 10에서 볼 수 있듯이 4개 이하의 특징을 추출하는 경우 제안한 특징 추출 방법은 기존의 방법에 비하여 성능이 떨어졌으며, 순차탐색 방법은 전반적으로 성능이 저조하였다. 이러한 성능저하는 실제 데이터의 분포가 정규 분포에서 크게 벗어나기 때문으로 판단된다. 그림 11은 표 5의 10차원 데이터에서 2개의 특징을 선택하여 2차원 공간에 클래스의 분포를 도기한 예로서 실제 정규 분포와 다른 모습을 확인할 수 있었으며, 특히 많은 수의 아웃라이어(outliers)가 관찰되었다. 이러한 경우 본 논문에서 제안한 방법과 같은 데이터의 통계적 특성을 이

용하는 parametric 특징 추출 방법보다 신경망과 같은 non-parametric 특징 추출 방법이 효과적이다<sup>[9]</sup>.

표 5. 원격 탐사된 클래스 데이터의 정보  
Table 5. Information of remotely sensed class data.

No. dimension	Class Species	Date	No. sample
8	OATS	78.06.02	259
	WINTER WHEAT	78.09.21	292
10	SUMMER FALLOW	77.05.03	209
	SUMMER FALLOW	77.06.26	643

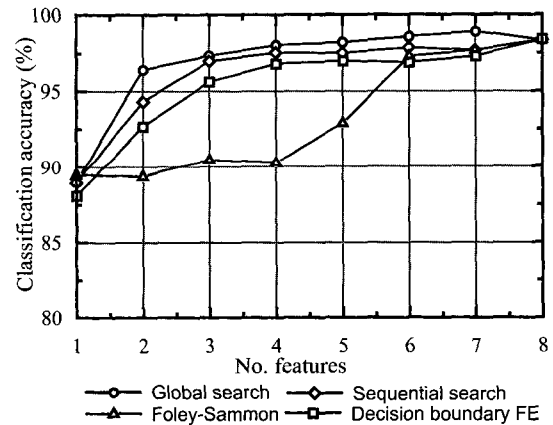


그림 9. 특징벡터 수의 증가에 따른 분류정확도 비교(8차원 실제 데이터)  
Fig. 9. Performance comparison(8 dimensional real data).

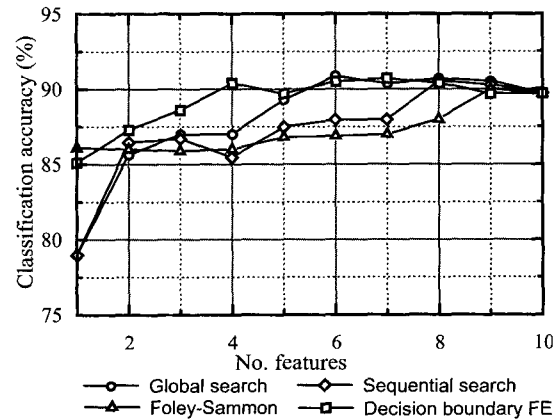


그림 10. 특징벡터 수의 증가에 따른 분류정확도 비교(10차원 실제 데이터)  
Fig. 10. Performance comparison(10 dimensional real data).



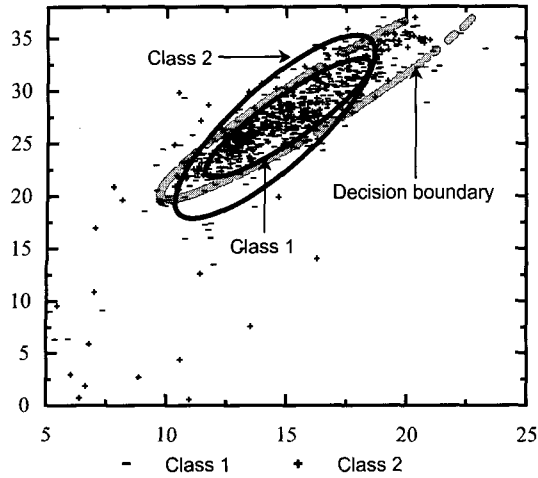


그림 11. 표 4의 10차원 실제데이터를 2차원 공간에 도시한 예

Fig. 11. Distribution of two classes of table IV (10 dimensional real data).

## V. 결 론

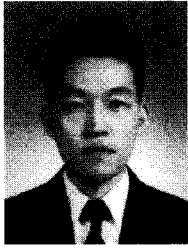
본 논문에서는 패턴 분류 및 특징 추출 시 유용한 정보를 제공하는 Bhattacharyya distance를 이용하여 에러를 예측하는 방법을 통하여 특징을 추출하는 방법을 제안하였다. 제안한 방법은 예측된 분류 에러가 최소가 되는 특징 벡터를 추출하는 방법으로 최적화 알고리즘인 전체탐색 및 순차탐색 방법의 적용 시 직접 분류 에러를 구하지 않고 Bhattacharyya distance를 이용하여 분류 에러를 예측하므로 고차원 데이터의 경우 고속의 특징 추출이 가능하다. 실험 결과 제안한 방법은 기존의 알고리즘에 비하여 비교적 우수한 성능을 나타냈으며, 에러 예측 성질을 이용하여 패턴 분류 시 필요한 최소 특징 벡터의 수를 예측할 수 있었다.

## 참 고 문 헌

[1] J. A. Richards, *Remote Sensing Digital Image Analysis*, Springer-Verlag, 1993.  
 [2] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, 1973.

[3] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, New York: Academic Press, 1990.  
 [4] D. Foley, J. Sammon, "An Optimal Set of Discriminant Vectors," *IEEE Trans. on Computers*, vol. c-24, no. 3, 1975.  
 [5] C. Lee and D. A. Landgrebe, "Feature extraction based on decision boundaries," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, pp. 388-400, 1993.  
 [6] C. Lee and D. A. Landgrebe, "Error estimation of the Gaussian ML classifier," *IEEE International Symp. on Information Theory*, pp. 535-535, 1997.  
 [7] C. Lee, E. Choi, "Bayes Error Evaluation of Gaussian ML Classifier," *IEEE Trans. on Geoscience and Remote Sensing*, 1999. (인쇄 중)  
 [8] C. Lee, E. Choi, J. Kim, "Optimal Feature Extraction for Normally Distributed Data," *SPIE AeroSense '98*, pp. 223-232, 1998.  
 [9] C. Lee and D. A. Landgrebe, "Decision boundary feature extraction for neural networks," *IEEE Trans. Neural Networks*, Vol. 8, No. 1, pp. 75-83, 1997.  
 [10] C. G. Cullen, *Matrices and Linear Transformation*. Addison-wesley Publishing Company, 1972.  
 [11] L. L. Biel and et. al., "A Crops and Soils Data Base For Scene Radiation Research," *Proc. Machine Process. of Remotely Sensed Data Symp., West Lafayette, Indiana*, 1982.

## 저 자 소개



崔 義 善(正會員)

1998년 2월: 연세대학교 전자공학과 졸업(공학사). 1998년 3월~2000년 2월: 연세대학교 전기·컴퓨터공학과 졸업(공학석사). 2000년 3월~현재: 연세대학교 전기·전자공학과 박사과정. 주관심분야:

패턴인식, 영상신호처리



李 哲 熙(正會員)

1980년 3월~1984년 2월: 서울대학교 전자공학과 졸업(공학사). 1984년 3월~1986년 2월: 서울대학교 대학원 전자공학과 졸업(공학석사). 1986년 9월~1987년 3월

: Technical University of Denmark (Researcher). 1987년 8월~1992년 12월: Purdue University Electrical Engineering(Ph. D.). 1993년 7월~1996년 8월: National Institutes of Health, Maryland, USA.(Visiting Fellow). 1996년 9월~1999년 8월: 연세대학교 기계전자공학부 조교수. 1999년 9월~현재: 연세대학교 기계전자공학부 부교수. 주관심분야: 영상신호처리, 음성신호처리, 패턴인식, 신경망