

클래스 정보를 이용한 PCA 기반의 특징 추출

PCA-based Feature Extraction using Class Information

박명수, 나진희, 최진영

Myoung Soo Park*, Jin Hee Na*, Jin Young Choi*

* 서울대학교 전기컴퓨터 공학부, 자동화시스템 공동연구소(ASRI)

요약

영상 데이터와 같이 큰 차원을 가지는 입력 자료들을 분류하고자 할 경우, 입력 자료의 차원을 줄일 수 있는 특징을 추출하는 전처리 과정은 매우 중요하다. 특징 추출(feature extraction)을 위해 PCA, ICA, LDA, MLP 등의 다양한 기법들이 개발되었는데, 이러한 기법들은 PCA, ICA와 같은 무감독 방식의 기법(unsupervised algorithm)과 LDA, MLP와 같은 감독 방식의 기법(supervised algorithm)으로 구분할 수 있다. 이 중에서, 감독 방식의 경우는 입력 정보와 함께 클래스 정보를 사용하기 때문에 데이터를 분류하기에 더 좋은 특징들을 뽑아낼 수 있다. 본 논문에서는 무감독 방식 기법인 PCA에 기반하면서도, 클래스 정보를 사용하여 자료 분류에 더욱 적합한 특징들을 추출할 수 있는 기법인 PCA-FX를 제안하였다. 제안한 기법에 의해 추출된 특징을 이용할 경우의 인식 성능을, Yale face database를 사용하여 다른 기법들의 성능과 비교하였다.

Abstract

Feature extraction is important to classify data with large dimension such as image data. The representative feature extraction methods for feature extraction are PCA, ICA, LDA and MLP, etc. These algorithms can be classified in two groups: unsupervised algorithms such as PCA, LDA, and supervised algorithms such as LDA, MLP. Among these two groups, supervised algorithms are more suitable to extract the features for classification because of the class information of input data. In this paper we suggest a new feature extraction algorithm PCA-FX which uses class information with PCA to extract features for classification. We test our algorithm using Yale face database and compare the performance of proposed algorithm with those of other algorithms.

Key words : 특징 추출 기법, 입력차원 감소, 감독방식 PCA, 분류 특징, PCA-FX

1. 서론

최근에는 영상자료와 같은 큰 입력 차원을 가지는 자료를 분류하기 위해, 특징 선택(feature selection) 기법이나 특징 추출(feature extraction) 기법들이 많이 이용되고 있다[1]. 이러한 기법들은 주어진 입력 차원보다 작은 차원을 가지면서 자료의 성질(characteristic)을 충분히 대표할 수 있는 특징(feature)들을 뽑아내는 것을 목표로 한다. 얻어진 특징들을 분류에 이용하면, 분류과정에서 요구되는 계산량을 크게 감소시킬 수 있고 동시에 입력에 포함된 불필요한 정보를 제거함으로써 분류 성능을 향상시킬 수 있다는 장점이 있다. 특징 선택 및 추출에 의한 입력 차원의 축소는 데이터 마이닝 등의 분야에서도 중요하게 다루어지는데, curse of dimensionality 등 큰 입력 차원으로부터 발생하는 현상들과 관련된 유용한 해결책으로 고려되고 있다[2].

널리 이용되는 특징 추출 기법으로는 PCA[3], ICA[4], LDA[5] 등의 통계적인 방식과 MLP[6, 7]를 이용하는 학

습에 기반하는 방식이 있다. 이러한 기법들은, 분류하고자 하는 자료의 소속 즉 클래스에 대한 정보를 이용하는지의 여부에 따라 두 가지로 구분할 수 있다. PCA, ICA 등의 기법은 클래스에 대한 정보를 이용하지 않는 무감독(unsupervised) 방식으로, 주어진 입력 자료의 분산 혹은 포함된 정보(information)에 기반하여 특징들을 추출한다. 이 경우 얻어지는 특징들은 입력 자료의 성질을 충실히 대표할 수 있지만, 분류에는 적합하지 않을 수 있다. LDA, MLP 등 클래스에 대한 정보를 이용하는 감독(supervised) 방식은 이러한 문제점을 해결하기 위해 개발되었다. 이 중 LDA는 분류에 적합한 특징을 추출할 수 있는 단순하면서도 강력한 방법이지만, 특정한 조건(예: 서로 다른 클래스 간의 차이가 충분히 클 것)을 만족하는 경우에만 좋은 특징 추출이 보장된다는 문제점이 알려져 있다. MLP를 이용하는 경우는 학습과정에서 지역 최소점(local minima)에 빠질 수 있다는 문제점 등의 한계로 많은 경우 좋은 성능이 보장되지 않는다. 비감독 방식의 기법들은 상대적으로 이러한 종류의 한계에서 자유로우며, 이에 비감독방식의 기법에 기반하는 감독방식 기법들의 개발이 시도되었다. 대표적인 것으로, 최근에 제안된 ICA-FX[8]와 같은 기법을 들 수 있다. ICA-FX는 ICA에 기반하고 클래스 정보를 이용하여 특징을 추출하는 기법으로, 비감독방식 기법인 PCA와 ICA에 비해 좋은 성능을 가지고 있으면서 동시에 LDA에 비해 일반적인 경우에도 적용

접수일자 : 2005년 4월 1일

완료일자 : 2005년 7월 15일

감사의 글 : 본 연구는 산업자원부 차세대 신기술개발사업(수퍼지능칩 및 응용기술 개발 과제)과 두뇌한국 2(BK21) 사업의 지원을 받아 수행되었습니다.

가능하다는 장점이 있다.

그런데 ICA-FX는 LDA 등 기존 감독방식 기법과 마찬가지로, 큰 입력차원을 가지는 실제 문제에 적용할 경우 계산량을 줄이기 위해 PCA를 이용해야 한다는 점을 주목할 필요가 있다[9, 10]. LDA를 쓸 경우에는 PCA를 이용하여 먼저 차원을 감소시키고, 다시 감소된 차원 상의 자료에 LDA를 이용하여 특징을 추출하게 된다. ICA-FX의 경우에도 PCA를 이용하여 먼저 차원을 감소시키고, ICA-FX를 이용하여 특징을 추출한다. 따라서 입력 자료의 차원을 줄이는 과정에 있어서 PCA는 아주 큰 역할을 담당하는데, 이로 인해 다음과 같은 상황이 발생한다. 첫째로 PCA의 특징의 추출 과정에 이용하는 기준(criterion)은 LDA나 ICA-FX의 기준과는 다르고 이로 인한 부작용(side effect)의 발생가능성이 있다. 즉, 먼저 PCA에 의해 감소된 차원 위에서 얻어지는 자료에는 ICA-FX나 LDA가 추출하고자 하는 정보가 배제되어 있을 가능성이 있고, 이로 인해 ICA-FX나 LDA는 적절한 특징을 추출하지 못할 수도 있다. 둘째로, 실제 LDA, ICA-FX 등의 기법들을 적용하기 위해서는 그 기법 뿐만 아니라 PCA를 추가적으로 구현해야 할 필요가 있다. 이는 구현상의 부담을 배가시킬 수 있다.

본 논문에서는 앞서 지적한 기법들의 문제점들을 가지지 않으면서, 클래스 정보를 이용하여 특징을 추출할 수 있는 새로운 감독 방식의 기법인 PCA-FX를 제안하였다. 제안한 방법은 PCA에 기반하므로 실제 문제에 적용할 때에도 PCA 하나만을 구현하는 것으로 충분하고, 구현상의 부담이 적다. 또한 PCA를 우선적으로 입력 자료의 차원을 감소시키는 데에 이용하더라도, PCA-FX는 PCA와 같은 종류의 기준에 의해 특징을 추출하므로 앞서 언급한 부작용의 가능성이 상대적으로 다른 알고리즘에 비해 낮은 편이다. 제안한 알고리즘들에 의해 추출된 특징들을 이용한 분류성능을 평가하기 위해서 Yale face database를 이용하였고, 그 결과를 다른 알고리즘들에 의해 얻어진 특징에 의한 분류성능과 비교하였다.

2. PCA-FX: PCA에 기반한 감독 방식 특징추출 기법

PCA는 입력 자료들의 분포에 따라 분산이 최대가 되도록 하는 입력 공간 상의 방향성분을 찾아내는 기법이다. 이러한 방향성분을 PC(Principal Component)라고 부르는데 PC는 일반적으로 입력 자료들을 이용하여 얻어지는 공분산행렬(covariance matrix)에 대해 고유치 문제(eigenvalue problem)를 풀어서 구할 수 있다. 고유벡터를 정규화, 즉 크기가 1이 되도록 제한하였을 때 얻어지는 해에서, 고유치는 각각 해당하는 고유벡터 위로 입력자료를 투영했을 때의 분산에 비례하는 값을 가진다. 따라서 해당 고유치의 크기가 제일 큰 고유벡터가 첫 번째 PC, 다음으로 큰 고유치에 대한하는 고유벡터가 두 번째 PC가 된다. 대부분의 경우 분산은 입력 자료의 차원에 비해 적은 수의 PC 위에 집중되는 경향을 보인다. 따라서 원래의 자료들을 이러한 소수의 PC들 위로 투영할 경우 얻어지는 값들로 구성된 벡터들(차원은 PC의 수, 개수는 원래의 자료의 개수)로 원래 자료에 담긴 대부분의 분산을 대표할 수 있다. 이 경우, 투영에 이용되는 PC들로 행렬을 구성할 수 있는데, 이 행렬을 특징공간으로의 변환행렬(transformation matrix)이라고 한다. PCA에 의한 특징추출은 이러한 변환을 구하는 과정을 포함하고, 변환에 의한 특징공간으로의 투영을 의미한다.

PCA를 통해 얻어진 특징을 추출하는 과정에서는 클래스 정보가 전혀 이용되지 않는다. 따라서, 변환행렬에 의해 추출되는 특징은 자료의 분류에는 적합하지 않을 수 있다. 그림 1과 같은 자료들이 주어지는 경우를 가정해보자. 그림에서 *와 ·으로 표시된 자료는 서로 다른 클래스에 속하는 자료다. 이 자료에서 주어진 자료의 분산이 최대가 되는 방향성분은 Z_1 이므로, PCA를 주어진 자료에 대해 수행하였을 경우 얻어지는 첫 번째 PC는 이 방향성분이 될 것이다. 이와 달리 클래스의 구분에 필요한 방향성분, 즉 다른 클래스에 속하는 자료들 간의 분산을 최대로 하는 방향성분은 Z_2 가 된다. 따라서 우리가 입력 차원을 감소시키고, 이를 통해 얻어진 특징으로 자료를 분류하고자 할 경우에 유용한 특징은, PCA를 통해 얻어진 Z_1 으로의 투영에 의해 얻어지는 특징이 아니라 Z_2 에 의해 얻어지는 특징이다. 따라서 이 경우 PCA에 의한 특징추출은 분류에는 적절하지 않다. 이와 같은 경우를 고려하면, 자료들의 적절한 분류를 위해서는 특징을 추출하는 과정에 클래스 정보를 이용할 필요가 있음이 명백해진다.

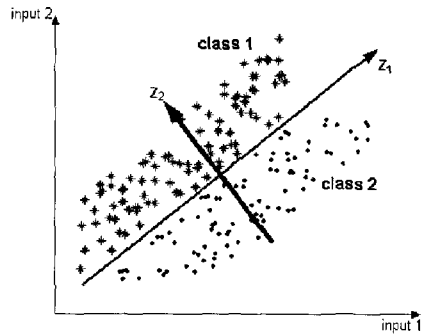


그림 1 클래스 정보가 필요한 예

본 논문에서 제안하는 PCA-FX는 클래스 정보를 이용하며 PCA에 기반한 특징추출 기법이다. 이 방법은 ICA-FX와 마찬가지로 기존의 기법들에 비해 보다 일반적인 문제에 적용가능하다. 또한 특징의 추출에 있어 PCA와 같은 기준을 사용한다. ICA-FX나 LDA에서 특징 추출에 이용하는 기준이 PCA의 것과 크게 다른 반면에, PCA-FX에서 이용되는 기준은 PCA의 것과 다르지 않다. 때문에, 실제로 영상처리 등에 응용할 경우에 ICA-FX나 LDA에서처럼 전처리를 위한 PCA와 함께 이용할 필요가 있어도, 이로 인해 발생할 수 있는 부작용의 가능성이 상대적으로 낮다.

PCA-FX는 다음과 같은 세 단계로 구성된다.

- 입력자료에 클래스 정보를 추가하는 단계
- PCA를 이용하여 특징을 추출하는 단계
- 변환행렬 W를 결정하는 단계

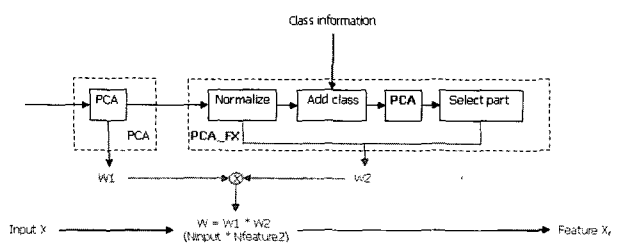


그림 2 PCA + PCA-FX에 의한 특징추출

첫 번째 단계는 입력 자료에 클래스에 대한 정보를 추가하는 과정이고, 두 번째 단계는 첫 번째 단계의 결과로 얻어진 입력자료에 PCA를 적용하는 단계로, 공분산 행렬을 구하는 과정과 이로부터 고유치 문제를 풀어 분산을 많이 포함하는 PC를 선택하는 과정을 포함한다. 마지막 단계는 두 번째 과정에서 얻어진 PC들로부터 특징추출을 위한 변환행렬을 결정하는 단계이다. 그림 2는 전처리 단계로 PCA를 이용하고, PCA-FX를 이용하여 특징을 추출하는 과정을 보여주고 있다. 각각의 단계에 대한 세부사항은 아래와 같다.

2.1 입력자료에 클래스 정보를 추가하는 단계

일단 클래스의 수를 N_{class} 라고 하고, 입력자료에 대한 클래스 정보를 아래와 같이 표현하자.

$$C(X) = [C_1, C_2, \dots, C_{N_{class}}] \quad (1)$$

여기서, C_i 는 X 가 i 번째 클래스에 속하면, P_i 라는 값을 가지고 그렇지 않은 경우에는 N_i 라는 값을 갖는다. P_i 와 N_i 는 모든 입력자료 X 에 대하여, C_i 의 평균=0, 표준편차=1이 되도록 하는 값이다. 식이 $2i$, 변수가 $2i$ 개이므로 값을 결정할 수 있다.

예: 입력자료 X_1, X_2, X_3 가 주어지고 이들 각각이 속하는 클래스가 1, 2, 2라고 하자. 이 경우 입력 자료에 대한 클래스 정보는 $C(X_1)=[P_1, N_2], C(X_2)=[N_1, P_2], C(X_3)=[N_1, P_2]$ 와 같이 표현된다. 그리고 이 경우, C_1 과 C_2 의 평균과 표준편차가 각각 1과 0이 되도록 하는 $P_1=\sqrt{6}/3, P_2=\sqrt{6}/6, N_1=-\sqrt{6}/6, N_2=-\sqrt{6}/3$ 이다.

앞서의 과정을 통해 구해진 클래스 정보 $C(X)$ 를 X 에 추가하여 아래와 같은 새로운 입력 X_{aug} 을 구한다.

$$X_{aug} = \begin{bmatrix} X_{1, norm} & C(X_1) \\ X_{2, norm} & C(X_2) \\ \vdots & \vdots \\ X_{M, norm} & C(X_M) \end{bmatrix} \quad (2)$$

여기서 M 은 입력 자료의 수를 의미한다. 그리고 $X_{i, norm}$ 은 각 차원의 평균과 표준편차가 주어진 입력에 대해서 각각 0과 1이 되도록 조정된 입력 자료를 의미한다. 입력 자료의 각 차원의 분산보다 주어진 클래스 정보에 해당하는 차원의 분산에 더욱 민감하게 PC가 결정되도록, 입력 자료의 각 차원의 평균과 표준편차를 조정한다. 모든 $X_i(i=1, 2, \dots, M)$ 에 대해 각 입력차원의 평균과 표준편차 \bar{X} 와 $\sigma(X)$ 를 구하고, $X_{i, norm}$ 를 계산한다.

$$[X_{i, norm}]_j = ([X_i]_j - \bar{X}_j) / [\sigma(X)]_j \quad (3)$$

여기서, $i=1, 2, \dots, M$ 이고, $j=1, 2, \dots, N_{input}$ 이다. $[\cdot]_j$ 는 벡터의 j 번째 차원을 의미한다. 이를 통해 얻어지

- 1) 실제 클래스 정보 $C(X)$ 의 표준편차는 X 의 표준편차에 비해 아주 작은 값이어도 큰 문제가 없다. 실험에 의하면, $C(X)$ 의 각 차원의 표준편차가 0.001, X 의 각 차원의 표준편차가 0.1일 경우에도 얻어지는 특징은 원래의 것과 동일하였다. 중요한 것은 X 의 각 차원의 표준편차를 동일하게 하는 것이다.

는 X_{aug} 의 차원은 $M \times (N_{input} + N_{class})$ 으로 주어진다. \bar{X} 과 $\sigma(X)$ 의 차원은 $1 \times N_{input}$ 이다.

2.2 PCA에 기반한 특징 추출 단계

X_{aug} 에 [3]에서와 같은 일반적인 PCA 기법을 적용하면 차원이 $1 \times (N_{input} + N_{class})$ 인 $N_{input} + N_{class}$ 개의 PC들을 구할 수 있다. 여기서, 주어진 입력 차원 N_{input} 을 원하는 차원 $N_{feature} < N_{input}$ 으로 줄이기 위해, 얻어진 PC들 중에서 큰 고유치에 해당하는 것부터 $N_{feature}$ 개의 PC를 선택한다. 얻어진 PC들의 각 성분을 2.1에서 얻어진 $\sigma(X)$ 를 이용하여 아래와 같이 변환한다.

$$[PC_{i, norm}]_j = [PC_i]_j / [\sigma(X)]_j \quad (4)$$

여기서 $i=1, 2, \dots, N_{feature}, j=1, 2, \dots, N_{input}$ 이다. PC의 나머지 N_{class} 개의 차원에 대해서는 $[PC_{i, norm}]_j = [PC_i]_j$ 로 한다.

구해진 PC를 행렬의 형태로 쓰면 다음과 같다.

$$W_{aug} = [PC_{1, norm}^T \quad PC_{2, norm}^T \quad \dots \quad PC_{N_{feature}, norm}^T] \quad (5)$$

$$= \begin{bmatrix} W_{input} \\ W_{class} \end{bmatrix}$$

여기서 W_{aug} 의 차원은 $(N_{class} + N_{input}) \times N_{feature}$ 이다. W_{input} 과 W_{class} 는 각각 입력자료와 클래스 정보에 해당하는 부분을 의미하는데, 각각 $N_{input} \times N_{feature}$ 이고, $N_{class} \times N_{feature}$ 차원을 가진다.

2.3 변환행렬 W를 결정하는 단계

입력자료 X 에 대해서 클래스 정보 $C(X)$ 와 W_{aug} 를 이용하면, 새로운 PC를 기저로 하는 특징공간 위로 투영한 결과 $X_{feature}$ 를 얻을 수 있다. 관계식은 아래와 같다.

$$X_{feature} = [X \ C(X)] W_{aug}$$

$$= [X \ C(X)] \begin{bmatrix} W_{input} \\ W_{class} \end{bmatrix} \quad (6)$$

$$= X W_{input} + C(X) W_{class}$$

변환행렬을 구하기 위해 이용한 입력자료에 대해서는 그것에 대한 클래스 정보를 알고 이용할 수 있다. 그러나, 변환행렬을 구한 후에 분류를 위해 주어지는 입력자료에 대해서는 클래스 정보를 알 수 없다. 따라서 위의 식을 이용하여 새로운 자료에 대한 $X_{feature}$ 를 구할 수는 없다. 앞서 언급한 바와 같이 클래스에 해당하는 정보의 표준편차는 입력자료에 비해 아주 작은 값으로 조정될 수 있다. 그러한 경우 W_{class} 에 의해 투영된 분산의 양은 W_{input} 에 의해 아주 작다. 클래스 정보의 표준편차를 충분히 작게 하면 클래스 정보에 의한 분산을 무시할 정도로 작게 만들 수 있고, 이러한 경우 아래와 같은 근사식을 이용하는 것이 가능해진다.

$$X_{feature} \approx XW \quad (7)$$

여기서 W 는 최종적으로 얻어진 변환행렬이며 W_{input} 과 같다.

얻어진 변환행렬 W 에는 클래스 정보가 직접적으로 나타

나지 않는다. 하지만 앞서 PCA를 적용하는 과정에서 입력 자료보다 주어진 클래스 정보에 더욱 민감하게 PC가 결정되도록 하였고, 이를 통해 W_{aug} 의 결정에 클래스 정보가 이용되었다. 따라서 이에 포함된 W_{input} 도 입력정보를 클래스 정보의 분산이 큰 방향, 즉 분류를 위해 유리한 방향으로 투영시키게 된다. 따라서 그러한 방향성분으로 구성된 주어진 변환행렬 $W = W_{input}$ 을 이용하면 분류에 적합한 특징을 얻을 수 있다[2].

3. 실험 결과

주어진 알고리즘의 성능평가를 위해 Yale database [11]를 이용하였다. 주어진 database에서 특징을 추출한 뒤에 이를 이용하여 얼굴인식을 하고 정확성을 비교하였다. Yale database는 15사람에 대한 165개의 흑백 이미지로 구성되어 있으며, 각 사람마다 11가지의 표정을 포함되어 있다. Yale database에는 얼굴에 맞추어 잘려진 것[tightly-cropped]과 전체 얼굴을 담고 있는 것[full face]의 두 가지 종류가 있는데, 실험에서는 첫번째 database를 이용하였다. 차원을 그림 3에서와 같이 21×30로 맞추고, 다시 각각을 1×630의 차원을 가진 벡터로 변환하여 제안한 기법을 위한 입력으로 사용하였다. 따라서 입력차원은 630이고, 자료의 수는 165개가 된다.



그림 3 Yale database 샘플

제안한 기법의 성능을 평가하기 위해 leave-one-out 방식이라는 일종의 교차평가(cross validation) 방법을 이용하였다. 전체 자료가 M 개 있을 경우에 우선 1개의 자료를 선택하고, 그 자료를 제외한 나머지 $M-1$ 개의 자료를 이용하여 분류기(classifier)를 구성하고 제외했던 한 개의 자료를 구성된 분류기를 이용하여 분류한다. 그리고 동일한 과정을 M 개의 자료 각각에 대해 반복하여, 얻어진 분류결과의 오차율(error rate)을 성능평가의 기준으로 삼는 방법이다. 제안한 특징추출 기법 및 비교하기 위한 다른 특징추출 기법의 성능평가를 위하여, 분류기는 공통적으로 단순한 Euclidean 거리에 의한 nearest neighborhood classifier를 이용하였다. 그리고 입력을 각기 다른 특징추출 기법을 이용하여 구함으로써 성능상의 차이를 확인하고자 하였다. 공통적인 실험과정은 아래와 같고 이는 그림 2에 주어진 과정과 같다.

- PCA를 통과하여 특정한 개수의 PC를 얻고, 그 결과에 대해 PCA-FX / LDA / ICA-FX를 적용하여 특정한 개수의 최종 특징을 얻는다.

2) 실제 자료에 대한 실험을 통해서, 입력 자료를 구해진 W_{class} 로 투영하여 얻어지는 값들의 분산이 W_{input} 을 통해 얻어지는 분산에 비하여 매우 작은 값을 가지게 됨을 확인할 수 있다. 클래스 정보의 표준편차를 충분히 작은 값으로 조정하면 그러한 경향은 더욱 강화된다.

실험을 통해 관찰하고자 하는 결과는 다음과 같다.

- PCA를 통해 얻은 PC의 수와 PCA-FX에 의해 얻은 최종 특징 수에 따른 분류오차율의 변화
- PCA, LDA, ICA-FX에 의한 특징들의 성능 비교

3.1 변수에 따른 분류 오차율의 변화 분석

PCA-FX를 통해 얻어진 특징들의 분류 오차율과 PCA를 통해 얻은 PC의 수의 관계를 확인하기 위해서, 각각 다른 수의 PC에 대해 특징을 얻고, 각각의 경우의 오차율을 구하였다. 동시에, 특정한 수의 PC에 대해서 다양한 숫자의 특징을 구하여 분류오차율을 결정함으로써 최종 특징 수와 분류오차율의 결과도 확인하고자 하였다. 결과는 표 1에 주어진 바와 같다. 표 1에서 굵게 표시된 숫자는 동일한 PC의 수에 대해 분류오차율이 제일 작은 경우를 의미한다.

우선, PCA를 통해 얻어지는 특징의 성능은 첫 번째 PCA를 통해 얻은 PC의 수에 따라 다르게 나타남을 확인할 수 있다. PC의 수가 증가할수록 전반적으로 오차율은 작아지는 경향을 보이지만, 항상 일관되게 나타나지는 않는다. 다음으로, PCA를 통해 얻어지는 특징의 성능은 최종 특징의 수가 증가할수록 감소했다가 증가하는 경향을 보인다. 이는 PC의 수에 관계없이 대부분의 경우에 확인된다. 즉 PC의 수가 많은 경우와 적은 경우 모두 PC의 개수가 증가할수록 처음에는 오차가 감소하다가 어떤 문턱값(threshold value)을 넘으면 증가하는 경향을 보인다. 표 1을 가지고 판단한다면, 문턱값은 약 13으로 전체 클래스 수에 가까운 값이라고 예상할 수 있다. 그림 4는 표 1의 결과 중 일부를 그래프로 표현한 결과이다. 앞서 기술한 경향성을 보다 뚜렷이 확인할 수 있다.

표 1 PCA-FX의 PC, 특징 수에 따른 오차율

PC 개수	특징의 개수						
	9	10	11	12	13	14	15
18	13.94						
19	13.33	11.15					
20	13.94	09.70	10.91	10.91		16.97	16.97
21	10.91	10.91	09.09	10.30		10.91	
22	12.73	10.91	11.52	10.30	11.52	11.52	
23	11.52	12.73	12.12	11.52	10.91	12.73	
24	11.52	09.70		11.52	06.06	06.67	09.09
25	10.30	12.73		12.73	10.30	12.12	
26	11.52	10.91	09.70	09.70	08.48	09.09	09.70
27	10.30	10.91		11.52	07.27	07.27	09.70
28	10.30	08.48	09.09	07.88	09.70	07.88	09.09
29	12.73	10.30	08.48	06.67	09.70	06.67	09.70
30	11.52	11.52		08.48	07.88	07.88	10.91
40			09.09	10.30	08.48	09.09	09.09

이 결과를 통해서도 앞서와 마찬가지로 PC의 개수가 일정한 경우, 오차율이 최종 특징의 수가 증가함에 따라 감소하다가 특징의 수가 어떤 임계치에 이르면 다시 증가함을 볼 수 있다. 그리고 오차율이 최소가 되는 특징의 개수는 항상 클래스의 수인 15보다 작게 나타남과 동시에 일정한 경향성

을 보이며 나타남을 확인할 수 있다. 실험결과에 따르면 PC의 수가 24이고 특징의 수가 13인 경우에 오차율이 가장 낮게 나타났다.

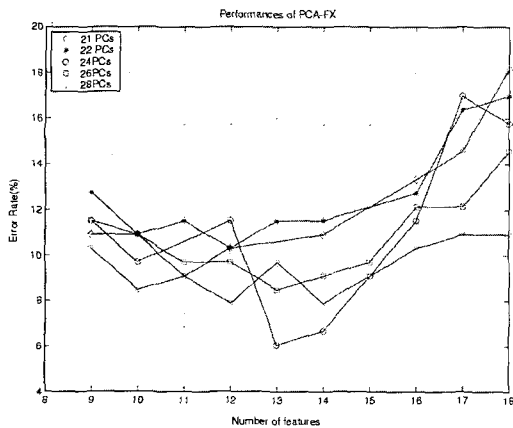


그림 4 특징의 수와 오차율과의 관계

3.2 PCA-FX와 PCA, LDA, ICA-FX를 이용하여 얻어지는 특징들의 성능 비교

앞서 PCA-FX를 통해 얻은 특징의 성능은 PC의 수와 최종 특징의 수에 따라 달라짐을 확인하였다. 실험 결과 이러한 경향은 PCA, LDA, ICA-FX에서도 동일하게 나타남을 확인할 수 있다. 그러나 특징 및 PC의 수에 따른 오차율의 변화는 기법에 따라 다르게 나타난다. 예를 들어, 그림 5에서 보는 바와 같이 ICA-FX는 PCA-FX와 마찬가지로 특정한 PC의 수에 대해서 특징의 수가 증가할수록 오차율이 감소하다가 증가하지만, 증가하기 시작하는 시점은 PCA-FX와 다르다. 그림 5에서는 ICA-FX에 비해 PCA-FX의 오차율이 보다 급격하게 변화하는 것을 알 수 있다. 그 결과, 최소 오차율을 보이는 경우 특징의 수가 ICA-FX에 비해 적은 것을 알 수 있다. 이와 같은 경향성은 PCA에서 구해지는 PC의 수가 다른 경우에 대해서도 유사하게 나타난다.

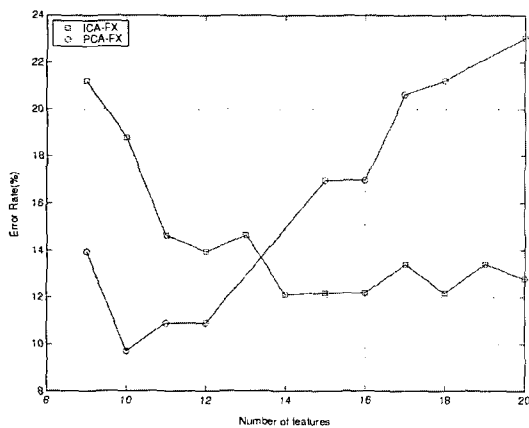


그림 5 PCA-FX와 ICA-FX의 오차율 비교 (PC의 개수가 20인 경우)

그림 4에서 나타난 경향성에 주목하여, PCA, LDA, ICA-FX에서도 PC와 특징의 개수를 변화시켜 가면서 최소 오차율이 되는 변수값을 구하는 실험을 하였다. 그 결과로

얻은, 각 기법에 있어서의 최소 오차율과 최소 오차율이 나타나는 변수값은 표 2와 같다.

표 2 PC와 특징의 수에 따른 최소 오차율

알고리즘	PC 수	특징 수	최소 오차율
PCA	22	9	27.27%
LDA	24	13	12.73%
ICA-FX	30	20	6.06%
PCA-FX	24	13	6.06%

표 2의 결과로부터 PCA-FX가 기존의 특징추출 기법들에 비해 적은 수의 PC를 통해 얻어진 적은 수의 특징들을 가지고 제일 좋은 성능을 보임을 확인할 수 있다.

4. 결 론

본 논문에서는 기존의 특징추출 기법들의 문제점을 지적하고, 이러한 문제점들을 극복할 수 있는 특징추출 기법인 PCA-FX를 제안하였다. PCA-FX는 PCA를 이용하여 특징을 추출하기 위한 변환행렬을 구하는 과정에서, 클래스에 대한 정보를 입력 정보에 추가하여 이용함으로써 좋은 분류 성능을 가지는 특징을 추출하는 기법이다. 실제로 Yale face database를 이용하여 제안한 기법의 성능을 다른 기법들과 비교평가해 본 결과, 기존의 다른 특징추출 기법들에 비해 보다 낮은 분류 오차율을 가지는 적은 수의 특징을 얻을 수 있었다. 동시에 이 경우에, PCA로부터는 적은 수의 PC만을 이용하므로 계산량 또한 다른 기법에 비해 절약되는 것을 확인할 수 있었다.

향후 과제로는 첫 번째로, PCA를 통해 얻어진 특징의 수와 오차율 간의 관계를 이론적으로 분석할 필요가 있다. 이 과정을 통해 PCA에 있어서 큰 분산을 가지는 방향성분을 추출함으로써 이루어지는 정보의 압축과정에 대한 통찰을 얻고, 이에 기반하여 PCA와 ICA의 차이점을 명확히 하고 각각의 장단점을 결합한 새로운 특징추출 기법을 개발하고자 한다. 두 번째로, PCA-FX의 성능에 관한 앞의 표를 참조하여 오차율과 특징 수와의 관계를 이론적으로 추정할 수 있는지 살펴보고자 한다. 만약 오차율을 최소로 하기 위해 필요한 특징의 수가 몇 개인지 이론적으로 알 수 있다면, trial-and-error의 과정 없이 최소의 오차율을 갖게 하는 PC와 특징의 수를 구할 수 있을 것이다. 실제 시스템에서 이러한 기법을 이용하고자 할 경우에 이러한 수치의 결정은 매우 중요한 문제라고 할 수 있다.

참 고 문 헌

- [1] K. J. Cios, W. Pedrycz, and R. W. Swinarski, Data mining methods for knowledge discovery, chapter 9, Kluwer Academic Publishers, 1998.
- [2] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Advances in Knowledge Discovery and Data Mining, AAAI and the MIT Press, 1996.
- [3] I. T. Jolliffe, Principal Component Analysis, Springer-

- Verlag, 1986.
- [4] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6., June 1995.
- [5] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, 2nd edition, 1990.
- [6] H. Lu, R. Setiono, and H. Liu, "Effective data mining using neural networks," *IEEE Transaction on Knowledge and Data Engineering*, vol. 8, no. 6, Dec. 1996.
- [7] R. Setiono, and H. Liu, "A connectionist approach to generating oblique decision trees," *IEEE Transaction on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol. 29, no. 3, June 1999.
- [8] Nojun Kwak and Chong-Ho Choi, "Feature extraction based on ICA for binary classification problems," *IEEE Transaction on Knowledge and Data Engineering*, Vol. 15, No. 6, pp. 1374-1388, Nov. 2003.
- [9] Nojun Kwak, Chong-Ho Choi, and Narendra Ahuja, "Face recognition using feature extraction based on independent component analysis," *ICIP2002*, Rochester, Sep. 2002.
- [10] W. Zhao, R. Chellappa and A. Krishnaswamy, "Discriminant analysis of principal components for face recognition," *Proceedings of 3rd IEEE International Conference on Automatic Face and Gesture Recognition*, pp.336-341, April 1998.
- [11] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, July 1997.



나진희(Na, Jin Hee)

2003년 : 서울대 전기공학부 학사
 2003년~현재 : 서울대학교 대학원
 전기컴퓨터 공학부 석박사 통합과정

관심분야 : 신경회로망, 영상처리
 Phone : +82-2-872-7283
 E-mail : jhna@neuro.snu.ac.kr



최진영(Choi, Jin Young)

1982년 : 서울대 제어계측공학과 학사
 1984년 : 서울대 제어계측공학과 석사
 1993년 : 서울대 제어계측공학과 박사
 1984~1994년 : 한국전자통신연구소
 연구원

1998~1999년 : University of California,
 Riverside 객원교수

1994년~2004년 : 서울대 전기공학부 부교수
 2004년~현재 : 서울대 전기공학부 교수

관심분야 : 적응제어, 신경회로망
 Phone : +82-2-880-8372
 E-mail : jychoi@neuro.snu.ac.kr

저 자 소 개



박명수(Park, Myoung Soo)

1998년 : 서울대 전기공학부 학사
 2000년 : 서울대학교 대학원
 전기컴퓨터공학부 석사
 2000년~현재 : 서울대학교 대학원
 전기컴퓨터공학부 박사과정

관심분야 : 인공지능, 기계학습, 신경회로망, 영상처리
 Phone : +82-2-872-7283
 E-mail : mspark@neuro.snu.ac.kr