

# SVDD 기반의 점진적 학습기능을 갖는 얼굴인식 시스템

## Face Recognition System with SVDD-based Incremental Learning Scheme

강 우 성<sup>1</sup>, 나 진 희<sup>2</sup>, 안 호 석<sup>3</sup>, 최 진 영<sup>†</sup>

Woo-Sung Kang<sup>1</sup>, Jin Hee Na<sup>2</sup>, Ho Seok Ahn<sup>3</sup>, Jin Young Choi<sup>†</sup>

**Abstract** In face recognition, learning speed of face is very important since the system should be trained again whenever the size of dataset increases. In existing methods, training time increases rapidly with the increase of data, which leads to the difficulty of training with a large dataset. To overcome this problem, we propose SVDD (Support Vector Domain Description)-based learning method that can learn a dataset of face rapidly and incrementally. In experimental results, we show that the training speed of the proposed method is much faster than those of other methods. Moreover, it is shown that our face recognition system can improve the accuracy gradually by learning faces incrementally at real environments with illumination changes.

**Keywords** : Face Recognition System, Incremental Learning, Support Vector Domain Description.

### 1. 서 론

시각, 감정, 행동, 환경 인식 등을 통한 인간-로봇 상호작용(human-robot interaction)은 최근 활발하게 연구되고 있는 분야 중 하나이다. 그 중에서도 시각은 로봇이 사람의 얼굴을 인식하고 그에 따르는 행동이나 반응을 취하는데 있어서 가장 기본적으로 필요한 정보이기 때문에, 이를 해결하기 위한 여러 기법이 제안되었다.

이와 같은 인공 시각 시스템 중 얼굴 인식 시스템은 크게 얼굴검출과 특징추출, 얼굴분류 알고리즘으로 크게 나눌 수 있는데, 각각 알고리즘이 사용되기 위해 중요시 되는 부분이 각각 다르다. 얼굴 검출 알고리즘에서는 얼굴을 빠르게 찾아내는 것이 중요하므로, 얼굴 검출에 사용될 알고리즘의 학습 속도보다는 주로 얼굴 검출 속도를 중요시하고 있다<sup>[2][13][14]</sup>. 이는 얼굴검출을 하기 위해 오프라인(offline)으로 한번 학습하고 그 이후에는 재 학습할 필요가 없기 때문이다. 이와는 다르게 얼굴분류에서는

빠른 학습속도를 중요시 하는데, 이는 인식될 사람이 늘어날 때마다 시스템을 여러 번 재 학습해야 하기 때문이다. 본 논문에서는 얼굴검출의 성능개선에 대해서는 다루지 않으며, 얼굴인식시스템의 학습속도를 개선하고 조명 등 외부 환경의 변화에 적응하기 위한 점진적 학습 알고리즘(Incremental Learning Algorithm)을 제안하도록 한다. 아울러 얼굴검출, 특징추출 방법을 동시에 구현하여 실제 환경에서 동작할 수 있는 얼굴 인식 시스템을 완성하여 성능을 평가하였다.

얼굴인식에 적용된 점진적 학습 방법으로 계층적 판별회기 모형(HDRM; Hierarchical Discriminating Regression Model)<sup>[8]</sup>, 점진적 주성분분석 (IICA; Incremental Principal Independent Component Analysis)<sup>[9]</sup>과 같은 방식이 제안되었다. HDRM 은 판단 트리 (Decision Tree)를 근간으로 하고 있기 때문에 클래스의 수가 늘어 감에 따라 복잡도가 높아 질 수 있고 탐색 속도가 느려 질 수 있다. IPCA 방법은 해가 여러 개 존재하여 학습되는 데이터의 순서에 따라 성능이 많이 달라 질 수 있다. 가우시안 혼합모델 (gussian mixture model)<sup>[20]</sup> 역시 점진적 학습방법으로 사용할 수 있지만 적절한 가우시안 성분의 개수를 정하는 것이 어렵고 학습시간이 시간이 많이 걸린다.

※ 본 논문은 BK21 및 ASRI 의 지원을 받아 연구되었음

† 교신저자 : 서울대학교 전기. 컴퓨터공학부 교수

<sup>1</sup> 서울대학교 전기컴퓨터공학부(박사과정)

<sup>2</sup> 서울대학교 전기컴퓨터공학부(석박사 통합 과정)

<sup>3</sup> 서울대학교 전기컴퓨터공학부(석사과정)

본 논문에서는 다른 접근 방법을 시도한다. SVDD (Support Vector Domain Description)<sup>[7]</sup>를 이용하여 해당 클래스에 대한 데이터만을 가지고 다중 클래스 분류기를 생성할 수 있는 방법을 시도하였다. 여기서 SVDD는 데이터의 도메인만을 표시하기 때문에 테스트 데이터가 겹치는 도메인에 포함되거나 각 클래스의 도메인 영역에 포함되지 않는 경우는 클래스를 판별할 수 없다. 이를 위해 클래스 판별함수를 새로 제안하였으며 이 판별함수의 중요 매개변수를 결정하는 방법을 제시 하였다. 이 방법을 사용함으로써 SVM(Support Vector Machine)<sup>[11]</sup>을 이용하는 방식보다 다중 분류문제를 해결하기에 적합하고, 클래스의 증가에 따른 학습 시간의 증가를 피할 수 있다.

조명 등이 변하는 실제 환경에서 제안된 시스템의 유용성을 보이기 위해, 11명의 사람을 대상으로 인식 성능을 테스트하였다. 조명이 변하는 조건을 고려하기 위해, 아침 9시부터 저녁 9시까지 실내에서 5일간 점진적 학습을 적용하여 95% 이상의 인식 성능을 획득할 수 있었다.

## 2. 얼굴인식 시스템 개요

제안된 얼굴 인식 시스템은 그림 1과 같이 이미지 획득 시스템, 얼굴검출, 특징추출 및 얼굴분류의 4가지 부분으로 구성되어 있다. 이미지 획득 시스템으로써, 실시간 얼굴 영상을 얻기 위해 웹 카메라를 사용하였고 이곳에서 획득된 이미지 프레임에서 얼굴 영역을 검출하기 위해 viola<sup>[2]</sup>에 의해 제안된 얼굴 검출 알고리즘을 사용하였다. 이곳에서 검출된 이미지는 30 x 20의 크기로 정규화

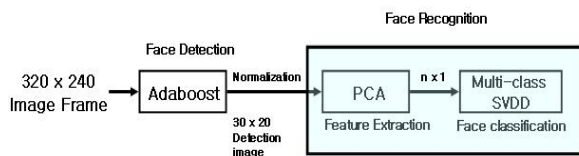


그림 1. 얼굴인식 시스템 구조



그림 2. 구현된 얼굴인식 시스템의 GUI

되고 이 이미지에서 다시 특징 추출을 위해 PCA<sup>[9]</sup>를 사용하였으며, 얼굴의 분류를 위한 학습을 위해 제안된 다중 클래스 SVDD (Support Vector Domain Description)<sup>[7]</sup>를 사용하고 이 결과를 이용하여 자료데이터의 분포를 추정하는 방법을 제안하여 사용하였다. 그림 2는 구현된 실제 얼굴 인식 시스템을 나타낸다.

## 3. 얼굴검출 알고리즘

제안된 구조에서는 얼굴의 검출을 위해 viola가 제안한 얼굴 검출방식<sup>[2]</sup>을 적용하였다. 이는 그림 3과 같은 단순한 유사 haar 특징(haar-like feature) 사용하는 weak classifier를 Adaboost 학습<sup>[10]</sup>에 의해 계산된 각 weak classifier의 가중치로 선형 결합하여 strong classifier를 구성하는 방식이다.

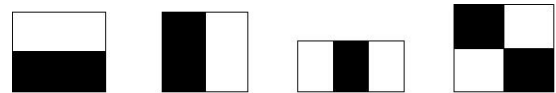


그림 3. 유사 haar 특징

그리고 적분 이미지(integral image)를 사용하기 때문에 이미지를 탐색할 윈도우는 크기에 상관없이 항상 같은 수의 값을 참조하게 함으로써 빠른 연산이 가능하다. 또한 그림 4와 같이 모든 계층(layer)을 통과한 부분을 얼굴 후보 영역으로 줄여나가는 직렬 구조를 사용함으로써, 실시간으로 얼굴을 검출하는 것이 가능하다.

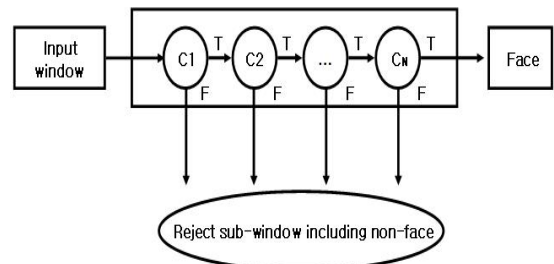


그림 4. 얼굴 검출을 위한 cascade 구조

## 4. 얼굴 특징추출 알고리즘

특징 추출 단계는 입력 영상의 차원을 줄이면서 분류에 적합한 특징들을 얻어낼 수 있는 변환행렬을 구하는 과정을 말한다. 이 경우 PCA<sup>[9]</sup>와 같은 방법들을 사용할 경우 주어진 데이터가 바뀌거나 새로운 데이터가 추가될 때마다 이에 대한 통계적 특성을 새로 얻어야 한다는 단

점이 있다. 특히 데이터의 차원이 크고 데이터의 수가 많아질수록 특징 추출을 위한 시간은 기하급수적으로 늘어나기 때문에 점진적으로 얼굴을 학습하고 테스트 하는 실제 얼굴인식 시스템에 적용하는 데에는 어려움이 따른다. 또한 기존의 방식에 의하면 새로운 데이터가 들어올 때마다 이에 알맞은 변환행렬을 새롭게 구해야 하고, 이에 따라 이전 특징 공간에 사용되었던 정보는 사용할 수 없기 때문에 분류기의 학습을 위해 새로 변환된 데이터와 추가 데이터를 모두 사용해야 하므로 결국 점진적 학습은 불가능하게 된다.

본 논문에서는 오프라인 상에서 많은 수의 다양한 얼굴 데이터를 이용하여 PCA의 변환행렬을 미리 구하여 얼굴 데이터가 들어왔을 때 점진적으로 얼굴을 학습할 수 있는 방법을 제안한다. 변환행렬을 미리 구하여 사용하면, 새로운 데이터가 들어올 때마다 특징 추출 단계에서 모든 데이터들의 통계적 특성을 구할 필요가 없고, 고정된 특징 공간에서만 연산을 하므로 classifier가 점진적으로 데이터들을 학습하게 할 수 있도록 할 수 있다.

오프라인에서 PCA의 변환행렬을 얻기 위해서 총 10155 장의 얼굴 이미지를 사용하였다. 실제 얼굴이 검출되는 환경에는 조명, 검출된 얼굴의 크기, 포즈, 표정 등의 변화가 다양하므로 이를 반영하기 위해 다양한 환경에서 얻은 얼굴들을 20x30 차원으로 축소시켜서 오프라인 학습을 위한 얼굴 데이터베이스를 구축하였다. 변환행렬을 구하기 위해 포즈 및 조명 변화를 고려한 Yale database<sup>[15]</sup>, 표정의 변화를 고려한 Jaffe database<sup>[16]</sup>, 얼굴의 회전을 고려한 ORL database<sup>[17]</sup> 등의 벤치마크 데이터와 인터넷 상으로 수집된 데이터를 사용하였다. 오프라인으로 얻은 변환행렬의 타당성을 검증하기 위해서는 Yale Face Data와 같은 벤치마크 데이터를 사용하였다. 투영 (projection) 하고자 하는 학습 데이터들의 집합을  $X$  라고 하고,  $X$  로 부터 얻은 변환행렬과 오프라인 데이터들의 집합으로부터 얻은 변환행렬을 각각  $W^T, W_{fixed}^T$  라고 할 때 각각의 변환 행렬로 투영된 특징 벡터들의 집합은 식 (1), (2)와 같이 나타낼 수 있다.

$$P_X = W^T X \quad (1)$$

$$P_{X\_fixed} = W_{fixed}^T X \quad (2)$$

이 경우  $W^T$  는  $X$  의 분산 정보를 최대화하는 기저 벡터(basis vector)들의 집합으로 주어지는데  $W^T$  와  $W_{fixed}^T$  로 투영된 특징 벡터들의 분산 정보를 식 (3)을 이용하여

비교해 보면 오프라인에서 다양한 얼굴들을 이용하여 얻은 변환행렬의 유용성을 보여줄 수 있다.

$$\text{var\_ratio} = \frac{\sum_{i=1}^{\text{feature}} \text{var}(P_{X\_fixed}(i))}{\sum_{j=1}^{\text{feature}} \text{var}(P_X(j))} \quad (3)$$

여기서 feature는 데이터의 투영을 위해 사용한 기저 벡터의 개수를 의미하는데, 기저 벡터의 수에 따른 분산 비 (variance ratio)값의 변화는 그림 5와 같이 나타난다.

예를 들어, 그림 5에서 보면 기저 벡터의 개수, 특징 벡터의 차원을 100 이상으로 설정하면 입력 데이터들의 분산에 대한 정보를 90% 이상 얻어낼 수 있음을 알 수 있다. 이로써, 완전한 데이터를 보존 할 수는 없지만 어느 정도 유용한 특징들을 추출함으로써, 빠르고 점진적인 학습이 가능하게 된다.

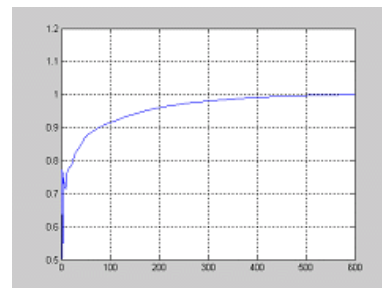


그림 5. 기저 벡터 개수에 따른 분산 비 값의 변화

## 5. 얼굴분류 알고리즘

### 5.1. 다중 클래스 분류기

얼굴인식은 기본적으로 여러 사람을 분류하는 것을 목적으로 하기 때문에, 다중 클래스를 분류할 수 있는 분류기(classifier<sup>[3][4][5][6]</sup>)를 사용해야 한다. 최근 얼굴 인식에 많이 쓰이는 분류기로 좋은 일반화 성능 때문에 SVM(Support Vector Machine)<sup>[11]</sup>이 많이 사용되고 있는데, 이는 원래 이진 분류를 기반으로 설계된 방식이기 때문에, 다중 클래스 분류를 위해서는 SVM을 조합하는 방법이 필요하다. 이에 여러 알고리즘이 제안되어 왔는데 one-against-all<sup>[3][6][11]</sup>, one-against-one<sup>[4][6]</sup>, DAG (Directed Acyclic Graph)<sup>[5][6]</sup> 등이 대표적인 학습방식들이다. one-against-all 알고리즘은 구현이 아주 간단하다는 이점이 있지만, 한 클래스에 해당하는 분류기를 학습하기 위해 모든 클래스의 데이터를 사용해야 하므로, 클래스 혹은 학습될 사람이

수가 늘어날 때마다 전체 시스템의 학습 시간이 기하급수적으로 늘어나게 된다<sup>6)</sup>.

one-against-one과 DAG 방식은 가능한 서로 다른 한 쌍의 클래스를 모두 고려하여 학습하는 방식으로써, 속도가 one-against-all보다 훨씬 빨라서 실제 응용에 가장 적합한 방식으로 알려져 있다<sup>6)</sup>. 하지만 이 방식 역시 클래스 혹은 학습될 사람의 수가 늘어날 때마다 분류기의 수가 기하급수적으로 늘어나므로, 시스템 구성을 위한 연산 량의 증가문제를 해결할 수 없다. 게다가 시스템의 자원이 한정된 경우에는 전체 학습 자체가 불가능한 경우가 발생할 수도 있다. 결국 이진 분류기를 사용하는 다중 클래스 분류기는 실제 시스템에 적용하기에 많은 어려움이 따른다. 따라서, 한 클래스에 해당하는 분류기를 학습하기 위해, 데이터의 중복 사용을 되도록이면 피하도록 하는 것이 중요하다.

본 논문에서는 SVDD<sup>7)</sup>를 이용하여 해당 클래스에 대한 데이터만을 가지고 다중 클래스 분류기를 생성할 수 있는 방법을 개발하였다. SVDD<sup>7)</sup>는 SVM과 같이 이차계획법(quadratic program)을 사용하고 클래스의 분류를 위해 지지 벡터(support vector)를 이용하는 것은 같지만, 특징 공간에서 데이터를 찾는 포함하는 가장 작은 원을 찾는 것을 목표로 한 목적함수를 갖는다는 점이 다르다. 여기서 전체 다중 클래스 분류기를 학습하기 위해서는 각 클래스에 대한 분류기를 구하기 위해 각 클래스의 데이터로 (4)의 식과 같은 이차계획법의 해를 구하면 된다.

$$\min_{R^m, a^m, \xi_j^m} (R^m)^2 + C \sum_j \xi_j^m, \quad \|\phi^m(x_j) - a^m\|^2 \leq (R^m)^2 + \xi_j^m \quad \forall j, \xi_j^m \geq 0. \quad (4)$$

여기서  $R^m$  는 특징 공간에서  $m$  번째 클래스를 나타내는 가장 작은 원의 반지름을 나타내며,  $a^m$  는 해당 영역의 중심을 나타낸다. 그리고  $\phi(x)$  는 입력 데이터를 특징 공간으로 사상하는 함수이다. 위의 해로부터 해당 클래스의 모든 데이터를 포함하는 원의 중심으로부터 임의의 테스트 데이터까지의 거리는 다음과 같다.

$$R^m(z) = K^m(z, z) - 2 \sum_i \alpha_i^m K(z, x_i^m) + \sum_{i,j} \alpha_i^m \alpha_j^m K(x_i^m, x_j^m), \quad m = 1, \dots, k.$$

여기서 SVDD는 데이터의 도메인만을 표시하기 때문에 테스트 데이터가 겹치는 도메인에 포함되거나 각 클

래스의 도메인 영역에 포함되지 않는 경우는 클래스를 판별할 수 없다. 이를 위해 클래스 판별함수를 (5)와 같이 제안하였다.

$$D^m(z) = \exp\left(\frac{-R^m(z)}{(\sigma^m)^2}\right), \quad m = 1, \dots, k. \quad (5)$$

여기서  $\sigma^m$  은 확률분포의 분산에 해당하는 값으로 영역의 경계를 결정하는 중요한 매개변수이다.  $\sigma^m$  은 지지 벡터에서 일정한 값을 갖도록 다음의 식에 의해 결정한다.

$$\exp\left(-\frac{R^m(x_{\text{support}}^m)}{(\sigma^m)^2}\right) = S, \quad \sigma^m = \sqrt{\frac{R^m(x_{\text{support}}^m)}{\ln S}}, \quad (6)$$

여기서  $S$  는 지지 벡터에서의 판별값을 나타낸다. 그림 6은  $\sigma^m$  를 결정하는 예를 나타낸 그림이다. 실험에서는  $S$  값을 0.01로 설정하였다.

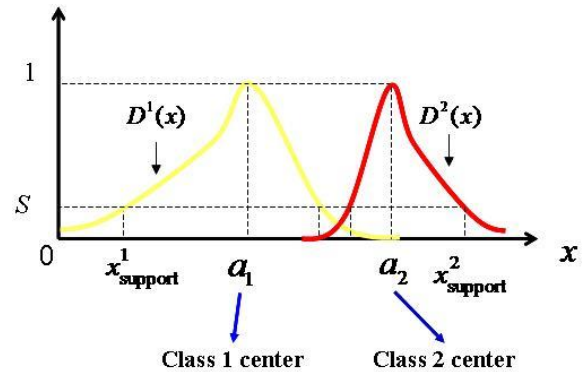


그림 6.  $\sigma^m$  파라미터 결정의 예

$D^m(z)$  은 테스트 데이터가 특징 영역의 중심으로부터 얼마나 가까운가를 나타낸다. 예를 들어 입력 데이터가 클래스  $m$ 에 가장 가까울수록  $D^m(z)$  의 값은 큰 값을 가지게 되며, 멀수록 작은 값을 갖게 된다. 이 방식은 데이터의 왜도(skewness)를 표현할 수 있는 일반화된 확률 분포 함수와 유사한 형태를 갖게 된다. 그림 7은  $D^m(z)$  에 대한 값을 등고선(contour map)을 사용하여 나

타낸 결과이다. 최종적으로 테스트 데이터가 어떤 영역에 속하는지 결정하기 위해서, (7)의 식을 이용한다.

$$\text{class of } x \equiv \arg \max_{m=1, \dots, k} D^m(x) \quad (7)$$

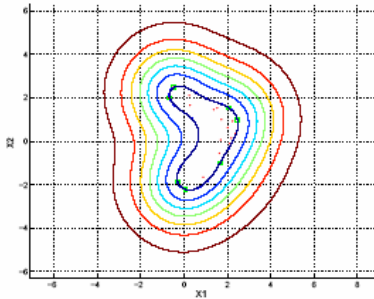


그림 7. 2차원 입력 공간에서의 등고선

위의 기법을 사용함으로써, 제안된 시스템은 새로운 사람이 추가될 때, 해당 사람에 대해서만 추가 학습하고, 기존의 학습된 결과는 보존함으로써, 점진적 학습이 가능하다. 다음의 표 1과 2는 실제 제안된 방법이 기존 방법에 비해 얼마나 학습 속도가 개선되었는지를 나타낸다.

[표 1] Yale face database에서의 학습속도 비교

Training : Test	one-against-all SVM	one-against-one SVM	proposed method	Multilayer Perceptron
4 : 7	10.29 s	3.25 s	0.24 s	990.60 s
5 : 6	16.81 s	4.95 s	0.27 s	989.24 s
6 : 5	25.71 s	6.61 s	0.28 s	1024.13 s
7 : 4	38.08 s	8.29 s	0.30 s	1300.98 s

[표 2] Yale face database에서의 인식성능 비교

Training:Test	one-against-all SVM	one-against-one SVM	proposed method	Multilayer Perceptron
4 : 7	89.7(3.3) %	91.0(3.0) %	90.6(3.3) %	81.3(4.5) %
5 : 6	90.2(3.3) %	91.2(3.5) %	90.2(4.0) %	82.5(4.2) %
6 : 5	90.7(3.8) %	91.7(4.2) %	90.5(3.2) %	85.3(3.6) %
7 : 4	91.9(3.9) %	92.1(4.2) %	93.3(4.1) %	88.8(4.8) %

표 1과 2의 결과로부터, 학습 속도는 기존의 방식에 비해 훨씬 빠르며, 데이터가 증가할 경우에 학습 속도가 기존 방식에 비해 많이 느려지지 않는다. 또한 기존 방식과도 비교할 만한 인식 성능을 가짐으로써, SVDD를 다중 클래스 분류 문제에 적용하는 것은 적합하다고 결론 지을 수 있다.

## 5.2. 점진적 학습

SVDD를 다중 클래스 분류기로 확장한 방식<sup>8)</sup>은 추가된 클래스 혹은 사람에 대한 해당 클래스만을 학습함으로써, 학습 연산 량을 감소시켰다. 하지만, 한 사람에 대한 정보가 많을 경우에는 이 때 연산 량의 증가를 피할 수 없다. 그래서, 해당 클래스의 묘사를 위해, 필요한 정보만을 학습에 참여시키고, 학습에 필요 없거나 분류기를 나타내는데 필요 없는 정보들은 메모리 낭비를 막기 위해, 제거해야 할 필요가 있다. 그래서 우리는 점진적 학습을 위해 지지 벡터 학습(support vector learning)에서 분류기를 표현하는 정보는 지지 벡터만으로 구성되어 있다는 특성을 이용하였다. 그래서 새로운 사람이 들어오게 될 경우 보존된 지지 벡터와 필요한 추가된 정보만을 재 학습에 사용하기로 하였다. 이는 전체 데이터 학습 후 구한 해와 지지 벡터만을 다시 학습하여 구한 해가 항상 같기 때문이다.

그리고, 추가된 데이터는 항상 분류기의 형성에 영향을 끼치는 것이 아니므로, 추가된 정보 중에도 필요한 정보만을 선택하여 학습에 참여시킨다. 즉, 추가로 학습되어야 할 데이터가 이전에 구해진 영역에 포함되는지 테스트하여 만약 그렇다면, 이 데이터는 학습될 정보에서 제외시킨다. 이 역시, 포함된 영역에 있는 데이터는 학습 후 분류기에 아무런 영향을 끼치지 않기 때문이다. 이와 같이 지지 벡터를 제외한 정보는 제거하고, 추가된 클래스 혹은 사람에 대한 정보만을 이용하되, 이 중에서도 학습에 영향을 줄 수 있는 데이터만 골라내어 학습함으로써, 점진적 학습이 가능하다. 이로써, 기존 방법보다 훨씬 빠르고 연산 량을 줄이는 것이 가능하다.

## 6. 실험 결과

제안된 시스템의 실제 시스템에서의 유용성을 보이기 위해, 우리는 11명의 사람을 대상으로 인식 성능을 테스트하였다. 5일간 실험을 시행하였으며, 조명이 변하는 조건을 고려하기 위해, 조명 조건이 변하는 실내에서 얼굴 인식 성능을 평가하였다. 우선, 3사람을 등록하여 인식 성능을 평가하고, 잘못 인식된 사람만을 추가 학습 한 후, 다음 날 다시 8명의 사람을 추가하여 학습하고 인식성능 테스트하였다. 테스트는 오전 10시부터 오후 8시 사이에 무작위로 테스트 하였으며, 하루 동안의 인식 성능을 그림 8에 막대 그래프로 나타내었다. 처음에는 학습된 시각과 테스트 시간이 다르므로 조명 조건이 달라서, 인식 성능은 좋지 못한 결과를 나타낸다. 하지만 틀린 사람만을 점진적으로 학습함으로써, 점차 인식 성능이 향상되었다.



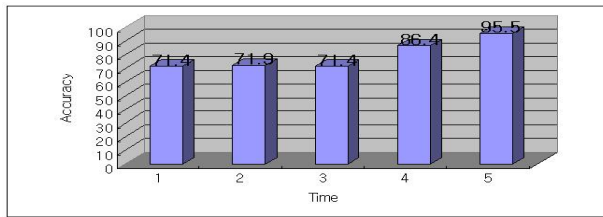


그림 8. 제안된 방법에 의한 인식성능 향상을 나타낸 결과

그림 9는 각각의 개인에 대한 인식 성능의 향상을 5일간 테스트 하여, 나타낸 그래프이다. 첫째 날 학습된 3명에 대한 인식 성능은 다음날 8명의 사람이 추가되어, 사람 Na로 표기된 사람을 제외한 나머지 두 사람에 대한 성능이 떨어짐을 볼 수 있다. 그 후 이 두 사람을 추가 학습한 후 성능을 조사하였더니, 두 사람의 성능이 좋아지는 대신, 다른 4명에 대한 인식률이 떨어지게 되었다. 하지만, 다시 틀린 사람을 점진적으로 학습하고 테스트한 결과 모든 사람에 대한 인식 성능은 증가하게 되었다. 그 후 더 좋은 인식 성능을 위해, 잘못 인식된 결과를 갖는 사람에 대해 다시 학습한 결과 두 사람만을 제외하고, 나머지 사람에 대해서는 완전한 인식을 하게 되는 결과를 얻게 되었다. 결국, 점진적 학습을 통해 전체 인식 성능은 인식률이 향상하는 추이를 보인다. 따라서, 점진적으로 계속 학습하게 되면, 다양한 조명 효과에서도 인식 성능이 향상되는 것을 볼 수 있다.

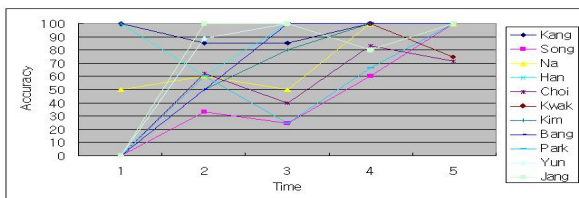


그림 9. 개개인에 대한 인식 성능의 향상

## 7. 결 론

기존의 얼굴 인식 시스템은 항상 학습될 사람이 추가될 때마다, 학습을 위한 연산량이 기하급수적으로 늘어나게 되어 학습 속도가 느려지게 되므로, 실제 환경에서 유용하게 사용되기 어렵다. 또한 실제 시스템에서는 학습하기 위해 사용될 수 있는 자원이 제한되어 있으므로, 연산량의 증가는 결국 시스템 학습이 불가능하게 되는 경우를 초래할 수 있다. 따라서, 최대한 중복될 수 있는 연산량을 줄임으로써, 학습 속도를 개선하는 것은 중요한 과제중의 하나라고 할 수 있다. 그래서 우리는 점진적 학습 방법으로 필요한 정보만을 학습에 사용함으로써 연산

량을 줄이고 속도를 개선할 수 있는 시스템을 제안하였다. 그리고 실험 결과로부터 점진적 학습을 통해 제안된 시스템이 조명 효과에 어느 정도 강인한 효과를 얻을 수 있다는 것을 보임으로써, 제안된 시스템은 실제 시스템 적용에 유용한 기법임을 결론 지을 수 있다.

## 참 고 문 헌

- [1] 나진희, 안호석, 박명수, 최진영, “시스템 재설정 및 진화를 위한 지능형 아키텍처 개발”, 한국 퍼지 및 지능 시스템학회 논문지.
- [2] Paul viola, Micheal J. Jones, “Robust Real-Time Face Detection”, International Journal of Computer Vision, 57(2), 137. 154, 2004.
- [3] Bottou, L., Cortes, C., Denker, J., Drucker, H., Guyon, I., Jackel, L., LeCun, U., Muller, U., Sackinger, E., Simard, P.m Vapnik, V. “Comparison fo Classifier Methods: A Case Study in Handwriting Digit Recognition. Proc. Int. Conf. Pattern Recognition” 1994.
- [4] KreBel, U. “pairwise Classification and Support Vector Machines – Advance in kernel Methods”, MIT press, Cambridge MA 1999.
- [5] J. C. Platt, N. Cristianini, and J. Shawe- Taylor, “Large Margin DAGs for Multiclass Classification”, in Advanced in Neural Information Processing System, Cambridge, MA: MIT Press, vol. 12, pp. 547-553, 2000.
- [6] Hsu, C. S., Lin, C. J. “A Comparison of Methods for Multiclass Support Vector Machines” IEEE Trans. Neural Networks, 2. Vol. 13, 2002.
- [7] Tax, D. M. J., Duin, R. P. W. “Support Vector Domain Description”, Pattern Recognition Letters 20 1999.
- [8] Woo-Sung Kang, Ki Hong Im, Jin Young Choi, “SVDD-based Method for Fast Training of Multiclass Support Vector Classifier”, ISNN 2006, LNCS 3971, 991-996.
- [9] I. T. Jolliffe, Principal Component Analysis, Springer-Verlag, 1986.
- [10] Freund, Y and Schapire, R. E. “A decision-theoretic generalization of on-line learning and an application to boosting”, In Computational Learning Theory, Springer-Verlag, pp 23-37.
- [11] Vladimir N. Vapnik, “Statistical Learning Theory”, John Wiley & Sons, Inc, 1998.
- [12] Osuna, E., Freund, R., and Girosi, F., “Training Support Vector Machine-An application to face detection”, In Proceedings of IEEE conference on Computer Vision and Pattern Recognition. 1997a.

- [13] Osuna, E., Freund, R., and Girosi, F., "Training Support Vector Machine-An application to face detection", In Proceedings of IEEE conference on Computer Vision and Pattern Recognition. 1997b.
- [14] H. Rowley, S. Baluja, and T. Canade, "Neural-network-based Face Detection", IEEE Transaction on Pattern Analysis and Machine Intelligence", 1998.
- [15] <http://cvc.yale.edu/projects/yalefaces/yalefaces.html>
- [16] <http://www.kasrl.org/jaffe.html>
- [17] <http://www.cl.cam.ac.uk/Research/DTG/attarchive/facedatabase.html>
- [18] J. Weng, C.H. Evans, and W.S. Hwang, "An incremental learning method for face recognition under continuous video stream", In Procs. of the Fourth International Conference on Automatic Face and Gesture Recognition, Grenoble, France, March 2000.
- [19] S. Ozawa, S. L. Toh, S. Abe, S. Pang, N. Kasabov., "Incremental Learning of feature space and classifier for face recognition", Neural Networks 2005.
- [20] Richard O. Duda, Peter E. hart, David G. Stork, PATTERN CLASSIFICATION, JOHN WILEY & SONS, INC, 2001.



**나 진 희**

2003 서울대학교 전기컴퓨터공학부(학사)  
 2003~현재 서울대학교 전기컴퓨터공학부(석박사 통합 과정)  
 관심분야: 패턴인식, 신경회로망, 영상처리



**안 호 석**

2005 성균관대 정보통신공학부(학사)  
 2005~현재 서울대학교 전기컴퓨터공학부(석사과정)  
 관심분야: 지능로봇, 인공지능, 신경회로망, 영상처리



**강 우 성**

2002 고려대학교 제어계측공학과(학사)  
 2004 서울대학교 전기컴퓨터공학부(석사)  
 2004~현재 서울대학교 전기컴퓨터공학부(박사과정)

관심분야: 패턴인식, 얼굴인식, 신경회로망, 커널머신



**최 진 영**

1982 서울대학교 제어계측공학과(학사)  
 1984 서울대학교 제어계측공학과(석사)  
 1993 서울대학교 제어계측공학과(박사)  
 1984~1994 한국전자통신연구소 연구원

1998~1999 University of California, Riverside 객원교수  
 1994~2004 서울대학교 전기공학부 부교수  
 2004~현재 서울대학교 전기공학부 교수  
 관심분야: 적응제어, 신경회로망, 패턴인식