

RBFNNs 패턴분류기와 객체 추적 알고리즘을 이용한 얼굴인식 및 추적 시스템 설계

Design of Face Recognition and Tracking System by Using RBFNNs Pattern Classifier with Object Tracking Algorithm

오 승 훈* · 오 성 권* · 김 진 율*
(Seung-Hun Oh · Sung-Kwun Oh · Jin-Yul Kim)

Abstract - In this paper, we design a hybrid system for recognition and tracking realized with the aid of polynomial based RBFNNs pattern classifier and particle filter. The RBFNN classifier is built by learning the training data for diverse pose images. The optimized parameters of RBFNN classifier are obtained by Particle Swarm Optimization(PSO). Testing data for pose image is used as a face image obtained under real situation, where the face image is detected by AdaBoost algorithm. In order to improve the recognition performance for a detected image, pose estimation as preprocessing step is carried out before the face recognition step. PCA is used for pose estimation, the pose of detected image is assigned for the built pose by considering the featured difference between the previously built pose image and the newly detected image. The recognition of detected image is performed through polynomial based RBFNN pattern classifier, and if the detected image is equal to target for tracking, the target will be traced by particle filter in real time. Moreover, when tracking is failed by PF, Adaboost algorithm detects facial area again, and the procedures of both the pose estimation and the image recognition are repeated as mentioned above. Finally, experimental results are compared and analyzed by using Honda/UCSD data known as benchmark DB.

Key Words : Particle Filter, RBFNNs pattern classifier, AdaBoost, Particle Swarm Optimization(PSO)

1. 서 론

네트워크 및 정보 발달로 인해 많은 분야에서 보안의 중요성이 화두가 되고 있다. 하지만 현재 사용되는 인식 기술은 사용자가 암기를 하고 있어야 하며, 분실 및 도용이 발생할 수 있다는 문제점이 존재한다. 이러한 문제를 보완하기 위해 지문, 홍채 인식과 같은 생체 인식 기술이 여러 분야에서 연구가 진행 중에 있으며, 생체 인식 기술은 인간이 가지는 고유한 신체적 특징을 이용한다는 점에서 기존의 인식 기술에서 발생하는 분실 및 도용과 같은 문제점을 해결하는 것이 가능하다. 본 논문에서는 생체 인식 기술 중 하나인 얼굴인식을 수행하며, 얼굴인식은 다른 생체인식 기술과 달리 인식 대상자가 인식부에 접촉할 필요가 없는 비접촉식으로 사용자의 불편함이 적다는 장점이 존재한다. 허나 정면 얼굴이미지만을 이용하여 인식을 수행하는 정적 상황에서의

인식은 포즈 변화에 대한 인식 성능의 저하가 발생할 수 있으며, 이를 보완하기 위해 본 논문에서는 동적 상황에서 인식을 수행하며, 인식 결과를 기반으로 검출된 얼굴 영역의 추적을 수행하는 시스템을 제안한다.

2차원 얼굴인식은 얼굴의 명암도를 이용하여 인식을 수행하기 때문에 조명변화를 최소화하기 위해 전처리 단계에서 히스토그램 평활화를 수행하여 얼굴 이미지의 명암도를 보정한다. 차원 축소 알고리즘인 주성분 분석법(PCA)을 통해 입력 데이터의 차원 축소를 수행한 이후 패턴분류기의 입력으로 사용이 되며, 다항식 기반 RBFNNs 패턴분류기를 이용하여 인식 성능을 확인한다. 학습 데이터베이스는 포즈 변화에 강인한 얼굴 인식 성능을 확인하기 위해 정면 얼굴과 다른 포즈(좌측, 우측)의 얼굴 이미지를 이용하여 포즈별 학습 모델을 구성한다. 각 학습 모델은 최적화된 RBFNNs 패턴분류기를 이용하여 모델에 최적화된 파라미터를 획득하며, 최적화 알고리즘은 입자군집 알고리즘(PSO)을 이용하였다.

비디오 영상 내에서 AdaBoost 알고리즘을 이용하여 인식 및 추적을 위한 얼굴 이미지를 검출하게 되며, 검출된 얼굴 이미지는 포즈 분류 이후 인식을 수행하여 사용자가 원하는 추적 대상 여부를 판단한다. 검출된 대상이 추적 대상이라 판단되면 검출된 얼굴 영역을 기반으로 객체 추적 알고리즘인 Particle Filter를

† Corresponding Author : Dept. of Electrical Engineering, The University of Suwon, Korea

E-mail : ohsk@suwon.ac.kr

* Dept. of Electrical Engineering, The University of Suwon, Korea
Received : November 05, 2014; Accepted : April 29, 2015

수행하여 대상의 얼굴 영역을 지속적으로 추적하게 되며, 이때 유클리디안 거리의 오차를 근거로 하여 추적 성공 여부를 판단하게 된다. 만약 추적이 실패하였다 판단한 경우 다시 AdaBoost를 통해 얼굴 영역의 재검출 및 재인식을 수행하게 되며, 이로 인해 기존의 객체 추적 알고리즘과 달리 별도의 영역 지정이 없이도 동일한 대상을 지속적으로 추적하는 것이 가능하다는 장점이 있다.

본 논문 2장에서는 데이터베이스 구축 및 각 포즈별 모델의 학습 과정에 대하여 설명하며, 3장에서는 테스트 이미지 검출 및 인식 결과를 통한 영역 추적에 대하여 설명한다. 4장에서는 실험에 사용한 데이터 및 각 실험에 대한 전체적인 개요를 설명하고, 마지막으로 결론에 대하여 설명한다.

2. 데이터베이스 구축 및 포즈별 학습 수행

본 논문에서는 기존의 정적인 얼굴 인식에서의 단점을 보완하기 위해 객체 추적 알고리즘과 지능형 패턴분류기인 RBFNNs 패턴분류기를 융합한 인식 및 추적 시스템을 제안하며, 시스템의 전체적인 흐름은 아래 그림 1과 같다.

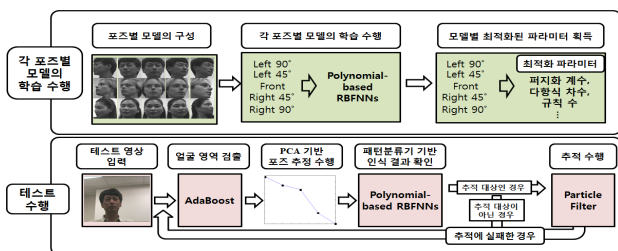


그림 1 제안된 얼굴인식 및 추적 시스템 흐름도
 Fig. 1 Flowchart of proposed face recognition and tracking system

동적인 상황에서 테스트 이미지를 입력받아 인식을 수행하기 때문에 정면 얼굴 이미지만을 이용하여 학습 데이터를 구성할 경우 정확한 인식 성능을 확인하는 것이 어렵다. 2장에서는 이러한 문제점 해결을 위한 정면 얼굴 이외의 다른 포즈의 얼굴 이미지를 입력받아 학습 데이터를 구성하고, RBFNNs 패턴분류기를 이용한 각 포즈별 모델의 학습 과정에 대하여 설명한다.

2.1 포즈변화에 대한 학습 데이터베이스의 구성

기존의 얼굴인식은 획득한 2차원 이미지의 명암도를 기반으로 인식을 수행하며, 따라서 주어진 학습 데이터와 데이터 데이터의 명암도 차이가 클수록 정확한 인식 성능을 획득하는 것이 어렵다. 기존의 얼굴인식에서는 인식 대상의 정면 얼굴 이미지를 이용하여 학습 데이터를 구성하고, 테스트 이미지 역시 인식 대상자가 카메라를 응시하고 있는 정면 얼굴 이미지를 획득하여 인식을 수행한다. 학습 데이터와 테스트 데이터 모두 같은 포즈의 정면 얼굴 이미지를 사용하기 때문에 별도의 성능 저하 없이 인식 성능을 확인하는 것이 가능하다. 하지만 이와 같은 인식 시스템에

서는 인식 대상자가 카메라를 응시하고 있어야 하며, 또한 다른 포즈의 얼굴 이미지가 입력되었을 경우 정확한 인식 성능 확인이 어렵다는 단점이 존재한다.

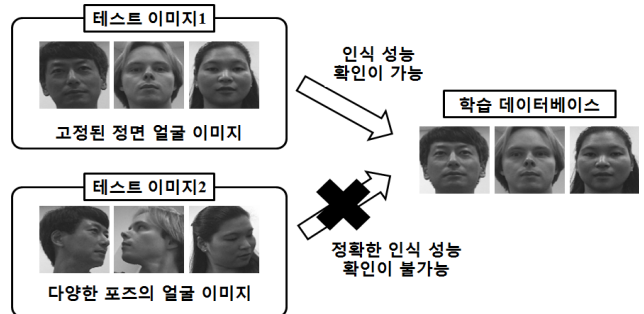


그림 2 정면 테스트 이미지와 다양한 포즈의 테스트 이미지
 Fig. 2 Test image of front and diverse pose

본 논문에서는 비디오 영상 내에서 AdaBoost 알고리즘을 통해 검출된 얼굴 이미지를 이용하여 인식 및 추적을 수행한다. 이때 AdaBoost 알고리즘을 통해 검출된 얼굴 이미지는 대부분 정면 얼굴 이외의 다른 포즈의 얼굴 이미지가 입력되며, 이와 같은 테스트 이미지는 기존의 학습 데이터를 이용할 경우 정확한 인식 성능을 확인하는 것이 어렵다. 따라서 다양한 포즈에서의 인식을 수행하기 위해서는 다양한 포즈를 충족시키는 포즈별 학습 데이터의 구성이 중요하며, 이를 위해 좌측, 정면, 우측을 포함하는 총 5방향의 포즈별 이미지를 이용하여 학습 데이터를 구성하였다.



그림 3 얼굴인식 데이터베이스의 구성
 Fig. 3 Configuration of face recognition database

위 그림 3과 같이 좌측 90°, 좌측 45°, 정면, 우측 45°, 우측 90°의 이미지를 이용하여 다양한 포즈에 대한 학습 데이터를 구성하였으며 이로 인해 다양한 포즈의 테스트 이미지가 입력되어도 인식 성능을 확인하는 것이 가능하다. 보다 세밀한 각도에서의 이미지를 이용하여 다양한 포즈별 학습 데이터를 구성할 경우 더욱 정확한 인식 성능을 확인하는 것이 가능하지만, 현실적으로 모든 포즈를 충족시키는 학습 데이터를 구축하는데 어려움이 존재하며, 또한 많은 양의 데이터는 학습 속도를 저하시키는 요인이 된다. 따라서 본 논문에서는 좌측, 정면, 우측을 대표하는 포

즈의 이미지를 이용하여 5개의 포즈별 데이터를 구성하였으며, 실제 테스트 이미지는 인식을 수행하기 주성분 분석법 기반 포즈 추정을 통해 가장 근사한 포즈를 판별하여 인식을 수행함으로써 기존의 얼굴인식 보다 포즈 변화에 강인한 인식 성능을 확인하는 것이 가능하다. 포즈 추정 방법에 대한 자세한 설명은 본 논문 3.2절에서 다루기로 한다.

2.2 RBFNNs 패턴분류기를 이용한 각 포즈별 데이터의 학습 수행

본 논문에서는 포즈변화에 강인한 인식 성능을 확인하기 위해 기존의 정면 얼굴만을 이용한 데이터베이스를 확장한 포즈별 데이터를 이용하여 데이터베이스를 구성하였으며, 테스트 이미지 입력 시 가장 유사한 포즈를 판별하여 인식을 수행함으로써 신뢰성 있는 인식 성능을 확인하는 것이 가능하다. 이를 위해 각 포즈별 데이터에 대한 학습을 별개로 수행하며, 최적화 알고리즘인 PSO를 통해 각 포즈에 해당하는 최적화된 파라미터 값을 획득한다. 학습을 수행하기 전 전처리 단계로 주성분 분석법을 이용하여 데이터의 차원을 축소하며, 이후 RBFNNs 패턴분류기를 이용하여 포즈별 데이터의 학습을 수행한다.

2.2.1 주성분 분석법(PCA)을 이용한 데이터의 차원 축소

포즈별 이미지를 이용하여 구성된 포즈별 학습 데이터는 각 이미지 픽셀 값의 명암도 값으로 구성되어 있어 그대로 인식에 사용할 경우 많은 양의 데이터로 인해 학습 성능 및 속도의 저하가 발생하게 된다. 이러한 문제 해결을 위해 학습을 수행하기 전 전처리 과정에서 데이터의 차원을 감소시키는 과정을 거쳐야 하며, 본 논문에서는 특징 추출 알고리즘인 주성분 분석법 [1](PCA : Principal Component Analysis)을 통해 데이터의 차원 축소를 수행한다.

주성분 분석법은 대표적인 선형변환 특징추출 방법으로 데이터의 공분산 행렬을 이용하여 구한 특징데이터에서 입력 데이터의 차원보다 작은 값의 차원을 이용함으로써 실제 입력보다 작은 양의 특징을 사용하는 방법이다. 주성분 분석법의 가장 큰 목적은 변화 후의 특징데이터가 변환 전의 실제 데이터의 정보를 최대한 유지하도록 하는 것이며, 주성분 분석법을 이용한 차원 축소 과정은 아래 단계와 같다.

[Step 1] 인식후보의 얼굴 벡터 집합 구성

$$S = \Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_M \tag{1}$$

[Step 2] 평균과 분산을 기준으로 이미지 정규화 수행

$$\Gamma_{ij} = (\Gamma_{ij} - \mu_i) \times \frac{\sigma}{\sigma_i} + \mu \tag{2}$$

[Step 3] 평균 얼굴 벡터를 계산

$$\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n \tag{3}$$

[Step 4] 인식후보 얼굴 벡터와 평균 얼굴 벡터와의 차이 벡터를 계산

$$\Phi_i = \Gamma_i - \Psi \tag{4}$$

[Step 5] 인식후보 얼굴에서 공분산 행렬을 계산

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \cdot \Phi = A \cdot A^T \tag{5}$$

[Step 6] M개의 고유벡터에서 가장 큰 고유값을 갖는 M'개만을 선택

$$C = U \Lambda U^T \tag{6}$$

[Step 7] 각 인식후보 얼굴과 고유벡터와의 사영을 통해 가중치를 획득

$$\Omega^T = [w_1, w_2, w_3, \dots, w_{M'}] \tag{7}$$

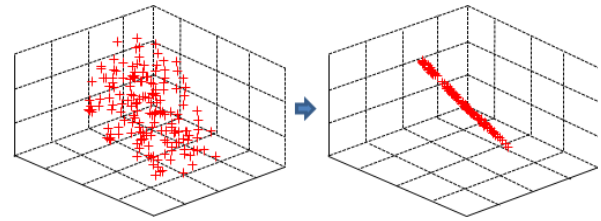


그림 4 주성분 분석법(PCA)
Fig. 4 Principal Component Analysis(PCA)

2.2.2 다항식 기반 RBFNNs 패턴분류기의 구조

인식 성능을 확인하기 위한 다항식 기반 RBFNNs 패턴분류기 [2,3,4,5]에 대하여 설명한다. 기본적인 신경회로망은 인간의 두뇌를 구현하기 위한 알고리즘으로, RBFNNs 패턴분류기는 신경회로망의 구조를 기반으로 하여 입력층, 은닉층, 출력층 이 3가지로 구분할 수 있다. 입력 데이터는 전처리 과정을 수행하여 인식에 사용하기 적합한 저차원 데이터로 축소되어, 은닉층의 입력으로 들어가게 되며, 은닉층에서 활성화수로 변환된 출력은 은닉층과 출력층 사이의 연결가중치 값과 곱해서 출력층의 최종 출력으로 얻어진다.

본 논문에서 사용한 다항식 기반 RBFNNs 패턴분류기의 구조는 위 그림 5와 같다. 기존의 RBFNNs와 비교하였을 때 구조적 모듈로는 동일하게 입력층, 은닉층, 출력층 이 3가지로 구분되나,

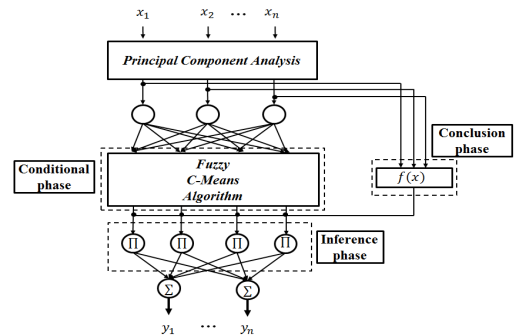


그림 5 다항식 기반 RBFNNs 패턴분류기의 구조
Fig. 5 Architecture of polynomial based RBFNNs pattern classifier

가능적 모듈로서 조건부, 추론부, 결론부 이 3가지로 구분이 가능하다. 또한 기존의 조건부의 활성함수로 사용하였던 가우시안 함수 대신 FCM(Fuzzy C-Means)클러스터링 알고리즘의 적합도 값을 사용함으로써 입력 데이터의 특성을 좀 더 잘 반영할 수 있도록 개선하였다. 또한 결론부의 연결가중치를 상수항에서 일차식과 이차식, 그리고 변형된 이차식과 같은 다항식의 형태로 확장하여 사용한다는 특징을 가지고 있다.

[Type 1] 1차 선형 추론(Linear)

$$f_j(x_1, \dots, x_k) = a_{j0} + \sum_{i=1}^k a_{ji}x_i \quad (8)$$

[Type 2] 2차 선형 추론(Quadratic)

$$k=2: f_j(x_1, \dots, x_k) = \quad (9)$$

$$a_{j0} + \sum_{i=1}^k a_{ji}x_i + \sum_{i=1}^k a_j(k+1)x_i^2 + a_{(2k+1)}x_1x_2$$

$$k \geq 3: f_j(x_1, \dots, x_k) = \quad (10)$$

$$a_{j0} + \sum_{i=1}^k a_{ji}x_i + \sum_{i=1}^k a_{j(k+1)}x_i^2 + a_{(2k+1)}x_1x_2 + \dots + a_{k(k+3)/2}x_{(k-1)}x_k$$

[Type 3] 변형된 2차 선형 추론(Modified Quadratic)

$$k=2: f_j(x_1, \dots, x_k) = a_{j0} + \sum_{i=1}^k a_{ji}x_i + a_{(2k+1)}x_1x_2 \quad (11)$$

$$k \geq 3: f_j(x_1, \dots, x_k) = \quad (12)$$

$$a_{j0} + \sum_{i=1}^k a_{ji}x_i + a_{(2k+1)}x_1x_2 + \dots + a_{k(k+3)/2}x_{(k-1)}x_k$$

$x = [x_1, x_2, \dots, x_k]$ 에서 k 는 입력변수의 수, R^j 는 j 번째 퍼지 규칙($j=1, \dots, c$), c 는 퍼지 규칙 수이며, $f_j = (x_1, \dots, x_k)$ 는 j 번째 규칙에 대한 후반부로서 j 번째 퍼지 규칙에 대한 로컬 모델이다.

모델의 출력은 아래 식처럼 구해진다.

$$\hat{y} = \sum_{j=1}^c u_j f_j(x_1, \dots, x_k) \quad (13)$$

이와 같이 다항식 형태의 연결가중치를 사용함으로써 아래 식의 퍼지 규칙 표현과 같은 언어적 관점에서의 해석이 가능해졌다.

$$\text{IF } x \text{ is } A_i \text{ Then } f_{ij}(x) \quad (14)$$

전반부 동정을 위해 FCM(Fuzzy C-Means)클러스터링 알고리즘을 사용하며, FCM 알고리즘은 비슷한 패턴, 속성, 형태 등의 기준을 통해 데이터를 분류하는 알고리즘으로, 데이터와 각 클러스터와의 거리를 기준으로 소속정도를 측정하여 데이터를 분류한다. 이를 이용하여 다항식 기반 RBFNNs 패턴분류기의 은닉층 활성함수 형태를 표현하였으며, 수행과정은 아래와 같다.

[Step 1] 클러스터의 개수, 퍼지화 계수를 선택하고 소속함수 $U(0)$ 를 초기화 한다.

$$U^{(0)} = \left\{ u_{ik} \in [0, 1], \sum_{i=1}^c u_{ik} = 1 \forall k, 0 < \sum_{k=1}^n u_{ik} < n \forall i \right\} \quad (15)$$

[Step 2] 각 클러스터에 대한 중심 벡터를 구한다.

$$u_{ij} = \frac{\sum_{k=1}^n (u_{ik})^m x_{kj}}{\sum_{k=1}^n (u_{ik})^m} \quad (16)$$

[Step 3] 중심과 데이터와의 거리를 계산하며, 이를 통해 새로운 소속함수($U(1)$)를 계산한다.

$$d_{ik} = d(x_k - v_i^{(r)}) = \left[\sum_{j=1}^l (x_{kj} - v_{ij}^{(r)})^2 \right]^{1/2} \quad (17)$$

$$u_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{ik}}{d_{jk}} \right)^{2/(m-1)}} \quad (18)$$

[Step 4] 오차가 허용범위 안에 도달하면 알고리즘을 종료하고, 그렇지 않으면 [Step 2]로 돌아간다.

$$\| U^{(r+1)} - U^{(r)} \| \leq \epsilon \quad (19)$$

기존 퍼지 추론 시스템에서는 후반부 파라미터 동정 시 주로 각 규칙에 파라미터를 동시에 구하는 전역학습 방법인 Least Square Estimator(LSE)를 주로 사용하였다. 이로 인해 퍼지 규칙 수가 많거나, 입력이 많은 경우에는 찾고자 하는 파라미터들이 늘어나서 연산시간이 오래 걸리는 단점이 있다. 본 논문에서는 이를 보완하기 위해 Weighted Least Square Estimator(WLSE)를 사용하여 각 규칙에 대한 파라미터를 독립적으로 구하는 지역학습 방법을 적용하였다. LSE는 오차제곱의 합이 최소가 되도록 계수를 측정하지만 WLSE는 오차제곱에 가중치가 곱해진다는 차이가 있다. WLSE에서의 성능 평가함수를 행렬식으로 표현하면 아래 식과 같다.

$$Q_W = \sum_{j=1}^c (Y - X_j a_j)^T U_j (Y - X_j a_j) \quad (20)$$

여기서 a_j 는 추정하고자 하는 j 번째 다항식의 계수, Y 는 출력 데이터, U_j 는 j 번째 입력공간에 대한 입력 데이터들의 소속 값을 의미한다. X_j 는 j 번째 로컬모델의 계수를 추정하기 위한 입력데이터 행렬을 의미하며 로컬모델이 선형일 경우 다음과 같이 정의된다.

$$X_j = \begin{bmatrix} 1 & x_{11} & \dots & x_{1k} \\ 1 & x_{21} & \dots & x_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{m1} & \dots & x_{mk} \end{bmatrix} \quad U_j = \begin{bmatrix} u_{j1} & 0 & \dots & 0 \\ 0 & u_{j2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & u_{jm} \end{bmatrix} \quad (21)$$

여기서 m 은 데이터의 수이며, j 번째 규칙에 대한 로컬모델인 다항식의 계수는 아래 식에 의해 구해진다.

$$a_j = (X_j^T U_j X_j)^{-1} X_j^T U_j Y \quad (22)$$

위와 같이 설계된 다항식 기반 RBFNNs 패턴분류기는 다차원 입, 출력 문제 해결이 용이하며, 네트워크 특성이 강인하고, 예측 능력이 뛰어나다는 장점을 지니고 있다.

2.3 최적화된 RBFNNs 패턴분류기의 설계

각각의 포즈별 데이터는 RBFNNs 패턴분류기를 이용하여 학습 및 테스트 데이터의 인식을 수행한다. 학습 수행 과정에서 속도와 성능을 보다 향상시키기 위해 각 데이터의 최적화된 파라미터가 필요하며, 이를 위해 본 논문에서는 입자군집 최적화 알고리즘을 이용하여 파라미터의 최적화를 수행한다.

2.3.1 입자군집 최적화 알고리즘(PSO)을 이용한 각 모델의 파라미터 탐색

입자군집 최적화 알고리즘[6] (Particle Swarm Optimization : PSO)은 물고기나 새 떼와 같은 무리의 행동 양식을 바탕으로 한 최적화 알고리즘이다. 복잡하고 불확실한 영역에서의 탐색이 가능하고, 간단한 수학 연산자를 사용하여 이론이 간결하고, 다른 확률적 방법보다 안정적인 수렴특징을 가진다는 특징이 있다. 본 논문에서는 이와 같은 최적화 알고리즘을 통해 실험 파라미터의 최적화를 수행하였으며, 최적화 알고리즘의 수행 과정은 아래 단계와 같다.

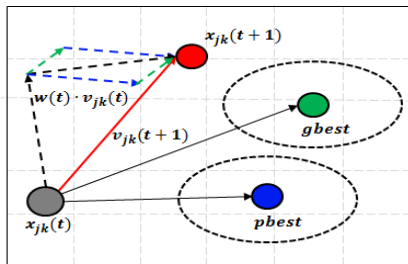


그림 6 입자군집 최적화 알고리즘
Fig. 6 Particle Swarm Optimization(PSO)

- [Step 1] 초기 Swarm과 Particle Velocity를 랜덤하게 생성
- [Step 2] 초기 Particle을 pbest로 선정하며, 그 중 최적의 값을 다시 gbest로 선정
- [Step 3] 아래 식을 통해 관성하중 값을 계산하며, 관성하중 값을 근거로 하여 j번째 Particle Velocity를 계산

$$w(t) = w_{max} - \frac{w_{max} - w_{min}}{it_{max}} \times t \quad (23)$$

$$v_{jk}(t+1) = w(t) \cdot v_{jk}(t) + c_1 \cdot r_1 (pbest_{jk}(t) - x_{jk}(t)) + c_2 \cdot r_2 (gbest_{jk}(t) - x_{jk}(t)) \quad (24)$$

- [Step 4] Particle Velocity를 바탕으로 Particle의 위치정보를 수

정

$$x_{jk}(t) = v_{jk}(t) + x_{jk}(t-1) \quad (25)$$

- [Step 5] 각 Particle의 적합도와 pbest의 적합도를 비교 후 재설정하며, 최적해의 pbest와 gbest를 비교 후 재설정
- [Step 6] 종료조건이 만족할 때까지 탐색을 계속하며, 최종적으로 최적의 위치정보를 가지는 gbest가 생성

위 최적화 알고리즘을 이용하여 퍼지화 계수, 연결가중치의 다항식 형태 및 노드의 수를 최적화 하였으며, 각 포즈별 데이터에 대하여 각각 최적화를 수행하여 각 모델에 최적화된 파라미터 값을 획득한다.

2.3.2 최적화된 RBFNNs 패턴분류기의 설계

본 논문에서는 기존의 정면 얼굴 이미지만을 사용하는 인식 시스템의 단점을 보완하기 위해 다양한 포즈별 얼굴 이미지를 이용하여 포즈별 데이터베이스를 구성하여 인식을 수행한다. 테스트 이미지와 가장 유사한 포즈를 판별하여 인식을 수행하기 때문에 각각의 포즈별 데이터에 대한 학습을 별개로 수행하여야 하며, 최적화 알고리즘을 통해 각 포즈별 데이터에 최적화된 파라미터 값을 획득하여야 한다. 앞 절에서 언급한 다항식 기반 RBFNNs 패턴분류기와 입자군집 최적화 알고리즘을 통해 최적화된 RBFNNs 패턴분류기를 설계하며, 포즈별 데이터의 학습을 수행한다. 최적화된 RBFNNs 패턴분류기를 통해 각 포즈에 최적화된 파라미터 획득 과정은 아래와 같다.

- [Step 1] 다양한 포즈별 이미지를 이용한 학습 데이터베이스의 구성

정면 얼굴 이외의 다른 포즈(좌측 90°, 좌측 45°, 우측 45°, 우측 90°)의 이미지를 이용하여 데이터베이스를 구성하며, 각각의 포즈별 데이터는 인식 대상의 2차원 이미지로서 입력 데이터는 이미지 픽셀 값의 명암도를 이용한다.

- [Step 2] 신뢰성 높은 학습 수행을 위해 각 포즈별 데이터의 분할

포즈별 데이터는 각각 별개로 학습을 수행하며, 학습 수행 시 발생할 수 있는 Overfitting 문제 해결을 위해 데이터는 학습과 검증으로 나누어 학습을 수행한다. 검증 데이터의 경우 실제 RBFNNs 패턴분류기의 파라미터 설정에는 영향을 미치지 않으며, 최적화를 수행하기 위한 목적함수를 구하는 과정에서 사용이 된다. 본 실험에서 사용한 데이터 분할에 관해서는 4장 실험 및 결과고찰 부분에서 다시 자세하게 설명하기로 한다.

- [Step 2-1] 데이터 분할 및 차원 축소

좌측 90°에 해당하는 데이터는 각각 학습 및 검증 데이터로

분할하게 되며, 학습 데이터는 전처리 과정으로서 주성분 분석법을 이용해 차원 축소를 수행한다. 검증 데이터 역시 차원 축소를 수행하지만 학습 데이터에서 획득한 공분산 행렬을 이용하여 차원 축소를 수행한다. 학습 데이터와 검증 데이터를 분할하지 않은 상태에서 차원 축소를 수행할 경우 검증 데이터가 파라미터 산출 과정에서 영향을 주기 때문에 정확한 인식 성능을 확인하기 어렵기 때문이다.

[Step 2-2] 각각에 포즈에 대한 데이터 분할 및 축소 수행

좌측 90° 이외의 다른 포즈(좌측 45°, 정면, 우측 45°, 우측 90°)에 데이터 역시 순차적으로 차원 축소를 수행하며, 수행 방법은 앞서 Step 2-1에서와 같다.

[Step 3] 입자군집 최적화 알고리즘 기반 최적화된 RBFNNs 패턴분류기를 이용한 학습 수행

각 포즈별 데이터는 전처리 과정으로 차원 축소를 수행하였으며, 인식 성능을 확인하기 위해 RBFNNs 패턴분류기를 통해 학습을 수행한다. 퍼지화 계수, 후반부 다항식 차수, 규칙의 수와 같은 학습에 사용되는 파라미터는 입자군집 최적화 알고리즘을 통해 최적화를 수행하였으며, 최종적으로 각 포즈별 데이터에 최적화된 파라미터를 획득한다.

[Step 3-1] 최적화를 통한 탐색 범위 내에서의 파라미터의 최적화 수행

앞서 과정에서 주성분 분석법을 통해 축소된 첫 번째 포즈(좌측 90°)에 데이터를 RBFNNs 패턴분류기의 입력으로 사용하여 인식을 수행한다. 이때 패턴분류기에서 사용하는 파라미터는 PSO를 통해 최적화 한다. 각각의 파라미터는 일정한 탐색 범위를 지정하며, 본 실험에서 사용한 자세한 탐색 범위는 4장 실험 및 결과고찰 부분에서 다시 다루기로 한다.

[Step 3-2] 최적화된 파라미터 값 획득

최적화 과정 수행 중 RBFNNs 패턴분류기를 이용하여 학습 및 검증 데이터의 성능을 확인하며, 인식 성능 및 성능 평가를 위한 목적함수는 아래 식을 통해 산출한다. 정해진 조건만큼 최적화를 수행하며, 최종적으로 획득한 최적화 파라미터는 이후 테스트 이미지의 인식에 사용된다. 목적함수는 Overfitting 문제 해결을 위해 학습과 검증 데이터의 평균을 이용하였다.

$$Classification\ Rate[\%] = 100 - \left(\frac{F}{N} \times 100 \right) \quad (26)$$

(N : 전체 이미지의 수, F : 오인식된 이미지의 수)

$$Objective\ function = \frac{(TR+ VA)}{2} \quad (27)$$

(TR : 학습 데이터의 인식 성능, VA : 검증 데이터의 인식 성능)

[Step 4] 각각에 포즈에 대한 최적화된 파라미터 값 획득

각 포즈별 데이터는 앞서 수행한 최적화 과정을 통해 최적화된 파라미터 값을 획득하며, 포즈별 이미지에 대한 학습을 별개로 수행하였기 때문에 각 포즈에 해당하는 최적화된 파라미터 값을 획득한다.

3. 테스트 이미지 검출 및 인식 기반 객체 추적

본 논문에서는 동적인 상황에서 인식을 수행하기 위해 객체 추적 알고리즘인 Particle Filter와 RBFNNs 패턴분류기를 융합한 얼굴인식 및 추적 시스템을 설계한다. 동적인 상황에서의 인식을 위해서는 비디오 영상 내에서 인식 및 추적을 수행하기 위한 테스트 이미지를 입력 받아야 한다. 3장에서는 AdaBoost 알고리즘을 이용하여 검출된 테스트 이미지의 포즈 분류와 인식 결과를 바탕으로 검출된 얼굴 영역의 추적에 관하여 설명한다.

3.1 AdaBoost 알고리즘 기반 비디오 내의 얼굴 이미지의 검출

2차원 얼굴인식에서는 인식 대상의 이미지를 입력받아 인식을 수행하며, 명암도를 바탕으로 인식을 수행하기 때문에 구성된 학습 데이터와 테스트 이미지간의 오차가 많은 경우 인식 성능이 저하되는 문제가 있다. 따라서 인식을 수행할 때 얼굴 이외의 다른 배경이 들어올 경우 인식에 영향을 미치게 되며, 이 같은 경우를 방지하기 위해 획득한 이미지 중 얼굴영역만을 검출하여 인식을 수행한다. 본 논문에서는 AdaBoost 알고리즘을 통해 얼굴 영역을 검출해내며, AdaBoost 알고리즘은 약한 분류기의 선형적 결합으로 높은 검출 성능을 갖는 강한 분류기를 형성하여 얼굴과 비 얼굴 영역의 분류가 가능하다. 이를 통해 전체 이미지 내에서 실험에 사용하기 위한 얼굴 영역만을 검출하는 것이 가능하다.

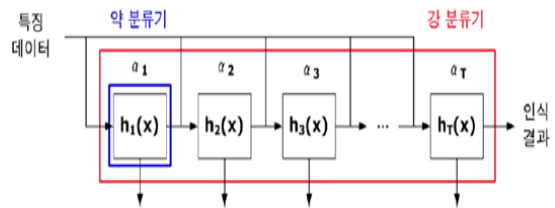


그림 7 AdaBoost 알고리즘

Fig. 7 AdaBoost algorithm

기존의 얼굴인식의 경우 AdaBoost 알고리즘을 이용하여 인식에 사용하기 위한 얼굴 이미지를 추출한다. 이 경우 정면 얼굴 이미지만을 이용하기 때문에 인식 대상자가 카메라를 응시하고 있는 상황에서 이미지를 획득하여야 하며, 이와 같은 얼굴인식 시스템에서는 포즈 변화에 대한 인식 성능의 저하가 발생할 수 있다는 단점이 존재한다. 본 논문에서는 이를 보완하기 위해 다양한 포즈별 이미지를 이용하여 데이터베이스를 구성하며, 테스트

트 이미지 역시 정면 얼굴 이미지 이외의 다른 포즈의 이미지를 입력받아 인식을 수행한다. 비디오 영상에서 AdaBoost 알고리즘을 수행할 경우 얼굴 영역에 해당하는 부분을 검출하는 것이 가능하며, 이와 같이 비디오에서 검출한 테스트 이미지는 아래 그림과 같이 정면 얼굴 이외의 다른 포즈의 이미지가 입력 될 수 있다.

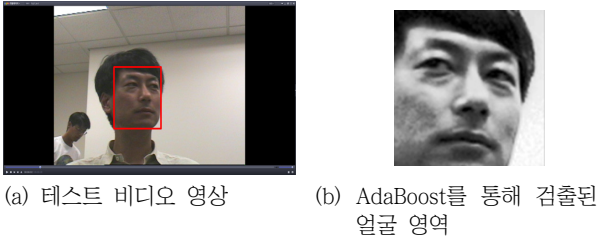


그림 8 테스트 비디오에서 검출된 이미지

Fig. 8 Detected image in test video

AdaBoost 알고리즘을 통해 다양한 포즈의 얼굴 이미지를 검출하는 것이 가능하며, 이렇게 획득한 테스트 이미지는 주성분 분석법 기반 포즈 추정을 통해 가장 유사한 포즈를 판별한 이후 인식을 수행한다. 이때 검출된 이미지와 학습 이미지와의 오차가 큰 경우 제대로 된 포즈 추정 및 인식 성능을 확인하는 것이 어려우며, 아래 그림은 본 논문에서 사용한 학습 데이터와 비디오 내에서 검출한 테스트 이미지를 나타낸다.

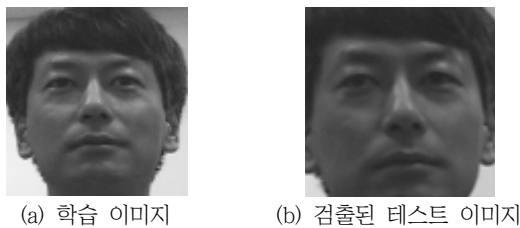


그림 9 학습 이미지와 비디오에서 검출된 테스트 이미지

Fig. 9 Training image and detected test image in video

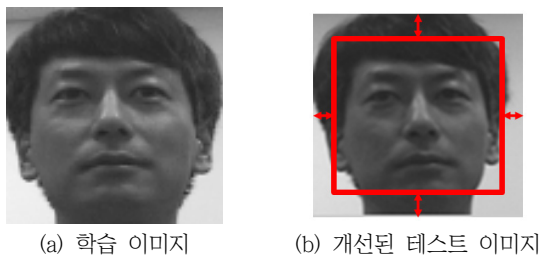


그림 10 검출된 얼굴 영역의 확장

Fig. 10 Extension of detected face area

AdaBoost 알고리즘을 이용할 경우 비디오 영상에서 얼굴 영역에 해당하는 부분만을 검출하기 때문에 위 그림과 같이 학습 이미지와 검출된 영역의 오차가 존재하는 경우가 있다. 이와 같

은 경우 잘못된 포즈 추정 및 인식 성능의 저하가 발생할 수 있으며, 이를 보완하기 위해 AdaBoost 알고리즘을 통해 검출된 얼굴 영역에서 일정한 영역을 추가로 획득함으로써 학습 이미지와 유사한 테스트 이미지를 검출한다.

3.2 포즈 추정을 통한 테스트 이미지와 유사한 포즈 판별

앞서 2장에서 설명한 것과 같이 검출된 테스트 이미지가 정면 이외의 다른 포즈의 얼굴 이미지가 입력될 수 있다. 이러한 다른 포즈의 얼굴 이미지의 인식을 수행하기 위해 각 포즈 변화에 대한 이미지를 이용하여 포즈별 학습 모델을 구성하고 포즈 추정 이후 인식을 수행한다. 포즈 추정을 통해 테스트 이미지와 가장 유사한 포즈의 학습 모델을 찾아내어 인식을 수행함으로써 보다 신뢰성 있는 인식 성능을 얻는 것이 가능하며, 본 논문에서는 특징 추출 알고리즘인 주성분 분석법을 이용하여 포즈 추정을 수행하였다. 주성분 분석법 기반 포즈 추정 과정은 아래 단계와 같다.

- [Step 1] 각 포즈별 모델에 PCA 알고리즘을 수행함으로써 각 학습 모델의 고유 얼굴벡터를 생성
- [Step 2] 각 포즈의 특징을 가장 잘 나타내는 주성분 벡터를 추출
- [Step 3] 획득한 테스트 이미지를 각 포즈별 주성분 벡터에 투영
- [Step 4] 테스트 이미지와의 특징값 오차가 적은 학습 모델을 가장 유사한 포즈라 판별
- [Step 5] 학습을 통해 획득한 포즈별 모델의 최적화된 파라미터 값을 이용하여 테스트 이미지의 인식을 수행

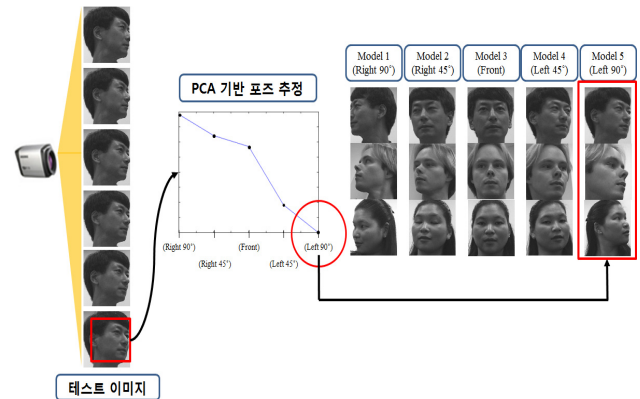


그림 11 포즈 추정을 통한 모델 판별

Fig. 11 Discrimination of model through pose estimation

포즈 추정을 통해 테스트 이미지와 가장 유사한 학습 모델을 판단하게 되는데, 입력된 테스트 이미지에 따라 학습 모델과의 특징값 오차가 큰 경우도 존재한다. 이러한 테스트 이미지의 인식은 포즈 추정을 수행하였다고는 하나 학습 이미지와 테스트 이미지의 오차가 크게 나타나기 때문에 정확한 인식 결과를 확인하기 어렵다. 이를 보완하기 위해 인접한 다른 포즈의 오차를 이용하여 학습 데이터베이스와 유사한 포즈인지를 판별하며, DB와 유사한 포즈만을 이용할 경우 더욱 정확한 인식 성능을 확인하는

것이 가능하다.

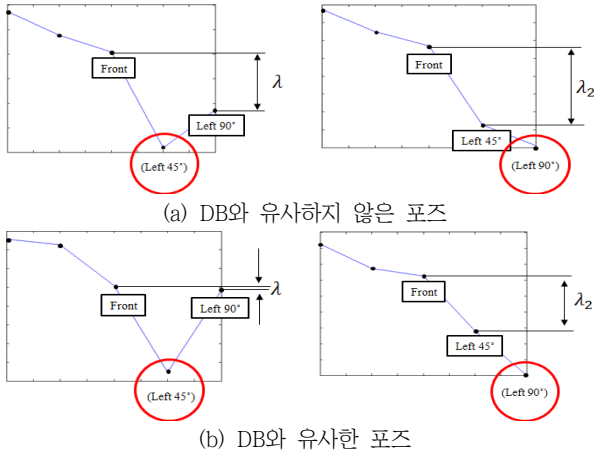


그림 12 인접 포즈와의 거리 값
Fig. 12 Value of distance between adjacent poses

3.3 객체 추적 알고리즘을 통한 얼굴 영역 추적

AdaBoost 알고리즘을 통해 검출된 얼굴 이미지는 앞서 설명한 포즈 분류 및 RBFNNs 패턴분류기를 통하여 인식을 수행한다. 인식 결과를 기반으로 추적 대상이라 판단되는 경우 객체 추적 알고리즘인 Particle Filter를 이용하여 검출 영역의 추적을 수행한다. Particle Filter는 객체 추적에 있어서 많이 사용되고 있는 알고리즘으로, 시물레이션에 기반을 둔 예측 기술의 하나로 SMC(Sequential Monte Carlo)방법이라고도 한다.

Particle Filter는 시스템에 적절하게 제안된 확률분포로 임의로 생성된 입력을 여러 가하여 그것들을 종합하여 시스템의 정보를 추측하는 방법으로 지정된 객체 영역을 추적한다. 기본적으로 예측 단계, 업데이트 단계, 재 샘플링 단계로 구성되며 각 프레임마다 이 과정을 반복하며 파티클을 집합을 갱신한다.

[Step 1] 예측 단계

시간 t-1에서 사용 가능한 모든 관찰(observation)이 주어지고, 시간 t에서 사후 확률을 예측하기 위해 아래 식을 사용

$$p(x_t|y_{t-1}) = \int p(x_t|x_{t-1})p(x_t|y_{1:t-1})dx_{t-1} \quad (28)$$

[Step 2] 업데이트 단계

시간 t에서, 관찰 y_t 가 사용 가능하면, 상태는 아래 식의 베이즈 규칙을 사용하여 업데이트가 가능

$$p(x_t|y_{1:t}) = \frac{p(y_t|x_t)p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} \quad (29)$$

Particle Filter에서 사후 확률은 N개의 유한개 Particle과 각 각의 중요 가중치 w_t^i 에 의해서 근사되며, 후보 Particle \tilde{x}_t^i 는 중

요 분포 $q(x_t|x_{1:t-1}, y_{1:t})$ 에서 추출되며 Particle의 가중치는 다음 식에 의해서 계산

$$w_t^i = w_{t-1}^i \frac{p(y_t|\tilde{x}_t^i)p(\tilde{x}_t^i|x_{t-1}^i)}{q(\tilde{x}_t^i|x_{1:t-1}, y_{1:t})} \quad (30)$$

[Step 3] 재 샘플링 단계

재 샘플링 단계에서는 Particle의 상태가 퇴화하는 것을 막기 위해 가중치가 높은 Particle을 중심으로 새로운 Particle을 다시 추출

기존의 객체 추적 알고리즘은 추적하고자 하는 인물이나 대상에 대한 영역을 미리 설정 하여야 하며, 이 경우 추적에 실패할 경우 다시 영역을 재설정해야 한다는 단점이 존재한다. 본 논문에서는 유클리디안 거리 오차를 이용하여 추적 성공 여부를 확인 하며, 추적 실패 시 AdaBoost 알고리즘을 통해 추적 영역을 재 설정하게 된다. 재설정된 얼굴 영역의 인식 및 추적을 수행함으로써 별도의 영역 설정 없이도 지속적인 대상의 추적이 가능하다는 장점이 있다.

3.4 제안된 인식 및 추적 시스템의 흐름도

본 논문에서는 인식을 수행하기 위해 포즈 추정 및 RBFNNs 패턴분류기를 이용하였으며, 추적을 수행하기 위해 AdaBoost 알고리즘과 객체 추적 알고리즘인 Particle Filter를 사용하였다. 기존의 객체 추적 알고리즘은 추적하고자 하는 영역을 사용자가 직접 지정하여야 하며, 추적에 실패하였을 경우 다시 영역을 재설정해야 한다는 단점이 존재한다. 본 논문에서는 AdaBoost 알고리즘을 통해 인식 및 추적을 수행하기 위한 얼굴 이미지를 검출하였으며, 정확한 인식 성능 확인을 위해 포즈 추정을 통한 유사한 포즈에서의 인식을 수행하였다.

인식 결과를 바탕으로 추적 대상이라 판단이 되면 검출된 얼굴 영역을 Particle Filter[7,8,9,10]를 통해 추적을 수행한다. 추적을 수행하는 도중 추적 성공 여부를 판단함으로써 추적 실패 시 AdaBoost 알고리즘을 수행함으로써 얼굴 영역의 재 검출을 수행한다. 추적 성공 여부는 현재(t)와 이전 시간(t-1)의 검출된 영역의 유클리디안 거리 오차를 기반으로 판단한다. 유클리디안 거리 오차가 지정한 임계값 보다 작은 경우 정상적으로 추적을 수행중이라 판단하며, 임계값 이상의 오차 발생 시 추적에 실패하였다 판단한다. 본 논문에서 제안한 테스트 이미지의 인식 및 추적 시스템의 흐름은 아래 단계와 같다.

[Step 1] 비디오 내에서 인식 및 추적을 수행하기 위한 테스트 이미지 검출

정면 얼굴 이미지가 아닌 비디오 내에서 AdaBoost 알고리즘을 이용하여 얼굴 이미지를 검출한다. AdaBoost 알고리즘을 사용할 경우 비디오 내에서 얼굴 영역과 비 얼굴 영역을 구분하는 것이 가능하며, 이를 이용하여 인식 및 추적을 위한 얼굴 영역을 검출

[Step 1-1] 얼굴 영역 검출에 성공한 경우

얼굴 영역 검출에 성공한 경우 비디오 내에 대상이 존재할 경우를 의미하며, 검출된 얼굴 영역은 포즈 추정 및 인식을 수행

[Step 1-2] 얼굴 영역 검출에 실패한 경우

AdaBoost 알고리즘을 수행하였을 때 얼굴 영역이 검출되지 않은 경우 비디오 내에 대상이 존재하지 않은 경우라 판단하며, 다시 [Step 1]으로 넘어가 비디오 영상 내에서 AdaBoost를 수행

[Step 2] 검출된 얼굴 영역의 포즈 추정 및 인식 수행

검출된 얼굴 이미지는 다양한 포즈의 이미지가 입력되므로 테스트 이미지의 포즈 추정을 통해 가장 유사한 포즈를 판별하여 인식을 수행한다. 주성분 분석법을 통해 테스트 이미지와 각 포즈의 특징값을 비교하며 그 오차가 가장 작은 포즈를 테스트 이미지의 포즈라 판단한다. 기존의 학습 과정에서 획득한 포즈별로 획득한 최적화된 파라미터를 이용하여 인식을 수행

[Step 2-1] 인식 결과가 추적 대상과 동일할 경우

검출된 테스트 이미지의 인식을 수행하였을 때 사용자가 설정한 추적대상과 동일할 경우 검출된 얼굴 영역을 객체 추적을 위한 영역으로 설정하며, 추적을 수행

[Step 2-2] 인식 결과가 추적 대상이 아닌 경우

인식 결과 추적대상이 아니라 판단되면, 비디오 영상 내에서 추적 대상이 존재하지 않으므로 다시 [Step 1]으로 돌아가 AdaBoost를 수행

[Step 3] 인식 결과 추적 대상이라 판단된 경우 Particle Filter를 이용한 추적 수행

검출된 얼굴 영역을 객체 추적을 위한 동적 영역으로 설정하며, 객체 추적 알고리즘인 Particle Filter를 이용하여 추적을 수행한다. 추적에 실패한 경우 다시 추적 대상의 얼굴 영역을 재검출 해야하기 때문에 검출된 영역의 유클리디안 거리 오차값을 기반으로 추적 성공 여부를 판단

[Step 3-1] 추적에 성공한 경우

유클리디안 거리 오차가 사용자가 지정한 임계값보다 작은 경우 정상적으로 추적을 수행하고 있다 판단하며, 검출 영역의 추적을 계속 수행

[Step 3-2] 추적에 실패한 경우

유클리디안 거리 오차가 임계값보다 큰 경우 추적에 실패하였다 판단하며, 이와 같은 경우 다시 [Step 1]으로 돌아가 비디오 영상 내에서 AdaBoost를 수행

[Step 4] 객체 추적 알고리즘을 통해 사용자가 설정한 추적 대상을 지속적으로 추적하며, 전체 비디오 frame 중 추적에 성공한 frame 수를 기반으로 추적 성능을 판단

$$Tracking\ Performance[\%] = \left(\frac{F_n}{F_{total}} \right) \times 100 \quad (31)$$

(Fn : 추적에 성공한 frame의 수,
Ftotal : 테스트 비디오의 전체 frame 수)

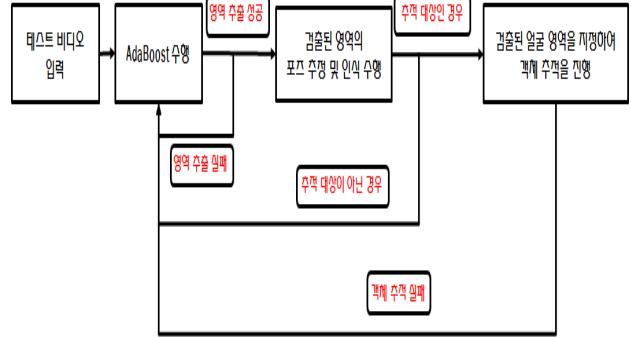


그림 13 제안된 객체 추적의 흐름도

Fig. 13 Flowchart of proposed object tracking system

4. 실험 및 결과고찰

4.1 실험의 전체 개요

본 논문에서는 기존의 정적인 얼굴인식의 단점을 보완하기 위해 동적 상황에서의 인식 및 추적을 수행하며, 이를 위해 객체 추적 알고리즘과 RBFNNs 패턴분류기를 융합한 얼굴인식 시스템을 제안한다. 각 포즈별 학습 데이터 구성 후 RBFNNs 패턴분류기를 통해 각 모델의 학습을 수행하며, 테스트 이미지가 입력되면 주성분 분석법 기반 포즈 추정을 통해 가장 유사한 모델에서의 인식을 수행한다. 이후 인식 결과를 바탕으로 추적을 수행하며, 추적에 실패할 경우 영역의 재검출 및 재인식을 통해 동일한 대상을 지속적으로 추적한다.

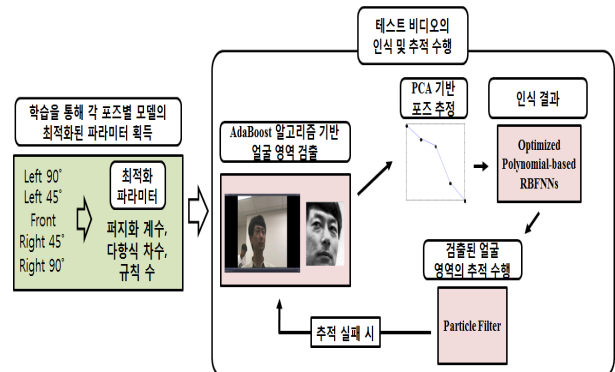


그림 14 제안된 인식 및 추적 시스템

Fig. 14 Proposed recognition and tracking system

실험을 위한 학습 데이터는 공인 DB인 Honda/UCSD[11] 데

이터를 이용하였으며, 인식 대상자 20명의 이미지를 이용하여 구성하였다. 동적인 얼굴인식을 위해 정면, 좌측 45°, 좌측 90°, 우측 45°, 우측 90°의 얼굴 이미지를 이용하여 모델을 구성하였으며, 각 모델에서는 인식 대상자 한명 당 5장의 이미지를 이용하여 100장의 이미지를 이용하여 학습을 수행하였다.

학습과정에서는 데이터의 overfitting 문제 해결을 위해 5-fold cross validation을 이용하여 5장의 이미지를 학습과 검증으로 나누어 학습을 수행하였으며, 각 fcv에서 획득한 성능의 평균을 통해 각 세대의 성능 평가를 수행한다.

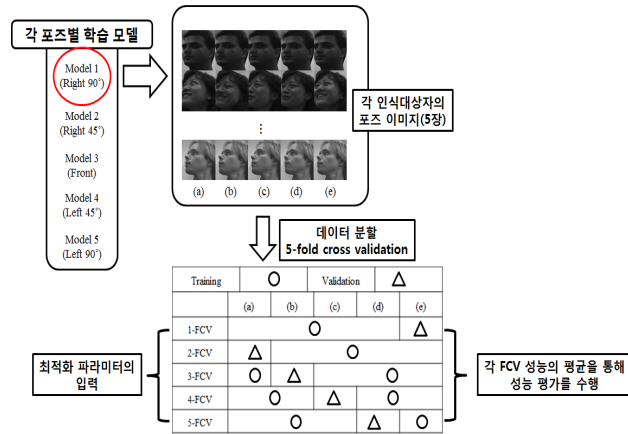


그림 15 5-묶음 교차 검증법을 통한 데이터 분할
Fig. 15 Data partitioning through 5-fold cross validation

5-fold cross validation을 이용하여 분할된 데이터에서 학습을 수행하여 최적화된 파라미터를 획득한다. 이후 최적화된 파라미터를 전체 데이터에 입력하여 테스트 데이터 인식을 위한 학습 모델을 획득하며, 각 포즈별 모델에서 이와 같은 과정을 수행함으로써 각 포즈 모델에 해당하는 학습 모델을 획득할 수 있다.

또한 학습과정에서 수행하는 최적화 파라미터의 탐색 범위 및 최적화 알고리즘의 파라미터는 아래 표와 같이 설정하였다.

표 1 최적화 파라미터의 설정 및 파라미터의 탐색 범위
Table 1 Setting of initial optimization parameters and search range of parameters

	Parameters	Values
PSO	No. of generations	20
	Swarm Size	40
	$[w_{min}, w_{max}]$	[0.4, 0.9]
	c_1, c_2	2.0
Search Range	Fuzzification coefficient	[2 ~ 3.0]
	Polynomial Type	2 : Linear 3 : Quadratic 4 : Modified Quadratic
	No. of nodes	[5 ~ 10]

테스트 이미지는 각 인식대상자의 비디오 영상에서 검출한 이미지 230장으로 구성하였으며, 실험을 통해 추적 및 인식 성능을 확인하였다. 추적 성능은 230장의 이미지 중에서 AdaBoost를 통해 검출된 얼굴 영역을 추적한 frame의 수를 기반으로 판단하였으며, 추적을 수행한 frame에서 검출된 얼굴 영역의 인식을 수행하여, 인식 성능을 확인하였다. 추적 및 인식 성능은 포즈 추정 방법에 변화를 주어 확인하였으며, 그에 따른 실험 결과는 4.3절에서 다루기로 한다.

4.2 제안된 인식 및 추적 시뮬레이터

본 논문에서는 다항식 기반 RBFNNs 패턴분류기와 Particle Filter를 융합한 인식 및 추적 시스템을 제안하였다. 성능 확인을 위해 공인 DB인 Honda/UCSD 데이터를 이용하였으며, 각각의 인식 대상은 실험을 수행하기 위한 테스트 비디오가 존재하며, 본 논문에서는 비디오 영상에서 검출한 230장의 이미지를 이용하여 인식 및 추적을 수행하였다. 시뮬레이션을 통해 추적 성능 및 인식 성능을 확인하며, 시뮬레이션의 구성은 다음과 같다.

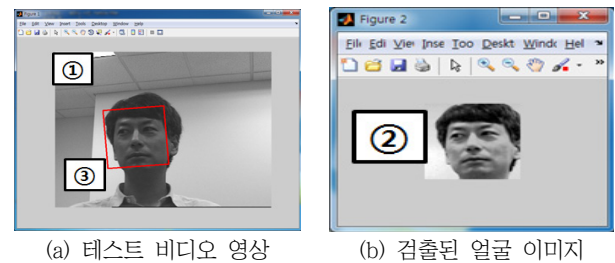


그림 16 인식 및 추적 시뮬레이터
Fig. 16 Recognition and Tracking simulator

- ① : 인식 및 추적을 수행하기 위한 테스트 비디오 영상 (Honda/UCSD data)
- ② : AdaBoost 알고리즘을 통해 검출된 얼굴 이미지
- ③ : 인식 결과를 통해 검출된 얼굴 이미지가 추적 대상이라 판단될 경우 추적을 위한 영역 설정

테스트 비디오에서 AdaBoost 알고리즘을 통해 얼굴 영역을 검출하며, 검출된 얼굴 이미지가 추적 대상이라 판단될 경우 위

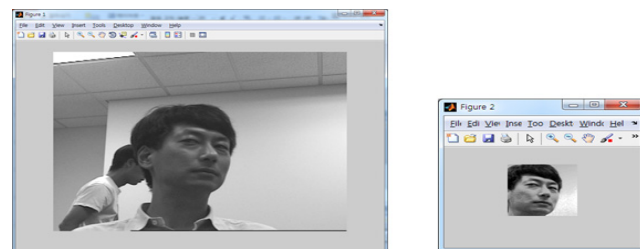


그림 17 추적 영역 설정에 실패한 경우
Fig. 17 In case of failure to setting the tracking area

그림과 같이 검출된 영역(③)의 추적을 수행한다. 만일 검출된 얼굴 이미지의 인식을 수행하였을 때 인식 결과가 추적 대상이 아니라 판단된 경우 얼굴 영역을 검출하지 않고 다시 AdaBoost를 수행한다.

4.3 실험결과

본 논문에서는 기존의 정적인 상황에서의 문제점을 해결하기 위해 객체 추적 알고리즘과 RBFNNs 패턴분류기를 융합한 인식 및 추적 시스템을 제안하였다. 포즈별 학습 모델 구성 후 테스트 이미지가 입력되면 포즈 추정을 통해 가장 유사한 모델에서의 인식 및 추적을 수행하며, 검출된 테스트 이미지를 전부 사용하는 방법(Case 1)과 인접한 포즈의 오차를 이용하여 검출된 테스트 이미지 중 DB와 유사한 포즈의 테스트 이미지만을 사용하는 경우(Case 2)에서의 실험을 수행하였다.

표 2 각 실험 사례

Table 2 Each experiment case

포즈 추정 방법	
Case 1	검출된 테스트 이미지를 전부 사용
Case 2	검출된 테스트 이미지 중 DB와 유사한 포즈의 이미지만을 사용

각 포즈별 모델은 각각 다항식 기반 RBFNNs 패턴분류기를 통해 최적화된 파라미터를 획득하며, 각 모델에 최적화된 파라미터 값은 아래 표와 같다.

표 3 최적화된 파라미터 값

Table 3 Optimized parameter value

	좌측	좌측	정면	우측	우측
	90°	45°		45°	90°
퍼지화 계수	2.29	2.49	2.33	2.12	2.04
다항식 차수	2	2	2	2	2
	[Linear]	[Linear]	[Linear]	[Linear]	[Linear]
규칙 수	9.67[10]	9.54[10]	9.52[10]	9.71[10]	10[10]

추적 성능의 경우 테스트 데이터로 주어진 230장의 테스트 데이터 중 AdaBoost 알고리즘을 통해 검출된 얼굴영역의 추적을 수행하여 전체 테스트 데이터 중 추적에 성공한 이미지 수를 바탕으로 성능을 산출한다. 인식 성능은 전체 230장 중 추적에 성공한 이미지에서만 인식을 수행하며, 각 실험대상별로 추적에 성공한 이미지가 다르기 때문에 실험 대상마다 인식에 사용하는 이미지의 수가 다르다.

$$\text{추적 성능}(\%) = \frac{\text{추적에 성공한 이미지}(\eta_1)}{\text{전체 테스트 이미지}(230)}, \quad (24)$$

$$\text{인식 성능}(\%) = \frac{\text{인식에 성공한 이미지}(\eta_2)}{\text{추적에 성공한 이미지}(\eta_1)}$$

Case 1의 경우 검출된 테스트 이미지의 포즈 추정 이후 인식을 수행하였으며, 추적 대상인 경우에만 추적을 수행하도록 하였다. 이에 대한 추적 성능 및 인식 성능은 위 표 4와 같으며, 실험 대상 중 특정 대상의 경우 오인식이 발생하여 평균 추적 성능보다 낮은 성능을 보이는 것을 확인할 수 있었다.

인식 성능은 추적을 수행하여 검출된 얼굴 이미지만을 이용하기 때문에 전체 230장 중 추적에 성공한 일부 이미지에서 인식을 수행하였다. 각 실험 대상에 따라 추적 성능이 다르듯 인식 성능도 다르게 나오는 것을 확인하였으며, 특정 대상의 경우 평균 인식 성능보다 낮은 성능을 보이는 것을 확인할 수 있었다. 본 논문에서는 포즈 추정 방법에 따라 실험을 수행하였으며, 아래 표 5에서는 DB와 유사한 포즈의 이미지만을 사용하는 경우(Case 2)에서의 추적 및 인식 성능을 나타내었다.

표 4 Case 1의 추적 및 인식 성능

Table 4 Tracking and recognition result of Case 1

Case 1(검출된 테스트 이미지를 전부 사용)					
실험 대상	1	2	3	4	5
A	189/230	139/230	165/230	144/230	171/230
B	82.17%	60.43%	71.74%	62.61%	74.35%
C	139/189	139/139	123/165	126/144	145/171
D	73.55%	100%	74.55%	87.50%	84.80%
실험 대상	6	7	8	9	10
A	144/230	175/230	162/230	103/230	167/230
B	62.61%	76.09%	70.43%	44.78%	71.30%
C	86/144	158/175	156/162	103/103	149/167
D	59.72%	90.29%	96.30%	100%	89.22%
실험 대상	11	12	13	14	15
A	166/230	98/230	229/230	165/230	175/230
B	72.17%	42.61%	99.57%	71.74%	76.09%
C	131/166	98/98	227/229	150/165	109/175
D	78.92%	100%	99.13%	90.91%	62.29%
실험 대상	16	17	18	19	20
A	182/230	152/230	190/230	108/230	176/230
B	79.13%	66.01%	82.61%	47.00%	76.52%
C	165/182	145/152	189/190	93/108	162/176
D	90.66%	95.39%	99.47%	86.11%	92.05%

A : 추적에 성공한 이미지/전체 이미지 B: 추적 성능
 C : 인식에 성공한 이미지/전체 이미지 D: 인식 성능
 평균 추적 성능 : 69.50 %, 평균 인식 성능 : 87.54 %

Case 2에서는 검출된 테스트 이미지의 포즈 추정 이후 DB와 유사한 포즈만을 재 검출하여 사용하였으며, 이에 따른 성능은 위 표 5와 같다. DB와 유사한 포즈만을 사용하기 때문에 기존의 Case 1보다 낮은 추적 성능을 보이는 반면 인식 성능에서는 Case 1보다 높은 성능을 보이는 것을 확인할 수 있었다. 이는 Case 2에서 보다 정밀한 포즈 추정을 통해 DB와 유사한 얼굴 이미지만을 검출하여 인식을 수행하였기 때문이다.

표 5 Case 2의 추적 및 인식 성능

Table 5 Tracking and recognition result of Case 2

Case 2(검출된 테스트 이미지를 전부 사용)					
실험 대상	1	2	3	4	5
A	189/230	128/230	148/230	144/230	164/230
B	82.17%	55.65%	64.35%	62.61%	71.30%
C	148/189	128/128	121/148	133/144	141/164
D	78.31%	100%	81.76%	92.36%	85.98%
실험 대상	6	7	8	9	10
A	110/230	175/230	162/230	101/230	167/230
B	47.83%	76.09%	70.43%	43.91%	72.61%
C	74/110	161/175	156/162	101/101	149/167
D	67.27%	92%	96.30%	100%	89.22%
실험 대상	11	12	13	14	15
A	166/230	83/230	229/230	164/230	169/230
B	72.17%	36.09%	99.57%	71.30%	73.48%
C	133/166	83/83	227/229	149/164	128/169
D	80.12%	100%	99.13%	90.85%	75.74%
실험 대상	16	17	18	19	20
A	151/230	152/230	185/230	108/230	175/230
B	65.65%	66.09%	80.43%	46.96%	76.09%
C	136/151	145/152	184/185	97/108	162/175
D	90.07%	95.39%	99.46%	89.81%	92.57%
A :추적에 성공한 이미지/전체 이미지 B: 추적 성능					
C :인식에 성공한 이미지/전체 이미지 D: 인식 성능					
평균 추적 성능 : 69.50 %, 평균 인식 성능 : 87.54 %					

5. 결 론

본 논문에서는 정적인 상황에서 얼굴 이미지를 추출하여 인식을 수행하는 기존의 얼굴인식 방법을 개선한 동적 상황에서의 인식 및 추적 시스템을 제안하였다. 테스트 이미지의 인식 성능 향상을 위해 학습 데이터 구성 시 각 포즈 변화에 대한 얼굴 이미지를 이용하였으며, 최적화된 RBFNNs 패턴분류기를 통해 각 모델에 최적화된 파라미터 값을 획득하였다. 테스트 이미지는 각 인식대상의 비디오 영상에서 검출한 230장의 이미지를 이용하였으며, 주성분 분석법을 기반으로 각 이미지의 포즈 추정 수행 후 인식 및 추적을 수행하였으며, 이에 대한 성능은 위에 언급한 결과와 같다.

각 사례별 실험을 통해 추적 성능은 유사하게 나오나, 인식 성능의 경우 학습 모델과 유사한 포즈의 테스트 이미지만을 사용한 경우가 더 높은 성능을 보이는 것을 확인하였으며, 추적 도중 영역 추적이 실패하여도 얼굴 영역 재 검출 및 재인식을 통해 동일한 대상을 지속적으로 추적하는 것을 확인하였다.

본 논문에서는 성능 확인을 위해 공인 DB인 Honda/UCSD 데이터를 이용하여 결과를 확인하였으며, 학습 포즈의 확장 및 포즈 추정 과정에서의 최적화를 통해 성능이 개선될 것이라 예상되며, 보다 다양한 사례에 대한 실험을 수행함으로써 제안된 인식 및 추적 시스템의 성능을 확인할 것이다.

감사의 글

본 연구는 경기도의 경기도지역협력연구센터사업의 일환으로 수행하였음[GRRC 수원2014-B2, U-city 보안감시 기술협력센터] 그리고 한국 산업단지공단의 10차년도 산업집적지경쟁력강화산업계획의 생산기술사업화 지원사업으로 연구를 수행하였음(NTIS-1415136442)

References

- [1] H. Addi and L. J. Williams, "Principal component analysis," Wiley Interdisciplinary Reviews: Computational Ststicstics, vol. 2, no. 4, pp. 433-459, 2010.
- [2] Byoung-Jun Park, Wook-Dong Kim, Sung-Kwun Oh, and Witold Pedrycz, "Fuzzy Set-oriented Neural Networks Based on Fuzzy Polynomial Inference and Dynamic Genetic Optimization", Knowledge and Information Systems(KAIS), Vol. 39, Issue 1, pp.207-240, Apr. 2014.
- [3] S. B. Roh, S. C. Joo, W. Pedrycz, and S. K. Oh, "The development of fuzzy radial basis function neural networks based on the concept of information ambiguity," Neurocomputing, Vol. 73, No.13-15, pp. 2464-2477. 2010.
- [4] S. K. Oh, W. Pedrycz, B. J. Park, "Polynomial-based Radial Basis Function Neural Net works realized with the Aid of Particle Swarm Optimization," Fuzzy Sets and Systems, Vol. 163, pp. 54-77, 2011.
- [5] S-K. Oh, W-D. Kim, and W. Pedrycz, "Polynomial based radial basis function neural networks (P-RBFNNs) realized with the aid of particle swarm optimization," Fuzzy Sets and Systems, Vol. 163, No. 1, pp. 54-77, 2011
- [6] J. Kennedy and R. Everhart, "Particle Swarm Optimization," Proc. of IEEE International Conference on Neural Networks," Vol. 4, pp. 1942-1948, 1995.
- [7] H. K. Ekenel, J. Stallkamp, and R. Stiefelhagen, "A video-based door monitoring system using local appearance-based face models," Computer Vision and Image Understanding, Vol. 114, Issue 5, pp. 596-608, 2010.
- [8] K. Nummiaro, E. Koller-Meier and L.van Gool, "A color-based particle filter," First International Workshop on Generative Model Based Vision, pp. 53-60, June 2002.

[9] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," IJCV, 2007.

[10] A. Levey, and M. Lindenbaum, "Sequential Karhunen-Loeve basis extraction and its application to images," Image Processing, IEEE Transactions on, Vol. 9, Issue 8, 1371 - 1374, Aug 2000.

[11] Honda UCSD Video Database, Available: <http://vision.ucsd.edu/content/honda-ucsd-video-database>, 2005, [Accessed: Oct 22, 2013]



김진율(Jin-Yul Kim)

1986년 서울대학교 전자공학과 졸업, 한국과학기술원 전기 및 전자공학과 공학 석사(1988), 공학박사(1993), 1994~현재 수원대학교 전자공학과 부교수
 관심분야: 신호처리시스템설계 영상 내 객체 추적.

Tel : 031-220-2323

Fax : 031-220-2494

E-mail : jykim@suwon.ac.kr

저 자 소 개



오승훈(Seung-Hun Oh)

2013년 수원대학교 전기공학과 졸업, 동 대학원 석사(2015), 2015년~현재 동 대학원 박사과정 관심분야는 퍼지추론 시스템, 뉴럴 네트워크, Granular Computing, 퍼지 제어기, 패턴분류 및 퍼지 모델링

Tel : 031-222-6544

Fax : 031-220-2667

E-mail : osh@suwon.ac.kr



오성권(Sung-Kwun Oh)

1981년 연세대학교 전기공학과 졸업, 동 대학원 석사(1983), 박사(1993). 1983-1989년 금성산전연구소(선임연구원). 1996-1997년 캐나다 Manitoba 대학 전기 및 컴퓨터공학과 Post-Doc. 1993-2004년 원광대학교 전기전자 및 정보공학부 교수. 2005~현재 수

원대학교 전기공학과 교수, 2002~현재 대한전기학회, 제어로봇시스템학회, 퍼지및지능시스템학회 편집위원. 관심분야는 퍼지 시스템, 퍼지-뉴럴 네트워크, 자동화 시스템, 고급 computational intelligence, 지능 제어등.

Tel : 031-229-8162

Fax : 031-220-2667

E-mail : ohsk@suwon.ac.kr