

SVM과 LDA를 이용한 마커 검출 및 인식의 성능 향상

강선경[†], 소인미[‡], 김영운^{***}, 이상설^{****}, 정성태^{*****}

요 약

본 논문에서는 SVM(Support Vector Machine)과 LDA(Linear Discriminant Analysis)를 이용하여 사각형 형태 마커 검출 및 인식의 성능을 향상시키는 방법을 제안한다. 본 논문의 방법에서는 사각형 형태의 마커 검출을 위하여 입력 영상을 이진 영상으로 변환하고 객체들의 윤곽선을 추출한 다음에 윤곽선을 선분으로 근사화 한다. 근사화된 선분으로부터 기하학적 특징을 이용하여 사각형을 찾는다. 마커의 사각형 영역을 찾은 다음에는 워핑 기법과 확대/축소 변환을 이용하여 사각형 영상을 정사각형 형태로 정규화한다. 정사각형 형태로 정규화한 다음에는 주성분 분석을 적용하여 특징 벡터의 크기를 줄인 다음에 SVM을 이용하여 마커 영상인지 아닌지를 검사한다. 마커 영상으로 판별된 영상에 대하여 LDA를 적용하여 특징 벡터의 크기를 더 줄이고 표준 마커에 대한 특징 벡터와의 최소 거리법에 의해 마커의 종류를 인식한다. 인식 실험 결과 SVM을 사용함으로써 마커 검출의 오류를 줄일 수 있었고 LDA를 사용함으로써 특징 벡터의 크기는 줄어들고 인식률이 높아짐을 알 수 있었다.

Performance Enhancement of Marker Detection and Recognition using SVM and LDA

Sun-Kyoung Kang[†], In-Mi So[‡], Young-Un Kim^{***}, Sang-Seol Lee^{****},
Sung-Tae Jung^{*****}

ABSTRACT

In this paper, we present a method for performance enhancement of the marker detection system by using SVM(Support Vector Machine) and LDA(Linear Discriminant Analysis). It converts the input image to a binary image and extracts contours of objects in the binary image. After that, it approximates the contours to a list of line segments. It finds quadrangle by using geometrical features which are extracted from the approximated line segments. It normalizes the shape of extracted quadrangle into exact squares by using the warping technique and scale transformation. It extracts feature vectors from the square image by using principal component analysis. It then checks if the square image is a marker image or a non-marker image by using a SVM classifier. After that, it computes feature vectors by using LDA for the extracted marker images. And it calculates the distance between feature vector of input marker image and those of standard markers. Finally, it recognizes the marker by using minimum distance method. Experimental results show that the proposed method achieves enhancement of recognition rate with smaller feature vectors by using LDA and it can decrease false detection errors by using SVM.

Key words: SVM(Support Vector Machine), LDA(선형 판별 분석), Marker Detection(마커 검출),
Marker Recognition(마커 인식), Principal Component Analysis(주성분 분석)

1. 서 론

의미가 부여된 마커를 카메라 영상으로부터 검출하고 인식하여 증감 현실(augmented reality), 로봇의 경로 이동, 객체 인식 등 다양한 분야에 활용하는 연구가 수행되고 있다. ARToolkit[1], ARTag[2], SCR[3], HOM[4]등의 마커 인식 시스템들은 증감 현실에서 자신의 위치를 파악하거나, 사용자의 시선 방향을 인식하거나, 카메라에 보이는 영상에 추가하여 3차원 객체를 디스플레이하거나 텍스트 정보를 출력하는데 사용되었다[5-8]. 참고 문헌 [3]에서는 원격 교육을 위한 촬영에서 마커를 추적함으로써 카메라를 자동으로 조정하는 용도로 활용하였다. 참고 문헌 [4]에서는 로봇이 마커를 인식함으로써 자신의 위치를 파악하는데 활용하였다. 이들 마커 시스템에서는 평면에 그려진 사각형 또는 원형 둘레를 가진 마커를 사용하였는데, 참고 문헌 [9]의 시스템에서는 사람 눈에는 보이지 않고 적외선 카메라로만 감지할 수 있는 마커를 사용함으로써 보다 깔끔한 주변 환경을 제공하는 방법을 제안하였다. 참고 문헌 [10]에서는 발광 다이오드를 사용해서 마커의 내용을 변화시킬 수 있게 함으로써 보다 동적인 시스템을 구축하는 방법을 제안하였다.

ARTag, SCR, HOM의 마커는 2차원 바코드처럼 격자 형태의 구조에 데이터를 코드화한 형태를 취하고 있으며 인식의 견고성을 더 높일 수 있고 많은 수의 마커를 쉽게 생성할 수 있는 장점을 가진다. 그러나 이들 마커는 사람이 봄서는 어떤 의미를 가진 마커인지 인지하기는 거의 불가능하다. ARToolkit, 참고문헌 [5,11] 등에서는 알파벳, 숫자, 기호 등 특정한 구조를 가지지 않는 마커를 사용하였지만 이들 마커에서도 마커 자체에 특별한 의미를 부여하지는 않았다. 다. 참고문헌 [12]에서는 그림 1과 같은 사람이 인지 가능한 마커를 인식하는 방법을 제안하였다.

참고 문헌 [12]에서 제안한 방법은 배경이 단순한

※ 교신저자(Corresponding Author) : 정성태, 주소 : 전북 익산시 신용동 원광대학교 공과대학 컴퓨터공학과(570-749), 전화 : 063)850-6886, FAX : 063)856-8009,

E-mail : stjung@wku.ac.kr

접수일 : 2006년 1월 4일, 완료일 : 2007년 5월 28일

* 정희원, 원광대학교 컴퓨터공학과

(E-mail : selfid@nate.com)

** 준희원, 원광대학교 컴퓨터공학과

(E-mail : s0301@wku.ac.kr)

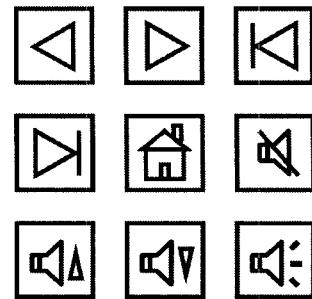


그림 1. 사람이 인지 가능한 마커 예

영역에 마커가 부착되어 있어야만 올바르게 동작하는 문제점을 가지고 있다. 즉, 배경이 복잡한 자연 영상인 경우에는 마커가 아닌 사각형 영역도 마커로 검출하는 문제점을 가지고 있다. 따라서 본 논문에서는 입력 영상으로부터 검출된 사각형 영역이 마커인지 아닌지를 판별하는 과정을 추가하여 배경이 자연 영상인 경우에도 마커 검출이 가능하도록 하는 방법을 제안한다. 그리고 참고 문헌 [12]에서는 특징 벡터의 추출에 주성분 분석을 사용하였는데, 주성분 분석과 LDA를 혼합하여 사용하여 인식률 성능을 향상시키는 방법을 제안한다.

본 논문에서는 입력 영상을 이진 영상으로 변환하고 객체들의 유판선을 추출한 다음에 유판선을 선분으로 근사화하고 근사화된 선분으로부터 기하학적 특징을 이용하여 사각형을 찾는다. 사각형 영역을 찾은 다음에는 사각형 영역의 영상에 대하여 워핑 기법을 적용하여 정사각형 형태로 정규화한다. 정규화된 영상에 대하여 주성분 분석을 수행하여 특징 벡터를 추출하고 이 특징 벡터에 대하여 SVM 분류기를 적용하여 사각형 영역이 마커 영역인지 아닌지를 판별한다. 마커로 판별된 영역에 대하여 LDA를 수행함으로써 특징 벡터의 크기를 줄이고 표준 마커에 대한 특징 벡터와의 최소거리법에 의해 마커의 종류를 인식한다. 실험 결과 본 논문의 방법은 참고 문헌 [12]의 방법에 비하여 특징벡터의 크기가 작으면서도 인

*** 준희원, 원광대학교 컴퓨터공학과
(E-mail : kyw1007@wku.ac.kr)

**** 정희원, 원광대학교 전기전자 및 정보공학부
(E-mail : s3ee@wku.ac.kr)

***** 종신희원, 원광대학교 전기전자 및 정보공학부
(E-mail : stjung@wku.ac.kr)

※ 본 연구는 2007년 정부(교육인적자원부)의 지원으로 한국학술진흥재단의 지원을 받아 수행된 연구임(지방연구중심대학육성사업/헬스케어기술개발사업단)

식률이 향상되는 것을 알 수 있었다.

본 논문의 II장에서는 마커 검출 방법에 대해서 설명하고 III장에서는 마커 인식 방법을 설명하며 IV장에서는 실험 결과를 설명하고 V장에서 결론을 맺는다.

2. 마커 검출

그림 2에는 자연 영상 배경으로 촬영된 마커의 예가 나타나 있다. 이러한 영상으로부터 마커를 검출하기 위해서는 먼저 사각형 영역을 검출하고 그 영역의 영상이 마커인지 아닌지를 검사해야 한다.

본 논문의 마커 검출 과정이 그림 3에 나타나 있다. 본 논문에서는 사각형 마커를 검출하기 위해 먼저 입력 영상을 이진 영상으로 변환하고 윤곽선을 검출한다. 여기에서 검출된 윤곽선은 연결된 픽셀들로 구성되어 있는데, 사각형 검출을 위해 윤곽선을 선분으로 균사화한다. 선분으로 균사화한 다음에는 윤곽선을 이루는 선분의 수, 윤곽선 내부의 면적, 볼록 객체 여부 등의 기하학적 특징을 검출하여 이로부터 윤곽선이 사각형인지 아닌지를 판단한다.

마커 검출의 첫 단계에서는 입력 영상을 이진 영상으로 변환한다. 사각형 마커는 흰색 배경에 검정색으로 되어있지만 카메라로부터 촬영된 입력 영상에서는 주변 환경의 빛의 밝기에 따라 배경과 마커의 명암도가 많은 영향을 받으므로, 이진화를 위한 적절한 임계값을 정하는 문제는 그리 간단한 문제가 아니다. 본 논문에서는 흑백 영상이 가질 수 있는 명암도 값인 0에서 255 사이에 일정 간격으로 9개의 임계값을 지정하여 이진 영상 9개를 생성하고 각 이진 영상

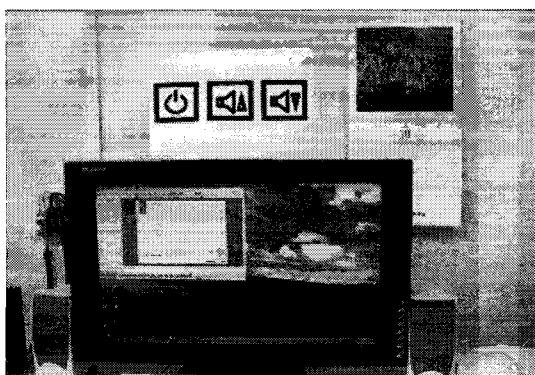


그림 2. 마커 촬영 예

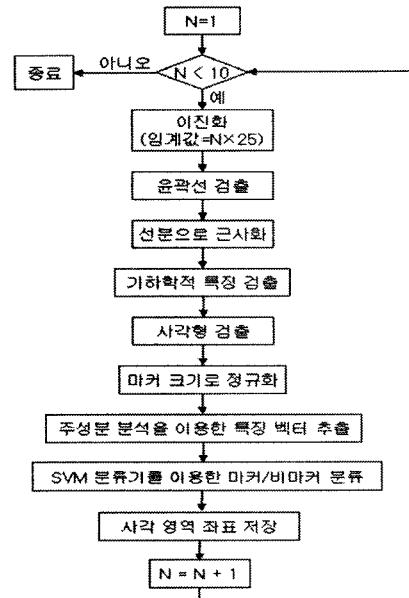


그림 3. 마커 검출 단계

으로부터 사각형을 검출하는 방법을 사용한다. 그림 4에는 그림 2에 대한 이진화 영상 예가 나타나 있다.

이진화를 수행한 다음에는 윤곽선 검출을 수행한다. 본 논문에서는 객체의 가장자리에 있는 한 픽셀로부터 시작하여 가장자리에 있는 픽셀을 계속 따라가다가 시작점에 도착하면 추적을 멈추는 방법[13]을 사용하였다. 그 다음에는 윤곽선을 선분으로 균사화한다. 윤곽선을 구성하는 픽셀들을 선분으로 균사화하는 방법으로 본 논문에서는 Douglas-Peucker 알고리즘[14]이라 불리는 방법을 사용한다. 선분으로 균사화한 다음에는 사각형인지 아닌지를 판단하기 위하여 기하학적 특징을 이용한다. 기하학적 특징으로는 윤곽선 내부의 면적, 각 선분 사이의 각도, 볼록 객체 여부, 윤곽선을 구성하는 선분의 수, 선분의 길이를 이용한다. 기하학적 특징을 계산하는 방법

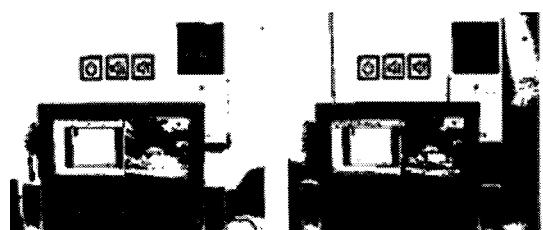


그림 4. 이진화 영상 예

은 참고문헌 [12]의 방법을 이용하였다. 본 논문에서 선분으로 균사화된 윤곽선이 사각형 마커의 윤곽선인지를 판별하는 조건으로 윤곽선을 구성하는 선분의 수가 4개이고 윤곽선 내부 면적이 100이상이고 이웃한 선분 사이각이 45도에서 135도 사이이며 볼록 다각형이어야 하도록 했다. 윤곽선 내부 면적을 검사하는 것은 마커를 인식하기 위해 최소한 마커 영상의 크기가 가로 세로 각각 10 픽셀씩은 되어야 할 것으로 판단했기 때문이다. 그림 5에는 그림 2의 영상에 대한 사각형 검출 결과가 나타나 있다. 모두 5개의 사각형 영역이 검출되었는데 3개는 마커 영역이고 2개는 마커가 아닌 일반 영역에서 사각형이 검출되었다.

검출된 사각형 영역의 영상이 마커인지 아닌지를 판별하기 위해서는 먼저 사각형 영역의 영상을 표준 마커 영상과 동일한 크기와 모양이 되도록 변환한다. 본 논문에서는 크기가 64×64 픽셀인 정사각형 형태의 마커를 사용한다. 일반 사각형 형태를 정사각형 형태로 변환하기 위해 본 논문에서는 위평 기법[15]과 확대/축소 변환을 사용한다. 먼저 위평 연산으로 직사각형 형태로 변환한 다음에 축소/확대 변환으로 크기가 64×64 인 정사각형 형태로 변환하였다. 그림 2의 영상에서 검출된 사각형 영역들에 대해 위평과 확대/축소 변환을 적용한 결과가 그림 6에 나타나 있다.



그림 5. 사각형 검출 결과



(a) 위평 결과



(b) 확대 축소 결과

그림 6. 추출된 사각형에 대한 정규화

정사각형 형태로 변환한 다음에는 이진화를 수행하는데, 이진화에 사용하는 임계값은 사각형 검출시에 적용되었던 임계값과 같은 값을 사용하면 된다. 그런데, 여러 가지 임계값에 대하여 이진화를 수행하고 마커를 검출하였으므로 같은 마커가 여러 번 검출될 수 있다. 같은 마커들이 여러 번 검출된 경우에는 그중에서 임계값이 중간인 경우에 검출된 마커를 선택하여 사용한다. 그림 3의 영상에서는 임계값이 50, 75, 100, 125인 경우에 마커가 검출되었는데, 이중에서 임계값이 75를 사용하여 이진화를 수행한다. 이진화 결과는 그림 7에 나타나 있다.

정규화와 이진화를 수행한 다음에는 영상을 주성분 분석 방법에 의하여 특징 벡터를 구한다. 주성분 분석이란 많은 축으로 구성된 데이터 공간에서 데이터가 어떤 형태로 나타나는지 분석하여 주성분 축을 구하고 데이터들을 주성분 축에 투영하여 데이터 공간의 차원을 축소하는 방법이다. 본 논문에서는 마커 영상과 비마커 영상 각각 700개씩 모두 1400개의 영상을 사용하여 주성분 분석을 수행하여 주성분 축을 구하였다. 그리고 정규화되고 이진화된 영상을 주성분 축에 투영함으로써 특징 벡터를 구하였다. 특징 벡터의 크기는 100차원을 사용하였다.

본 논문에서는 특징 벡터를 구한 다음에는 SVM 분류기를 이용하여 마커 영상인지 아닌지를 분류하였다. SVM은 이진패턴 분류 문제를 해결하기 위해 제안된 학습방법으로 각 범주의 대표적인 특성을 갖는 벡터를 이용하여 최적의 하이퍼면리를 찾아서 두 범주를 분류하는 기법이다[16,17]. SVM은 인식 성능과 처리 속도 성능이 매우 뛰어나 최근에 다양한 분야에 적용되고 있다. 분류기들을 훈련하기 위해서 사용되는 기법들 중 대부분이 훈련 예상적인 경험적 위험을 최소화한다는 아이디어에 기초하는 반면에 SVM은 일반화 예상의 상한을 최소화하는 구조적 위험 최소화이라 불리는 다른 추론원리에 의해서 동작한다. 유클리드 공간에서 두 개의 범주를 갖는 N개의 객체가 존재할 때, 그림 8에서 점선으로 표시된 바와 같이 두 범주를 분류하는 하이퍼면은 무수



그림 7. 이진화된 영상

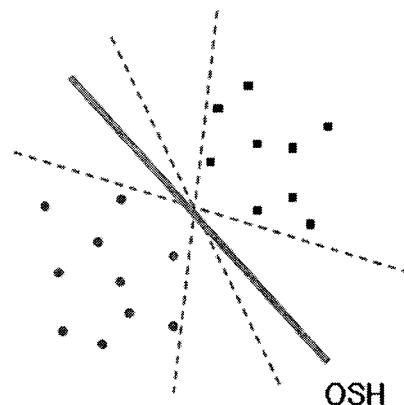


그림 8. 하이퍼분리면

히 많이 존재하게 되고, 굵은선으로 표시한 바와 같이 두 범주를 가장 효과적으로 나눌 수 있는 최적의 하이퍼분리면(Optimal Separating Hyperplane : OSH)이 존재하게 된다.

최적의 하이퍼분리면은 하이퍼분리면과 객체 사이의 거리를 최대로 하여 두 범주 간의 구분한다. 하이퍼분리면을 벡터의 내적으로 표현하면 (식 1)과 같다.

$$(W \bullet X) + b = 0 \quad W \in R^n, \quad b \in R \quad (1)$$

SVM의 학습 단계는 하이퍼분리면들 중에서 최적 하이퍼분리면을 찾는 과정이고 SVM의 테스트 단계는 임의의 N 차원 벡터의 입력 객체 $X_i \in R^n$ 을 두 개의 범주 $y \in \{+1, -1\}$ 로 예측하는 과정이다. 학습을 통해 계산된 최적 하이퍼분리면에 응답하는 분류 결정 함수 $f(X)$ 는 (식2)와 같다.

$$f(X) = \text{sgn}((W \bullet X_i) + b) \quad (2)$$

본 논문에서는 참고문헌 [18]의 방법을 이용하여 SVM 학습과 인식 알고리즘을 구현하였다. 이 방법에서는 최적화 하이퍼플레인을 찾기 위해 풀어야 할 QP(Quadratic Programming) 문제[18]를 작은 문제들로 분할하여 풀어나가는 방법을 사용함으로써 수행속도를 개선하였다.

SVM을 이용한 마커와 비마커 분류 과정이 그림 9에 나타나 있다. 그림 9(a)에는 700개의 마커와 700개의 비마커 영상에 대한 3차원 특징벡터가 나타나 있다. 마커는 'o' 기호로 표시하였으며 비마커는 '+' 기호로 표시하였다. 마커 학습 데이터와 비마커 학습

데이터를 구분하는 최적의 하이퍼분리면을 SVM 학습에 의하여 구한 결과가 그림 9(b)에 나타나 있다. 최적의 하이퍼분리면에 대한 식은 $2.3875x + 4.1207y + 2.4769z = 0$ 이다. 그림 9(b)에서는 최적의 하이퍼분리면에 의하여 마커들에 대한 특징벡터가 가려져서 나타나지 않고 비마커들에 대한 마커들이 나타나 있다. 그림 9(c)에는 최적의 하이퍼분리면을 옆에서 본 모습이 나타나 있다. 여기에서는 최적의 하이퍼분리면 위에 비마커들에 대한 특징 벡터가 존재하고 아래에는 마커들에 대한 특징 벡터가 존재하고 최적의 하이퍼분리면이 이들을 잘 분리하는 것을 알 수 있다.

마커와 비마커에 대한 특징벡터의 크기를 2차원으로 한 경우에는 그림 10과 같이 특징벡터들이 분포하고 마커와 비마커를 분리하는 하이퍼분리면이 존재하지 않는다. 따라서 특징 벡터의 크기를 적어도 3이상으로 설정해야 함을 알 수 있다.

3. 마커 인식

본 논문에서는 마커 인식을 위해 최소 거리 식별법을 사용한다. 패턴 인식에서 인식 대상이 되는 각 부류를 대표하는 패턴을 표준 패턴이라 하는데, 표준

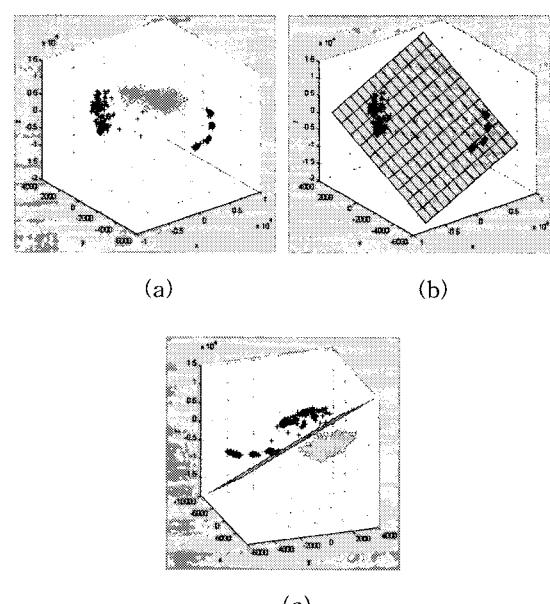


그림 9. (a) 마커와 비마커 학습데이터에 대한 특징벡터 (b) SVM 학습으로 구한 최적의 하이퍼분리면 (c) 최적의 하이퍼분리면을 옆에서 본 모습 ('o' : 마커, '+' : 비마커)

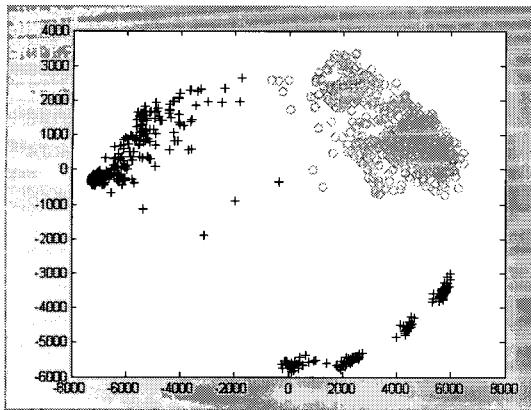


그림 10. 특징 벡터의 크기가 2인 경우의 학습 영상에 대한 특징 벡터 ('o' : 마커, '+' : 비마커)

패턴은 각 부류마다 하나 또는 여러 개 존재 할 수 있다. M 개의 부류 w_1, w_2, \dots, w_M 가 있고 부류 w_i 를 대표하는 표준 패턴을 y_i 라 하고 입력 패턴을 x 로 나타내기로 하자. 각 패턴을 N 차원의 특징 벡터로 나타내면 $y_i = (y_{i1}, y_{i2}, \dots, y_{iN})$, $x = (x_1, x_2, \dots, x_N)$ 이 될 것이다. 그리고 패턴 x 와 패턴 y_i 의 거리 $D(x, y_i)$ 는 다음 (식3)으로 계산할 수 있다. 최소 거리 식별법에서는 입력 패턴 x 를 $D(x, y_i)$ 의 값이 최소가 되는 부류 w_i 로 인식한다.

$$D(x, y_i) = \sqrt{\sum_{k=1}^N (x_k - y_{ik})^2} \quad (3)$$

앞 절에서 설명한 방법에 의하여 입력 영상으로부터 구한 정사각형 마커 영상의 크기는 64×64 이다. 영상의 픽셀 값을 그대로 특징 벡터로 사용하면 특징 벡터의 차원은 4096이 되고 최소거리 식별법에서 많은 양의 연산을 필요로 하게 된다. 따라서 인식의 수행 속도를 높이기 위해 특징 벡터의 차원을 줄이는 방법이 널리 사용되는데, 참고문헌 [12]에서는 주성분 분석을 이용하여 특징 벡터의 차원을 줄였다. 그러나 주성분 분석(PCA)은 각 클래스간의 관계를 고려하지 않고, 전체 데이터의 분산을 최대로 하는 방향으로만 선형 투사하기 때문에 각 클래스간의 겹침이 발생하는 단점이 있다. 그래서 본 논문에서는 PCA와 더불어 대표적인 특징 벡터 차원 축소 기법 중의 하나인 LDA를 사용하였다. LDA는 클래스간의 분산과 클래스내 분산의 비율을 최대화하는 방식으로 데이터에 대한 특징 벡터의 차원을 축소하는

방법이다[20].

그림 11에는 PCA와 LDA에 의한 차원 축소의 차이점을 잘 나타내고 있다. 원과 삼각형으로 표시된 두 클래스에 대한 객체들이 그림과 같이 2차원 상에 존재할 때에 PCA와 LDA는 이들을 1차원에 사상시킴으로써 차원을 축소할 수 있다. PCA는 각 클래스에 대한 고려를 하지 않고 전체 객체들의 분포를 분석하여 주성분 축을 구하기 때문에 그림에 나타나 있는 바와 같이 1차원으로 축소된 공간에서 두 클래스들이 섞여서 서로 구분이 어렵게 되는 문제가 발생할 수 있다. 반면에 LDA는 각 클래스의 구분이 잘되는 방향으로 주축을 설정함으로써 그림에 나타나 있듯이 두 클래스의 구분이 명확하게 되는 것을 알 수 있다.

실제 데이터에 대한 PCA와 LDA의 차이를 살펴보기 위해 마커와 비마커의 분류를 위해 사용하였던 1400개의 학습 영상에 LDA를 적용해보았다. 그림 12에는 LDA에 의해 차원이 축소된 특징 벡터들이 나타나 있다. 그림 11에서 PCA에 의해 차원이 축소된 경우와 비교해보면 PCA를 사용한 경우에는 마커와 비마커의 분리가 불가능하지만 LDA를 사용한 경우에 특징 벡터들이 마커와 비마커를 분류할 수 있음을 알 수 있다. 이와 같이 LDA를 이용한 차원 축소 방법이 클래스들 사이의 분류를 더 용이하게 하는 것을 알 수 있다. 여기에서는 단지 PCA와 LDA의 차이를 보이기 위해서 LDA를 적용해 본 것이고 마커와 비마커의 분류를 위해서는 PCA를 적용한 다음에 SVM을 적용하였다. LDA를 적용하면 분류를 더 용이하게 할 수 있지만 마커와 비마커를 분류하는데에는 PCA만 사용해서 차원을 축소하고 SVM을 적용해도 충분한 것으로 판단되어 LDA는 적용하지 않았다.

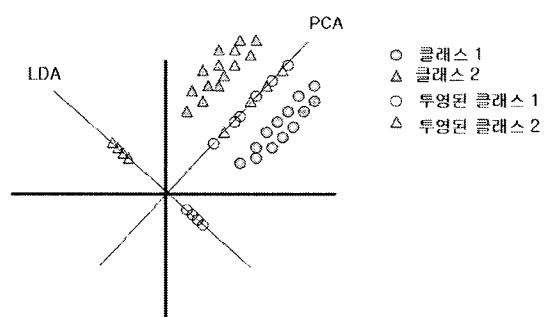


그림 11. PCA와 LDA의 차원 축소

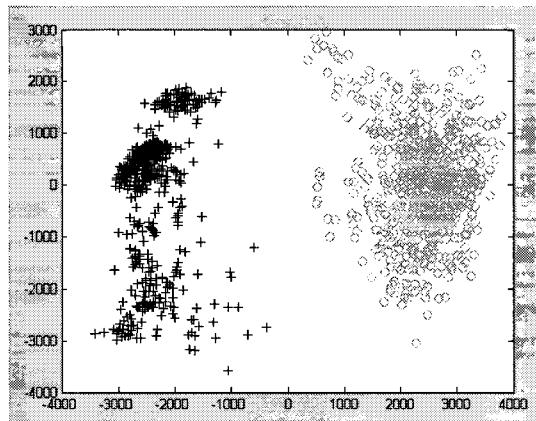


그림 12. LDA를 사용한 경우의 마커/비마커 분류 학습 영상에 대한 특징 벡터(‘o’ : 마커, ‘+’ : 비마커)

PCA와 LDA가 마커 인식에 어떤 영향을 미치는지를 살펴보기 위해 마커가 10개인 경우에 대하여 각 마커 당 10개의 학습 영상을 사용하여 PCA와 LDA를 적용해 보았다. 이때 특징 벡터의 크기는 10으로 설정하였다. 각 클래스의 특징 벡터들이 좁은 영역에 분포하고 즉, 클래스 내에서의 분산이 작고 전체적으로 볼 때에 클래스들이 넓게 퍼져 있으면 즉, 전체 분산이 크면 클래스 사이의 분류가 더 용이하게 된다. PCA와 LDA를 적용한 다음 각 클래스 내에서의 분산과 전체 분산을 측정한 결과가 표 1에 나타나 있다. 표에 나타나 있듯이 전체 분산을 클래스 분산으로 나눈 값이 LDA를 사용한 경우에 평균적으로 3.8배 정도 커지는 것을 볼 수 있다. 이와 같이

LDA를 사용하는 것이 PCA를 사용하는 것보다 클래스 간 변별력이 더 뛰어나다는 것을 알 수 있다.

4. 실험 결과

본 논문에서 제안된 방법은 CPU가 Pentium IV 3.2Hz이고 주메모리 용량이 1GB인 시스템에서 Visual C++ 6.0을 이용하여 구현하였다. 먼저 마커와 비마커 영상의 분류를 위한 실험을 수행하였다. 비마커 영상은 인위적으로 생성하지 않고 가정이나 사무실 등의 자연 영상으로부터 사각형 형태의 영상을 추출하여 사용하였다. 비마커 영상을 추출하기 위해서 노트북 컴퓨터에 USB 카메라를 장착하고 여러 장소를 옮겨 다니면서 실시간으로 영상을 캡쳐하였다. 캡쳐된 영상으로부터 실시간으로 사각형 영역을 검출한 다음에 이진화를 수행하고 워핑과 확대/축소 연산을 적용하여 일정한 크기의 정사각형 형태로 정규화하였다. 그림 13에는 일정한 크기로 정규화되고 이진화된 비마커 영상의 예가 나타나 있다. 자연 영상에서 나타나는 사각형 영역의 영상은 그림 13 (a)에 나타나 있는 바와 같이 마커와는 크게 구분되는 단순한 형태의 영상이 대부분이다. 마커 영상은 50 가지의 마커를 다양한 각도에서 촬영한 영상으로부터 검출하여 일정한 크기의 정사각형 형태로 정규화하였다. 그림 13 (b)에는 마커 영상의 예가 나타나 있다.

마커 영상과 비마커 영상은 각각 1000개씩 준비하

표 1. PCA와 LDA에 대한 클래스 내의 분산과 전체 분산 비율

마 커 클래스	PCA			LDA		
	클래스 분산(A)	전체 분산(B)	비율(B/A)	클래스 분산(A)	전체 분산(B)	비율(B/A)
1	6.9665e+005	2.0937e+006	3.0054	2.4543e+005	1.8632e+006	7.5916
2	7.3563e+005	2.8318e+006	3.8495	1.0544e+005	1.5227e+006	14.4414
3	3.5491e+005	2.3129e+006	6.5169	5.2218e+004	1.0070e+006	19.2845
4	6.6154e+005	1.3555e+006	2.0490	9.5782e+004	0.6972e+006	7.2790
5	9.3919e+005	1.5604e+006	1.6614	9.3392e+004	0.4939e+006	5.2885
6	8.8028e+005	1.3900e+006	1.5790	8.3486e+004	1.2049e+006	14.4324
7	5.5360e+005	0.1517e+006	0.2740	7.2682e+004	0.1508e+006	2.0748
8	3.1768e+005	0.2204e+006	0.6938	6.1860e+004	0.0948e+006	1.5325
9	4.5917e+005	0.4574e+006	0.9961	9.5229e+004	0.1517e+006	1.5930
10	5.7993e+005	0.0517e+006	0.0891	1.0514e+005	0.0112e+006	0.1065

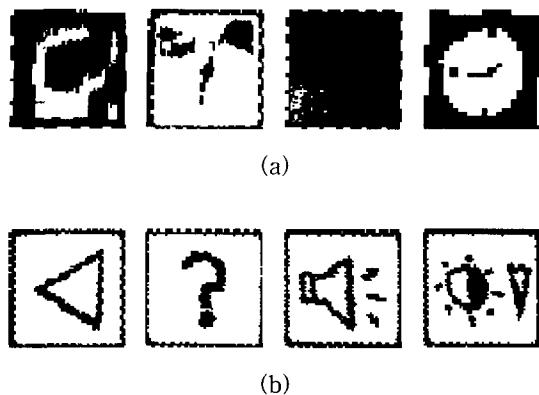


그림 13. (a) 비마커영상의 예 (b) 마커 영상의 예

였다. 그중에서 마커와 비마커 각각에 대하여 700개의 영상을 이용해서 SVM 학습을 수행하였고 나머지 300개씩의 영상에 대하여 SVM 분류 실험을 수행하였다. 600개의 영상에 대해 분류실험을 수행해본 결과 모든 경우에 마커와 비마커를 올바르게 구분하였다. 이와 같이 본 논문에서는 SVM을 이용하여 입력 영상에서 검출된 사각형 영역의 영상이 마커인지 아닌지를 판별함으로써 비마커를 마커로 오인하여 인식하려는 오류를 없앨 수 있었다. 본 실험에서 마커와 비마커를 100% 올바르게 분류할 수 있었던 것은 자연 영상에서는 마커와 유사한 형태의 패턴이 거의 존재하지 않기 때문인 것으로 분석된다.

마커의 인식 실험을 위하여 총 50개의 마커를 사용하였으며 마커가 20 종류, 30 종류, 40 종류, 50 종류일 때로 구분해서 실험함으로써 마커 수의 증가에 따른 인식률의 변화를 살펴보았다. 각 마커에 대하여 15 장의 영상을 촬영하였으며, 다양한 각도에서 촬영하고 각 마커 별로 촬영 각도를 비슷하게 유지하기 위하여 디지털 카메라를 이용하여 정지 영상으로 촬영하였다. 그림 14에는 마커 촬영 예가 나타나 있다.

촬영된 15장의 영상 중에서 10장의 영상은 학습에 사용하였고 나머지 5장의 영상은 인식실험에 사용하였다. 표 2에는 인식 실험 결과가 나타나 있다. 표 2에서 영상 수는 인식 실험에 사용된 마커 영상의 수를 나타낸다. 본 논문에서는 PCA를 수행한 다음에 LDA를 적용하여 특징 벡터를 생성하였다. LDA는 클래스 사이의 분산과 클래스 내의 분산의 비율을 최대화 하는 방식으로 특징 벡터의 차원을 축소하는 방법이므로 주성분축의 최대 개수는 클래스의 수와

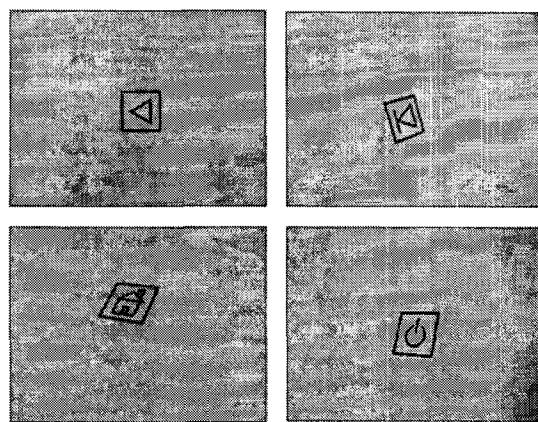


그림 14. 마커 촬영 예

같게 된다. 따라서 표 2에 나타나 있듯이 마커의 수와 같거나 작은 크기의 특징 벡터만 생성되었다. 반면에 PCA의 경우에는 주성분 축의 최대 개수가 학습에 사용된 영상 수와 같게 된다. 본 실험에서는 표 2에 나타나 있는 바와 같이 PCA의 경우 특징 벡터의 크기를 80개까지만 표시했는데, 특징 벡터의 크기를 더 증가 시켜도 인식률이 더 증가하지 않았고 오히려 더 떨어지는 경우도 발생하기 때문이다. 일반적으로 특징 벡터의 크기가 증가하면 인식률이 증가하다가 일정 수준에 도달하면 오히려 인식률이 떨어진다. 실험 결과에 따르면 본 논문에서 PCA와 LDA를 같이 사용함으로써 기존의 PCA만 사용한 경우보다도 특징 벡터의 크기가 작으면서도 인식률이 더 향상되었다. 특징 벡터의 크기가 작아지면 기억 공간의 사용량이 절약되고 연산량이 줄어드는 장점이 있다. 이와 같은 실험 결과에서 LDA를 사용함으로써 특징 벡터의 크기가 작아지면서도 인식률은 더 높아지는 것은 LDA의 분류 성능이 PCA보다 더 뛰어나기 때문인 것으로 분석된다.

표 2. 인식 결과

마 커 수	영 상 수	특징 벡터의 크기에 따른 인식률										
		PCA + LDA				PCA[12]						
		20	30	40	50	20	30	40	50	60	70	80
20	100	96				90	91	93	95	95	95	95
30	150	96	96.7			90.7	92.7	94	95.3	95.3	96	96.7
40	200	96.5	96.5	97		90.5	93.5	95	95.5	96.5	97	96
50	250	94.8	98	97.6	98	92	93.2	94.8	96	96	96.8	97.6

표 3에는 SVM 분류기를 추가함으로써 얼마만큼의 CPU 시간이 추가되었는지가 나타나 있다. 마커 검출은 하나의 마커 검출에 소요되는 시간을 비교하였다. 본 논문의 방법과 참고문헌 [12]의 방법 모두 CPU가 Pentium IV 3.2Hz이고 주메모리 용량이 1GB인 PC를 사용하여 동일한 환경 하에서 실험을 수행하였으며 USB 카메라에서 입력된 320 x 240 크기의 영상에 대하여 마커를 검출하였다. 본 논문에서 제안된 방법에서는 마커 검출에 0.088초가 소요되어 참고문헌 [12] 보다 약간 증가하였지만 초당 11.4프레임의 처리 속도를 낼 수 있으므로 실시간 처리에 큰 문제가 되지는 않는다. 인식 단계에는 PCA만 적용하는 것과 PCA와 LDA를 함께 적용하는 것 사이에 수행 시간상의 차이는 없다. 다만 학습단계에서 LDA 수행을 위한 추가 시간이 소요되지만 학습은 사전에 한번만 수행되는 것이므로 인식의 수행 시간에는 영향을 미치지 않는다.

5. 결 론

본 논문에서는 SVM과 LDA를 이용하여 마커 검출 및 인식의 성능을 향상시키는 방법을 제안하였다. 본 논문에서는 입력 영상을 이진 영상으로 변환하고 객체들의 윤곽선을 추출한 다음에 윤곽선의 기하학적 특징을 이용하여 사각형을 찾고 위평 기법을 이용하여 사각형 영상을 정사각형 형태로 정규화한 다음에 PCA를 통하여 사각형 영상으로부터 특징 벡터를 추출한다. 특징 벡터에 SVM 분류기를 적용하여 사각형 영역의 영상이 마커 영상인지 비마커인지를 판별함으로써 마커가 아닌 시각형 영상은 인식 대상에서 제외시킨다. 마커 영상으로 판별된 영역에 대해서 LDA를 적용하여 특징 벡터의 크기를 줄이고 표준 마커에 대한 특징 벡터와의 최소 거리법에 의해 마커의 종류를 인식한다. 마커의 종류를 변화시켜가면서 인식 실험을 해본 결과 마커의 종류가 증가하더라도 신뢰할만한 인식률을 얻을 수 있었고 실시간에 가까운 수행 속도를 얻을 수 있었다.

표 3. 수행 속도 비교

	소요 시간 (단위: 초)	
	본 논문	참고문헌[12]
마커 검출	0.088	0.059

참 고 문 헌

- [1] I. Poupyrev, H. Kato, and M. Billinghurst, *ARToolkit User Manual, Version 2.33*. Human Interface Technology Lab, University of Washington, 2000.
- [2] M. Fiala, "ARTag, a fiducial marker system using digital techniques," *In Proc. of Conference on Computer Vision and Pattern Recognition*, pp. 590-596, 2005.
- [3] X. Zhang, Y. Genc, and N. Navab, "Mobile computing and industrial augmented reality for real-time data access," *In Proc. of IEEE Int'l Conference on Emerging Technologies and Factory Automation*, pp. 583-588, 2001.
- [4] M. Appel and N. Navab, "Registration of technical drawings and calibrated images for industrial augmented reality," *In IEEE Workshop on Applications of Computer Vision*, pp. 48-55, 2000.
- [5] N. Osawa and K. Asai, "Distributed automatic camera control system tracking markers for distance education," *In Proc. of International Conference on Information Technology: Research and Education*, pp. 377-381, 2005.
- [6] H. Slay, B. Thomas, and R. Vernik, "Using ARToolkit for passive tracking and presentation in ubiquitous workspaces," *In Proc. of International Augmented Reality Toolkit Workshop*, pp. 46-53, 2003.
- [7] A. Woolard, V. Lalioti, N. Hedley, J. Julien, M. Hammond, and N. Carrigan, "Using ARToolKit to prototype future entertainment scenarios," *In Proc. of International Augmented Reality Toolkit Workshop*, pp. 69-70, 2003.
- [8] W. Piekarski and B.H. Thomas, "Using ARToolKit for 3D hand position tracking in mobile outdoor environments," *In Proc. of International Augmented Reality Toolkit*, 2002.
- [9] Y. Nakazato, M. Kanbara, and N. Yokoya, "Wearable augmented reality system using

- invisible visual markers and an IR camera," *Proc. of International Symposium on Wearable Computers*, pp. 198~199, 2005.
- [10] Y. Kishino, M. Tsukamoto, Y. Sakane, and S. Nishio, "Realizing a visual marker using LEDs for wearable computing environment," *In Proc. of International Conference on Distributed Computing Systems Workshops*, pp. 314~319, 2003.
- [11] G. Adorni, G. Destri, M. Mordonini, and F. Zanichelli, "Robot self-localization by means of vision," *In Proc. of Euromicro Workshop on Advanced Mobile Robot*, pp. 160~165, 1996.
- [12] 강선경, 이상설, 정성태 "사각형 마커 검출 및 인식 시스템 개발," *한국컴퓨터정보학회논문지*, 제11권 4호, pp. 97~107, 2006. 9.
- [13] I. Pitas, *Digital Image Processing schemes and Application*, John Wiley and Sons Inc., New York, 2000.
- [14] J. Hershberger and J. Snoeyink, "Speeding up the Douglas-Peucker line simplification algorithm," *In Proc. 5th Intl. Symp. Spatial Data Handling. IGU Commission on GIS*, pp. 134~143, 1992.
- [15] R. Crane, *A simplified approach to image processing*, Prentice Hall, 1997.
- [16] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer-verlag, New York, 1995.
- [17] CJC. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *In Data Mining and Knowledge Discovery*, Vol. 2 No. 2, pp. 121~167, 1998.
- [18] J. Platt, "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines," *Microsoft Research Technical Report MSR-TR-98-14*, 1998.
- [20] K. Fukunaga, *Statistical Pattern Recognition*, Academic Press, New York, 1989.



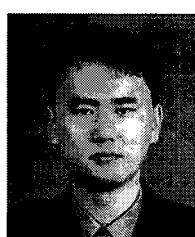
강 선 경

2000년 2월 원광대학교 전기전자 및 정보 공학부(공학사)
2004년 2월 원광대학교 정보컴퓨터교육전공(교육학석사)
2006년 3월~현재 원광대학교 컴퓨터 공학과 박사과정
관심분야 : 휴먼 컴퓨터 인터페이스, 영상처리, 영상인식, 임베디드시스템



소 인 미

1994년 2월 원광대학교 컴퓨터 공학과(공학사)
2002년 8월 원광대학교 정보컴퓨터교육전공(교육학석사)
2004년 3월~현재 원광대학교 컴퓨터공학과 박사과정
관심분야 : 영상인식, 컴퓨터 비전, 영상처리



김 영 운

2003년 2월 원광대학교 전기전자 및 정보 공학부(공학사)
2005년 8월 원광대학교 컴퓨터 공학과(공학 석사)
2006년 3월~현재 원광대학교 컴퓨터 공학과 박사과정
1994년~1999년 영원한 친구 개발팀장
2000년~현재 파라(PARA) 대표
2003년~현재 원광보건대학 정보컨텐츠과 겸임교수
관심분야 : 컴퓨터그래픽스, 영상처리, EAI, XML



이 상 설

- 1984년 2월 고려대학교 전자공
학과 졸업
1989년 2월 한국과학기술원 전
기및전자공학과 석사학
위 취득
1994년 2월 한국과학기술원 전
기및전자공학과 박사학

위 취득

1994년 ~ 현재 원광대학교 전기전자및정보공학부 교수
관심분야 : 병렬컴퓨터구조, SoC, 영상 및 통신 VLSI,
임베디드시스템



정 성 태

- 1987년 2월 서울대학교 컴퓨터
공학과(공학사)
1989년 2월 서울대학교 컴퓨터
공학과(공학석사)
1994년 8월 서울대학교 컴퓨터
공학과(공학박사)
1995년 3월 ~ 현재 원광대학교

전기전자 및 정보공학부 교수

관심분야 : 영상인식, 영상처리, 컴퓨터 그래픽스