

Support Vector Machines을 이용한 다중 클래스 문제 해결

(Solving Multi-class Problem using Support Vector Machines)

고 재 필^{*}

(Jaepil Ko)

요 약 최근 기계학습 분야에서 커널머신을 이용한 대표적 학습기로 Support Vector Machines (SVM)이 주목 받고 있다. SVM은 통계적 학습이론에 기반하여 뛰어난 일반화 성능을 보여주며, 다양한 패턴인식 문제에 적용되고 있다. 그러나, SVM은 이진 분류기이므로 일반적인 다중 클래스 문제에 곧바로 적용할 수 없다. SVM을 다중 클래스 문제의 하나인 얼굴인식에 도입하기 위한 방법으로는, One-Per-Class와 All-Pairs가 대표적이다. 상기 두 방법은 다중 클래스 문제를 여러 개의 이진 클래스 문제로 분할하고, 이들을 다시 종합하여 최종 결정을 내리는 출력코딩이라는 일반적인 방법에 속한다. 본 논문에서는 이진 분류기인 SVM의 다중 클래스 분류기 확장 방안으로 출력코딩 방법론을 설명한다. 또한 출력코딩 방법론의 대표적인 이론적 기반인 ECOC(Error-Correcting Output Codes)를 근간으로 하는 새로운 출력코딩 방법론을 제안하고, 얼굴인식 실험을 통해 SVM을 기반 분류기로 사용할 경우의, 출력코딩 방법의 특성을 비교·분석한다.

키워드 : 서포트벡터머신, 출력코딩방법론, ECOC, 다중분류기, 얼굴인식

Abstract Support Vector Machines (SVM) is well known for a representative learner as one of the kernel methods. SVM which is based on the statistical learning theory shows good generalization performance and has been applied to various pattern recognition problems. However, SVM is basically to deal with a two-class classification problem, so we cannot solve directly a multi-class problem with a binary SVM. One-Per-Class (OPC) and All-Pairs have been applied to solve the face recognition problem, which is one of the multi-class problems, with SVM. The two methods above are ones of the output coding methods, a general approach for solving multi-class problem with multiple binary classifiers, which decomposes a complex multi-class problem into a set of binary problems and then reconstructs the outputs of binary classifiers for each binary problem. In this paper, we introduce the output coding methods as an approach for extending binary SVM to multi-class SVM and propose new output coding schemes based on the Error-Correcting Output Codes (ECOC) which is a dominant theoretical foundation of the output coding methods. From the experiment on the face recognition, we give empirical results on the properties of output coding methods including our proposed ones.

Key words : Support Vector Machines, Output Coding, ECOC, Multiple Classifiers, Face Recognition

1. 서론

최근 기계학습 분야에서 커널머신을 이용한 대표적 학습기로 Support Vector Machines(SVM)이 주목 받고 있다[1]. SVM은 통계학자인 Vapnik에 의해 1995년 제안된 것으로, 구조적 위험 최소화를 근간으로 하는 통계적 학습 이론에 기반하여 뛰어난 일반화 성능을 보여

준다[2,3]. SVM은 패턴인식, 데이터마이닝, 생체인식 등의 다양한 응용분야를 가지며, 각 분야에서 SVM을 적용한 사례들이 성능향상을 보고함으로써 SVM의 이론적 근거를 공고히 하고 있다[4]. 그러나, SVM은 이진 분류기이므로 일반적인 다중 클래스 문제에 곧바로 적용할 수 없다. SVM을 다중클래스 문제에 적용하기 위한 방법은 크게 두 가지로 구분된다. 첫째, SVM 자체를 다중 클래스분류기로 일반화 하는 것이 있으며, 둘째 이진 SVM을 다중으로 사용하는 방법이 있다. 대표적인 전자의 방법으로 Weston이 제안한 방법론[5] 학습 알고리즘의 계산복잡도가 매우 높아 실용화하기 어려울 뿐

· 본 연구는 금오공과대학교 학술연구비에 의하여 연구된 논문임

† 정 회 원 : 금오공과대학교 컴퓨터공학과 교수

nonezero@kumoh.ac.kr

논문접수 : 2005년 4월 15일

심사완료 : 2005년 10월 11일

만 아니라, 성능 또한 후자와 유사하다[6]. 따라서, 실제 응용에서는 적용의 용이성과 성능에 힘입어, 다중 클래스 문제를 여러 개의 이진 클래스 문제로 분할하고, 이들을 다시 종합하여 최종 결정을 내리는 출력코딩[7] 방법인 후자가 선호된다.

출력코딩 방법으로써, SVM을 다중 클래스 문제의 하나인 얼굴인식에 도입하기 위한 방법으로는, OPC(One-Per-Class)와 All-Pairs가 대표적이다[8]. SVM을 분류기로 사용하고자 하는 얼굴인식과 같은 응용에서 출력코딩방법은 매우 유용하다. 얼굴인식 연구는 주로 얼굴의 지역적 특징추출 보다는 얼굴영상 자체를 인식에 사용하기 때문에 얼굴영상 표현방법에 집중하여 왔다. 이에 따라, PCA(Principal Component Analysis)[9], LDA(Linear Discriminant Analysis)[10], ICA(Independent Component Analysis)[11], LFA(Local Feature Analysis)[12], NMF(Non-negative Matrix Factorization)[13], Direct-LDA[14], Kernel-PCA[15], Kernel-LDA[16], Kernel-ICA[17], Kernel Direct-LDA[18] 등 무수히 많은 특징추출방법이 연구되고 얼굴인식에 적용되었다. 이러한 방법들은 변환 방법의 기준이 선형/비선형 또는 전역/지역적 방법이나에 따라 표 1과 같이 구분될 수 있다.

표 1 특징추출 방법 구분

구분	전역	지역
선형	PCA, LDA, Direct-LDA	ICA, LFA, NMF
비선형	Kernel-PCA, Kernel-LDA, Kernel Direct-LDA	Kernel-ICA

특징추출에 대한 활발한 연구에 비해, 얼굴 매칭 단계에서는 단순한(simple) 분류기인 NN(Nearest Neighbor) 분류기[19]가 적용되어 왔다. 그러나, NN 분류기는 얼굴영상의 다양한 변화에 대처하기 어려울 뿐만 아니라, 얼굴영상이 주로 고차원 공간에서 표현되는 관계로 차원의 저주문제[19]에 취약할 수 있다. 이에 비해 SVM은 차원의 크기에 덜 민감하다고 알려져 있어 얼굴인식에 매우 효과적이다. 따라서, 얼굴영상의 다양한 변화를 수용할 수 있는 신경망과 같은 복잡한(complex) 분류기로서 SVM의 도입 필요성과 클래스 수가 유동적이라는 얼굴인식 응용을 고려하면, 분류할 클래스의 추가나 제거에 비교적 자유로운 출력코딩 방법은 적절한 선택이 될 수 있다.

출력코딩 방법의 성능은 기반 분류기의 성능에 의존적인데, SVM은 기반 분류기로서 매우 적절하며, 얼굴인식에 성공적으로 적용되었다[8,20]. 대표적인 출력코딩 방법에는 OPC(one-per-class)[21], All-Paris[22], ECOC

(error-correcting code)[23] 등이 있다.

OPC에서 각 이진 분류기는 하나의 클래스와 나머지 클래스들을 구분하고, All-Paris에서는 하나의 클래스를 또 다른 하나의 클래스와 구분하는 방식이다. 출력코딩 방법의 주요한 이론적인 기반인 ECOC는 여러 이진 분류기에 클래스를 중복 학습시켜 몇 개의 이진 분류기에서 생길 수 있는 오류를 극복할 수 있다.

본 논문에서는 SVM을 다중 클래스 문제에 적용하기 위한 방법으로써, 얼굴인식 등에서 단편적으로 적용된 OPC, All-Pairs 방법을 출력코딩이란 일반적인 방법론으로 정형화하여 재기술 하고, 기존의 주요 방법론들 소개하고 장·단점을 설명한다. 그리고, 출력코딩 방법론의 주요 이론적 기반인 ECOC를 근간으로 하는 새로운 출력코딩 방법들을 제안하고, 얼굴인식 실험을 통해 SVM을 기반 분류기로 사용할 경우의, 출력코딩 방법의 특성을 비교·분석한다.

본 논문의 구성은 다음과 같다. 2장에서는 SVM에 대해 간략히 설명하고, 3장에서는 출력코딩방법을 정형화하여 기술하고, 주요 방법론을 소개한다. 4장에서는 출력코딩 방법의 특성분석을 위한 제안 방법론을 설명하고, 5장에서는 실험결과와 분석을, 그리고 마지막 6장에서 결론을 내린다.

2. Support Vector Machines

두 클래스 군집을 선형 분리하는 결정평면과 가장 가까운 샘플을 Support Vector(SV)라고 하고, SV와 결정평면과의 거리를 마진이라고 한다. SVM은 분리 마진을 최대화하는 결정평면을 구한다. 그림 1의 왼쪽은 선형 분리 가능한 샘플에 대한 결정평면 예를 보여준다. 이때 무수히 많은 결정평면이 존재할 수 있으나, 그림의 오른쪽과 같이 분리마진을 최대화하는 것을 구함으로써, 높은 일반화 성능을 기대할 수 있다. 그림에서 큰 동그라미는 SV를 나타낸다.

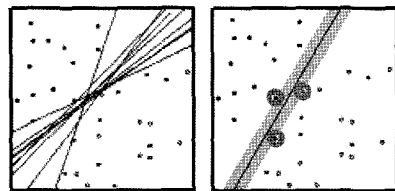


그림 1 결정평면에 따른 분리 마진

학습집합 $\{(x_1, y_1), \dots, (x_l, y_l)\}$ 이 주어졌을 때, 결정평면은 $w \cdot x + b = 0$ 이다. 여기서, $x \in R^n$ 는 입력벡터이고, w 는 가중치 벡터, $y \in \{-1, +1\}$ 는 타겟값, b 는 바이어스

를 나타낸다. 이때, 입력벡터 \mathbf{x} 에서 결정평면까지의 마진은 $2\|\mathbf{w}\|$ 이며, 이를 최대화하기 위해, 식 (1)과 같이 비용함수를 정의한다.

$$\Phi(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2 \tag{1}$$

타겟값과 결정평면식의 곱은 항상 양수라는 제약조건을 적용하면, 식 (1)은 라그랑지안 최적화 기법을 이용해 식 (2)와 같이 최적의 \mathbf{w}_o 와 b_o 을 구할 수 있다.

$$\mathbf{w}_o = \sum_{i=1}^l \alpha_{oi} y_i \mathbf{x}_i, \quad b_o = -\frac{1}{2} \mathbf{w}_o \cdot [\mathbf{x}_r + \mathbf{x}_s] \tag{2}$$

여기서 α 는 라그랑쥐 계수이다. 이때, SV인 \mathbf{x} 에 대해서만 라그랑쥐 계수가 0이 아닌 값을 가지므로써, SV만이 \mathbf{w} 를 계산하는데 의미를 가지게 된다. 학습 데이터 중 마진에 직접 관여하는 벡터만 골라내는 이러한 능력이 SVM의 큰 특징 중 하나이다.

선형적으로 분리 가능하지 않는 경우는 오류에 대한 패널티($\xi \geq 0$)를 적용하여 수식 (3)과 같은 비용함수를 통해 최적화가 가능하다. 대신 경험적으로 결정해야 하는 마진의 최대화와 분류 어려움에 대한 조절 인자인 C 가 요구된다.

$$\Phi(\mathbf{w}, \xi) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^l \xi_i \tag{3}$$

분리면이 비선형인 경우에는 커널 함수를 도입하여 특징공간에서의 선형분리를 시도한다.

$$\sum_{i=1}^{N_s} \alpha_i y_i K(\mathbf{s}_i, \mathbf{x}) + b = 0 \tag{4}$$

여기서, N_s 는 SV의 수이고, \mathbf{s}_i 는 SV를, 그리고 K 는 커널 함수를 나타낸다. 커널 함수는 입력벡터를 고차원의 특징벡터로 변환한 후 내적을 구하는 과정을 대치한 함수로, 표 2와 같은 함수들이 주로 사용된다.

표 2 대표적 커널 함수

종류	커널 함수	비고
Polynomial	$(\mathbf{x}^T \mathbf{x}_i + 1)^p$	p 는 degree
Radial Basis Function Network	$\exp\left(-\frac{1}{2\sigma^2} \ \mathbf{x} - \mathbf{x}_i\ ^2\right)$	σ^2 는 variance
Perceptron	$\tanh(\beta_0 \mathbf{x}^T \mathbf{x}_i + \beta_1)$	β_0 은 slope, β_1 은 bias

SVM에 대한 보다 자세한 사항은 참고문헌 [3,24]을 참고하기 바란다. 마지막으로 SVM이 다음 장에서 소개

될 출력코딩 방법의 기반 분류기로 사용되었을 때의 장점은 다음과 같다. 첫째, SVM 분류기의 실수 출력은 출력코딩에서의 마진디코딩 (3장 참조) 적용이 가능하며, 둘째, 커널 함수를 도입하여 비교적 복잡한 문제도 학습할 수 있다는 점이다.

3. 다중 클래스 학습에서의 출력코딩

3.1 출력코딩 방법이란

출력코딩의 명칭: 다중 클래스 문제를 여러 개의 이진 클래스 문제로 분할하여 해결하는 방식에 대한 다양한 이름이 있어왔다. 출력코딩은 오류가 있는 통신채널에서 안전한 전송을 위한 코딩이론을 다중 분류기 문제에 도입한 ECOC를 주요 이론적 근거로 삼고 있으며, 출력코딩(OC)이란 용어는 여기서 유래를 찾을 수 있다. ECOC 출현 이전에 이미 널리 사용되고 있었지만 이론적 배경이 없던 OPC는 ECOC 개념으로 통합할 수 있다. ECOC 및 OPC는 하나의 이진 분류기에 모든 클래스 샘플이 관여한다는 공통점이 있다. 이에 반해, All-Pairs는 학습에 단지 두 개의 클래스만 관여한다. 즉, 학습에 관여하지 않는 클래스가 존재한다. 이점은 All-Pairs 방식을 OPC 및 ECOC와 구분 짓는 매우 중요한 기준이 된다.

본 논문에서는 All-Pairs와 ECOC 개념을 하나로 묶어 일반화된 ECOC라고 정의하고, 이를 출력코딩이라고 명명한다. 참고로, 일반화된 ECOC 개념은 논문 [7]에서 정립하였다. 그림 2는 출력코딩 방법론에 대한 명칭과 단일화 흐름을 보여준다.

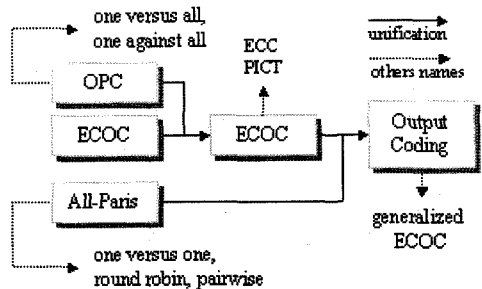


그림 2 출력코딩 방법론의 명칭 및 단일화 흐름

정형화: 출력코딩 방법은 분할과정과 복원과정으로 구성되어 있다[25]. 분할 또는 인코딩 단계에서는 K 개의 클래스를 L 개의 이진 분류기로 분할하는 것을 나타내는 분할행렬 $D \in \{-1, 0, +1\}^{L \times K}$ 을 생성한다. 이진 분류기 f 는 분할행렬의 행 $D(l, \cdot)$ 에 따라 학습된다. 만약 $D(l, k) = +1$ 이면 클래스 K 에 속한 모든 샘플들은 양의

클래스로, $D(l, k) = -1$ 이면 음의 클래스로 f_l 을 학습하고, $D(l, k) = 0$ 이면 f_l 을 학습하는데 사용되지 않음을 의미한다[26]. 여기서 D 의 행은 코드워드(codewords)라고 불리운다[27].

복원 또는 디코딩 단계에서는, 간단한 최근인접(nearest-neighbor) 규칙이 주로 사용되며, 입력벡터 \mathbf{x} 에 대한 클래스는 $f(\mathbf{x})$ 와 컬럼 $D(k, \cdot)$ 사이의 어떤 유사도 측정치: $s: \mathbf{R}^L \times \{-1, 0, 1\}^L \rightarrow [-\infty, \infty]$ 를 최대화 하는 클래스로 결정된다[27].

$$\text{class_output} = \arg \max_k s(f(\mathbf{x}), D(\cdot, k)) \quad (5)$$

유사도가 만약 식 (6)과 같이 실수 값을 가지는 마진에 기반하여 정의가 된 경우, 이를 마진(margin) 디코딩이라고 부른다. 여기서 마진(margin)은 이진 분류기의 출력값과 대응되는 타겟값의 곱으로 정의된다.

$$s(f(\mathbf{x}), D(\cdot, k)) = \sum_l f_l(\mathbf{x}) D(l, k) \quad (6)$$

이진 분류기가 식 (7)과 같이 이산적인 2개의 값 $h(\mathbf{x}) \in \{-1, 1\}$ 을 출력하는 경우를 해밍(hamming) 디코딩이라고 한다.

$$s_H(h(\mathbf{x}), D(\cdot, k)) = 0.5 \times \sum_l (1 + h_l(\mathbf{x}) D(l, k)) \quad (7)$$

일반적으로 실험적 및 이론적 두 경우 모두 마진디코딩이 해밍디코딩보다 우월한 결과를 보인다[7].

3.2 대표적 출력코딩 방법

출력코딩 방법은 분할방법 및 복원방법에 따라 구분되나, 본 논문에서는 분할방법을 기준으로 대표적 방법론을 소개한다. 보다 자세한 구분 및 이에 따른 방법론 소개는 [28]을 참조하기 바란다.

One Per Class(OPC): 각 이진 분류기 f_l 는 하나의 클래스와 나머지 다른 클래스들을 구분하도록 학습된다. 만약 K 개의 클래스가 있는 경우, 양의 클래스를 하나씩만 가지는 K 개의 이진 분류기를 생성하게 된다. 복원은 주로 각 이진 분류기의 출력 값이 가장 큰 분류기에 학습된 양의 클래스로 결정하는 방식을 따른다. 본 방식에 의해 생성되는 이진 클래스 문제는 매우 복잡한 문제가 될 소지가 크다[26]. 왜냐하면, 학습에 사용되는 음의 클래스는 이질적인 다양한 클래스들로 구성되기 때문이다. 복잡한 문제에 대한 학습은 쉬운 문제에 대한 학습보다 학습을 위한 샘플을 더 많이 요구하게 되고, 이렇게 학습된 이진 분류기 출력 값에 대한 신뢰도 역시 저하될 수 있다. 그러나, 얼굴인식과 같이 하나의 클래스에 속하는 학습 샘플의 수가 적은 응용에서는 하나의 이진 분류기에 모든 클래스 샘플이 사용된다는 장점도 있다.

All-Pairs: 각 이진 분류기 f_{ij} 는 클래스 i 와 클래스 j 를 구분하도록 학습되고, 이러한 이진 분류기는 가능한

클래스 조합쌍 만큼 생성한다. 따라서, 풀어야 할 이진 문제는 단지 두 개의 클래스에 속한 샘플들을 구분하는 비교적 단순한 문제로 귀결된다. 단순한 분류 문제의 학습은 비교적 적의 학습샘플로 가능하며, 이렇게 학습된 이진 분류기에 대해 높은 신뢰도의 출력 값을 기대할 수 있다. All-Paris는 분할과정에서 이러한 장점이 있는 반면, 디코딩 과정에서는 다음과 같은 단점이 존재한다. 만약 클래스 i 및 클래스 j 에 속하지 않은 입력샘플 \mathbf{x} 가 이진 분류기 f_{ij} 에 주어지면, 즉 학습에 관여하지 않은 입력이 주어지면 이진 분류기는 의미 없는 출력을 하게 된다[29]. 이를 *넌센스출력의문제*라고 한다. 이로 인해, 클래스의 수가 늘어날수록 전체 성능은 저하된다. 따라서, 성능향상을 위해서는 입력 샘플의 클래스가 학습되어 있는 (있을 것으로 기대되는) 관련 이진 분류기들의 의견만 고려하는 방법론이 필요하다. K 클래스 문제인 경우, 본 분할 방법을 통해 만들어지는 이진 문제의 개수는 $K C_2 = K(K-1)/2$ 이다. 클래스 결정은 해밍디코딩에 기반한 투표방식[30]이 주로 사용된다. 디코딩 시간을 줄이기 위해 트리구조의 디코딩 방식이 제안되었다[8,31]. 디코딩 과정에서 사후확률 개념을 도입한 방식은 Pair-Wise Coupling(PWC)[30]라고 불리운다.

Error-Correcting Output Codes (ECOC): ECOC의 분할방식은 코딩이론에서 도입되었다. ECOC는 이진 분류기의 일부가 잘못된 분류를 하더라도 정정능력을 부여한다. 이를 위해서는 어떤 클래스를 학습하는 이진 머신을 여러 개 둬으로써 얻어진다. 하나의 클래스를 여러 개의 머신이 담당하게 됨으로써, 하나가 오류를 범하더라도 다른 하나가 올바른 답을 출력할 가능성이 생긴다. 이에 따라, 4개의 클래스를 구분하기 위해, OPC기법은 하나의 분류기가 하나의 클래스를 담당하도록 4개의 이진 분류기를 필요로 하지만, ECOC는 여러 개의 분류기가 하나의 클래스를 담당하게 됨으로 4개 이상의 이진분류기를 필요로 하게 된다. 앞서 설명한 분할행렬로 말하자면 ECOC가 더 많은 행을 요구하게 되는 것이다.

행을 늘림으로써 정정능력이 향상될 수 있지만, 단순히 행을 늘리는 것이 아니라 예리정정능력을 좌우하는 해밍거리를 고려해야 한다. 예리정정 능력은 분할행렬의 "열간 해밍거리"에 의존하며, 정정 가능한 오류의 최대 개수는 식 (8)과 같다.

$$\text{No. of Correct} = \frac{\Delta - 1}{2} \quad (8)$$

여기서, Δ 는 분할행렬의 각 열이 갖는 최소해밍거리를 나타낸다.

분할행렬의 열간 해밍거리를 증대시키기 위한 일차적인 조건은 행의 수를 늘이는 것이다. 전통적인(현재 설

명하고 있는) ECOC는 분할행렬의 행을 확대시킴으로써, 즉 학습해야 할 이진 머신의 수를 확대시킴으로써, 성능향상을 피할 수 있으며 이는 ECOC의 근간이 되어 왔다. All-Pairs 역시 매우 많은 이진 머신으로 구성되는데, 전통적인 ECOC와의 차이는 분할행렬에 "0"이 들어간다는 점이다. 분할행렬에 0을 도입함으로써 All-Pairs 기법을 포함할 수 있는 일반화된 ECOC를 출력 코딩이라고 본 논문에서는 사용한다(3.1 참조).

ECOC에 대한 더 자세한 사항은 참고문헌 [23]에서 확인할 수 있으며, 지금까지 ECOC는 다음과 같은 다양한 응용에 성공적으로 적용되어 왔다: 문서분류[32,33,34], 음성인식[35], 구름분류[36], 얼굴인식[37].

Correcting Classifier(CC): CC는 ECOC의 근간이 되는, 분할행렬의 해밍거리를 크게 하는데 초점을 두어 개발한 방법론으로 전형적인 ECOC 방법론의 하나이다 [29]. 또한 All-Pairs의 난센스출력의 문제를 다루기 위해 제안되었다. 각 이진 분류기 f_{ij} 는 클래스 i, j 와 나머지 클래스들을 구분하기 위해 학습된다. 따라서, 본 방법에 의해 생성되는 이진 문제의 수는 All-Pairs와 동일하다. CC는 OPC, All-Paris에 비해 급격한 성능향상 결과를 보여주었는데, 이는 CC의 분할행렬이 식 (9)와 같이 클래스 수의 약 2배에 해당하는 큰 해밍거리를 갖게 되어 에러정정능력이 높기 때문으로 추정된다.

$$\Delta = 2(K - 2) \tag{9}$$

단점으로는 하나의 이진 분류기를 학습하기 위해 모든 클래스의 샘플들이 관여한다는 점이다. 이는 OPC와 동일하지만, 클래스의 수가 K 일 때, 필요한 이진 분류기의 수는 OPC 및 CC에 대하여 각각 K , $K C_2 = K(K-1)/2$ 이므로 전체 분류기를 학습하는데 걸리는 시간이 급격히 늘어난다.

그림 3은 4개의 클래스에 대한 OPC, Paris, CC의 분할행렬을 보여준다. 각 행은 하나의 이진 문제에, 각 열은 하나의 클래스에 대응된다. (a)와 (c)에서는 모든 행이 1 또는 -1로 구성되어 있어 하나의 이진 분류기를

$\begin{bmatrix} +1 & -1 & -1 & -1 \\ -1 & +1 & -1 & -1 \\ -1 & -1 & +1 & -1 \\ -1 & -1 & -1 & +1 \end{bmatrix}$	$\begin{bmatrix} +1 & -1 & 0 & 0 \\ +1 & 0 & -1 & 0 \\ +1 & 0 & 0 & -1 \\ 0 & +1 & -1 & 0 \\ 0 & +1 & 0 & -1 \\ 0 & 0 & +1 & -1 \end{bmatrix}$	$\begin{bmatrix} +1 & +1 & -1 & -1 \\ +1 & -1 & +1 & -1 \\ +1 & -1 & -1 & +1 \\ -1 & +1 & +1 & -1 \\ -1 & +1 & -1 & +1 \\ -1 & -1 & +1 & +1 \end{bmatrix}$
(a)	(b)	(c)

그림 3 클래스의 수 $K=4$ 일 때의 OPC, All-Paris, CC의 분할행렬

학습하는데 모든 클래스가 관여함을 알 수 있다. 이에 반해, (b)에서는 0인 요소가 있으며 그 열의 클래스는 학습에 관여하지 않음을 나타낸다.

4. ECOC 개념 기반 OPC 변형 코딩 방법론

본 장에서는 ECOC 개념을 가지고 OPC기반의 새로운 방법론을 제안하고, 이들의 특성을 설명하기 위한 새로운 측정치를 정의한다. 또한 마진디코딩 기반 출력코딩 방법의 성능분석을 위해 문제복잡도를 새롭게 정의한다. 해밍디코딩 기반 OPC는 에러정정 능력이 전혀 없으므로 에러정정능력을 부여하기 위해, 추가적인 이진 분류기를 도입함으로써 몇 가지 코딩방법론을 제안한다.

4.1 새로운 분할 방법

트리기반분할: 전체 클래스를 두 개의 군집으로 나누면서 트리를 형성해 나간다. 부모 노드에 속한 클래스들을 자식 노드에 배분할 때, 클래스 번호에 따라 정렬한 후 기준열을 따라 엇갈리게 배분한다. 트리 생성 후 루트 노드를 제외한 각 노드는 분할행렬의 한 행에 대응시킨다. 한 행에 대응되는 이진 문제는 해당 노드에 속한 클래스는 양의 클래스로, 이웃(사촌) 노드에 속한 클래스는 음의 클래스로 구성한다. 리프 노드는 OPC에 해당되고, 루트 노드는 전체 클래스의 반을 양의 클래스로, 나머지 반을 음의 클래스로 설정한다. 이런 식으로 하면, 클래스의 수가 K 인 경우, $2 \times (K-1)$ 개의 이진 문제가 생성된다. 그림 4에 클래스의 수가 8개인 경우를 보인다.

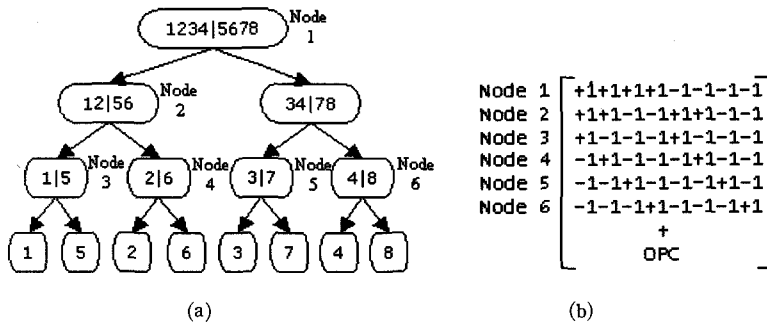


그림 4 8개 클래스에 대한 트리 기반 분할기법, (b)는 생성되는 분할행렬

본 방법은 요구되는 이진 분류기의 수를 대폭 줄일 수 있는 장점이 있다. 그러나, 트리의 깊이에 따라, 양의 클래스로 구분되는 클래스의 수와 음의 클래스로 구분되는 클래스의 수가 다르게 되고, 이에 따라 생성되는 이진 문제의 복잡도도 달라지게 된다. 따라서, 각 이진 분류기의 신뢰도 역시 차이가 날 것이며 디코딩 과정에서 이를 고려해야 할 것이다.

***N*-Shift 분할:** 본 기법에서는 양의 클래스에 할당할 클래스의 개수 N 을 먼저 설정한다. 클래스 번호에 따라 차례대로 N 개의 클래스를 양의 클래스로 할당하고 나머지 클래스를 음의 클래스로 할당하여 하나의 이진 문제를 구성한다. 나머지 이진 문제는 현재 만들어진 이진 문제를 기반으로 우이동(right-shifting)을 시켜 구성한다. 그리고 마지막으로 OPC에서 생성하는 이진 문제를 추가한다. 이런 식으로 하면, 클래스의 수가 K 인 경우, $2 \times K$ 개의 이진 문제가 생성된다. 그림 5는 K 가 4인 경우에, N 을 각각 2, 3으로 했을 때 생성되는 분할행렬을 보여준다.

$$\begin{array}{c} \begin{bmatrix} +1 & +1 & -1 & -1 \\ -1 & +1 & +1 & -1 \\ -1 & -1 & +1 & +1 \\ +1 & -1 & -1 & +1 \end{bmatrix} \\ + \\ \text{OPC} \end{array} \quad \begin{array}{c} \begin{bmatrix} +1 & +1 & +1 & -1 \\ -1 & +1 & +1 & +1 \\ +1 & -1 & +1 & +1 \\ +1 & -1 & -1 & +1 \end{bmatrix} \\ + \\ \text{OPC} \end{array}$$

(a) $N=2$ (b) $N=3$

그림 5 4개의 클래스에 대한 2-Shift 및 3-Shift 분할 행렬

4.2 새로운 디코딩 방법

서로 다른 수준의 난이도를 가진 이진 문제를 학습한 이진 분류기의 출력을 동일한 수준에서 취합하여 최종 결정을 내리는 것은 바람직하지 않다. 이를 보완하는 방법으로, 하나는 이질적인 수준의 출력 자체를 최종 클래스 결정을 위한 정보로 활용하는 방안이 있고, 또 다른 하나는 이질적인 수준의 출력에 적절한 가중치를 부여하는 방안이 있을 수 있다. 전자를 위한 상대거리디코딩 기법과 후자를 위한 가중치기반디코딩 기법을 제안한다.

상대거리디코딩(Relative Distance Decoding): 특정 클래스에 속한 입력에 대하여 서로 다른 클래스들로 학습된 이진 분류기들의 출력은 서로 다를 가능성이 큰데, 우리는 이것을 *습관적출력*이라고 정의한다. 예를 들어, 어떤 분류기가 클래스 i 에 속한 샘플에 대해 습관적으로 0.8을 출력하고, 클래스 j 에 대해서는 0.5를 출력할 수 있다는 것이다. 이러한 특성은 해당 분류기를 학습과정에서 형성된다고 볼 수 있다. 따라서, 각각의 이진 분류기는 주어진 클래스 샘플에 대해 서로 다른 출

력 범위를 가지므로, 같은 출력 값이라도 다르게 해석해야 한다. 분류기 마다 습관적출력이 다른 점은 오히려 클래스를 구분하는 정보로 활용할 수 있다. 본 논문에서는 이를 활용하기 위해 *평균템플릿*을 도입한다. 평균템플릿은 학습 샘플에 대하여 각 이진 분류기의 출력 평균을 계산하는 식 (10)을 통해 도출한다.

$$D'(i, j) = \left(\sum_{\mathbf{x} \in C_j} f_i(\mathbf{x}) \right) / |C_j| \quad (10)$$

여기서, $|C_j|$ 는 클래스 j 에 속한 샘플의 수를 나타낸다. 최종 상대거리디코딩은 식 (11)과 같이 정의한다.

$$\begin{aligned} rd(f(x), D'(\cdot, k)) &= 1 / (1 + \exp(Ad + B)) \\ d &= \sum_l \|f_l(x) - D'(l, k)\|_2 \end{aligned} \quad (11)$$

여기서, A 와 B 는 상수로 실험을 통해서 결정한다.

가중치기반디코딩(Weighted Decoding): 이진 문제를 구성할 때, 양의 클래스로 분류되는 클래스의 수가 (전체 클래스의 반 이하이면서) 커질수록 이진 문제의 복잡도는 올라갈 소지가 크다. 복잡도가 높은 이진 문제에 대한 이진 분류기의 출력은 그렇지 않은 분류기에 비해 출력한 값에 대한 신뢰도가 낮을 수 있다. 디코딩 과정에서 이를 고려하고자 한다. 기반 분류기 l 에 대한 가중치 w_l 는 다음과 같이 계산한다.

$$\begin{aligned} w_l &= 1 / \sum_k L(D(l, k)) \\ L(D(l, k)) &= \begin{cases} 1 & \text{if } D(l, k) > 0 \\ 0 & \text{else} \end{cases} \end{aligned} \quad (12)$$

여기서 $L(D(l, k))$ 은 어떤 클래스가 양의 클래스에 속해 있는지를 검사한다. *가중치기반디코딩*은 다음과 같이 정의한다.

$$s(f(x), D(\cdot, k)) = \sum_l w_l f_l(x) D(l, k) \quad (13)$$

본 디코딩 기법은 분할기법의 복잡도를 판단하는데 사용될 수도 있다. 만약 본 디코딩 기법을 통해 어떤 분할기법의 성능향상이 이루어졌다면, 그 분할기법은 복잡한 것으로 판단할 수 있다.

4.3 직관적 문제 복잡도(Intuitive Problem Complexity)

본 절에서는 이진 분류 문제의 복잡도를 추정하기 위한 직관적인 수치를 정의하여, 새로운 분할기법 설계 시 하나의 기준이 될 수 있도록 한다. 하나의 이진 분류기를 학습하기 위해 여러 개의 클래스로 구성된 새로운 두 개의 클래스를 *슈퍼클래스*라 하자. 슈퍼클래스의 크기는 얼마나 많은 클래스가 하나의 그룹으로 모아졌는지를 나타낸다. 직관적으로, 하나의 그룹에 모이는 클래스의 수가 늘어날수록, 즉 양의 슈퍼클래스 크기 $|C_l^+|$ 또는 음의 슈퍼클래스 크기 $|C_l^-|$ 가 커질수록, 이와 관

련된 이진 분류 문제의 복잡도는 커진다고 볼 수 있다. 이러한 관점에서, 가장 단순한 경우는, 클래스의 개수가 K 인 경우, $|C^+|$ 또는 $|C^-|=1$ 이며, 가장 복잡한 경우는 $|C^+|+|C^-|=K/2 \gg 2$ 인 경우이다. 직관적으로 어떤 이진 분류 문제의 복잡도를 $|C^+|$ 와 $|C^-|$ 에 관하여, 다음과 같은 직관적문제복잡도(IPC)를 정의하자.

$$IPC \equiv \text{Min}(|C^+|, |C^-|) \quad (14)$$

클래스의 수가 K 인 경우, OPC에서 양의 슈퍼클래스 크기는 1이고, 음의 슈퍼클래스 크기는 $K-1$ 이다. All-Pairs에서는 각 이진 분류기가 하나의 양 클래스와 하나의 음 클래스를 구분하도록 학습되므로, 각 슈퍼클래스의 크기 역시 1이다. CC는 두 개의 양 클래스와 나머지 클래스를 구분하도록 학습되므로, 양의 슈퍼클래스 크기는 2이고, 음의 슈퍼클래스 크기는 $K-2$ 이다. 제안한 트리기반에서는 각 이진 분류기가 서로 다른 크기의 슈퍼클래스를 가진다. 루트 노드가 가장 복잡한 경우로, 각 슈퍼클래스의 크기가 반반인 경우이다. 마지막으로, N -Shift에서는 양의 슈퍼클래스 크기는 N 이고, 음의 슈퍼클래스 크기는 $K-N$ 이다. 표 3에 각 분할기법의 슈퍼클래스 크기 및 직관적문제복잡도 IPC를 정리하였다.

표 3 슈퍼클래스의 크기 및 IPC

분할기법	OPC	All-Pairs	CC	Tree-Based	N-Shift
$ C^+ $	1	1	2	$K/2$	N
$ C^- $	$K-1$	1	$K-2$	$K/2$	$K-N$
IPC	1	1	2	$K/2$	N

표 3에 따르면, 트리기반 기법이 다른 분할기법에 비해 가장 복잡한 문제이며, 두 번째 복잡한 경우는 N 값에 따라 N -Shift 또는 CC가 된다.

5. 실험결과

5.1 데이터 셋

본 실험에서는 ORL 얼굴데이터(<http://www.uk.research.att.com/facedatabase.html>), Yale 얼굴데이터(<http://cvc.yale.edu/projects/yalefaces/yalefaces.html>) 셋을 사용한다. ORL 및 Yale 데이터 셋은 얼굴인식에서 널리 사용되는 데이터 셋이다. ORL 데이터 셋은 400장의 얼굴 영상으로, 40명에 대해 10장씩의 영상으로 구성되어 있다. 한 사람에 대한 10장의 영상은 조명, 표정, 포즈 등 매우 다양하게 구성되어 있으며, Yale 데이터 셋은 15명에 대해 다양한 표정 및 조명으로 한 사람당 11장씩

총 165장으로 구성되어 있다.

SVM에 적용할 데이터 셋은 전처리 및 주성분 분석(PCA) 방법에 기반하여 획득하였다. 그림 6은 전처리 후 정규화된 얼굴영상을 보여준다.

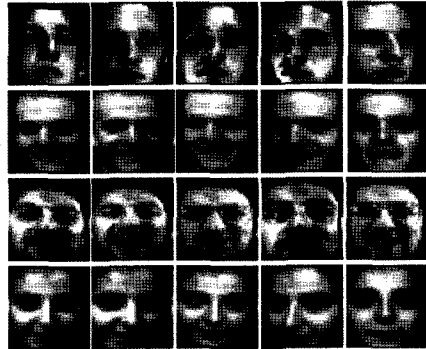


그림 6 ORL 데이터 셋에 대한 정규화된 얼굴영상 일부

주성분 분석을 위해서 모든 데이터를 사용하였고, 변환결과를 두 부분 즉, 참조 셋과 테스트 셋으로 구분하였다. 얼굴영상에 대한 특징벡터의 크기는 얼굴인식 평가 표준 프로토콜인 FERET[38]에서 제시한 PCA기반 인식방법론에 따라 최고 인식률을 보이는 48차원으로 결정하였다. 출력코딩 방법의 특성을 비교하기 위해서 사용한 기반 분류기인 SVM의 구현 알고리즘은 SMO [39]이며 사용한 커널 함수는 RBF이다. 무작위로 한 사람에 대해 5장씩 두 그룹으로 나누어 학습과 테스트 셋을 구성하였고, 이를 5번 수행하여 평균을 내었다. 실험에서 적용한 디코딩 방법을 표 4에 제시하였다.

표 4 다양한 디코딩 기법

기호	의미
HM	해밍 디코딩
MG	마진 디코딩
RD	상대거리 디코딩
WHM	가중치기반 해밍 디코딩
WMG	가중치기반 마진 디코딩

전처리 후, SVM의 입력으로 사용되는 데이터 셋의 특성은 표 5에 정리하였다.

표 5 데이터 셋의 특성

데이터 셋	셋	학습데이터수	테스트데이터수	클래스수	차원
ORL	5	200	200	40	48
Yale	5	90	75	15	48

다음 절에서는 각 분할기법에 따른 인식실험 결과를

제시한다. 실험결과는 SVM의 인자 C 를 1에서 10까지 변화시켜가면서, 그리고 RBF 커널을 사용하는 관계로, 분산 인자는 0.2에서 1.0까지 변화시켜가면서 얻은 결과 중 최고 인식률로 도출하였다.

5.2 특성분석

본 절에서는 대표적인 출력코딩 방법인 OPC, All-Pairs, CC, 그리고 제안한 OPC기반 방법들의 특성을 실험적으로 다음 항목들에 대해 비교·분석 한다.

- 중복학습과 해밍디코딩 성능간의 관계
- 해밍디코딩과 마진디코딩의 성능 분석
- 직관적문제복잡도와 성능간의 관계
- 전반적 성능 분석

중복학습과 해밍디코딩 성능간의 관계: 중복학습은 ECOC에서 에러복원 능력을 위해, 하나의 클래스를 여러 개의 이진 분류기에 중복하여 학습시키는 것을 의미한다. ECOC에서 에러정정능력은 분할행렬의 최소 해밍 거리에 달려있으며, 해밍거리는 중복학습과 다시 연관된다. 클래스의 개수가 40개인 경우(ORL 얼굴영상) 각 출력코딩 방법에서 요구되는 이진 분류기의 개수와 최소 해밍거리를 표 6에 정리하였다. 여기서, 분할행렬의 0 요소와 관련된 해밍거리는 0.5로 가정하였다.

그림 7은 각 분할방법에 대해 해밍디코딩을 적용하였을 때의 인식성능이다. 인식성능과 표 5를 대조해 보면, 인식성능이 이진 분류기의 수와 최소해밍거리에 비례함을 확인할 수 있다.

OPC의 인식성능은 동일한 해밍거리를 가짐에도 불구하고

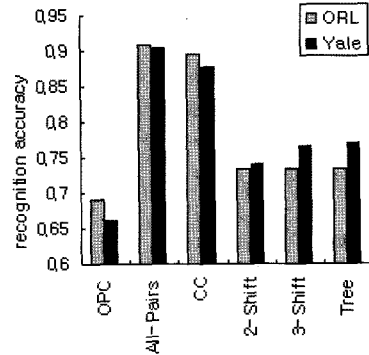


그림 7 해밍디코딩 시의 인식성능 비교

하고, N -Shift와 트리기반 기법에 비해 상당히 떨어져 있다. 이러한 관찰결과는 OPC가 중복학습을 하지 않아, 에러정정 능력이 전혀 없기 때문이라고 볼 수 있다. 즉, N -Shift 및 트리기반 기법이 OPC보다 상대적으로 높은 성능을 보이는 이유는, OPC가 가지는 이진 분류기에 몇 개의 추가적인 이진 분류기를 더 보유함으로써 중복학습이 이루어졌기 때문이다. 따라서, 해밍디코딩을 사용하는 경우, 분할기법의 성능은 최소해밍거리 뿐만 아니라, 중복학습의 양, 즉 이진 분류기의 개수에도 의존한다고 결론지을 수 있다.

해밍디코딩과 마진디코딩의 성능 분석: 그림 8에 따르면, All-Pairs를 제외한 모든 분할기법에 대해 마진디코딩이 해밍디코딩보다 우수함을 알 수 있다. 이것은 마진디코딩이 이진 분류기의 개수 또는 최소해밍거리에는

표 6 분할기법에 따른 요구 이진분류기 수 및 최소해밍거리

분할기법	이진분류기 수		최소해밍거리	
	$K=40$	K -Class	$K=40$	K -Class
OPC	40	K	2	2
All-Pairs	780	$K(K-1)/2$	390	$(K(K-1)/2 - 1)/2 + 1$
CC	780	$K(K-1)/2$	76	2 ($K-2$)
N -Shift	80	$K+K$	2	2
Tree-Based	78	$(K-2)+K$	2	2

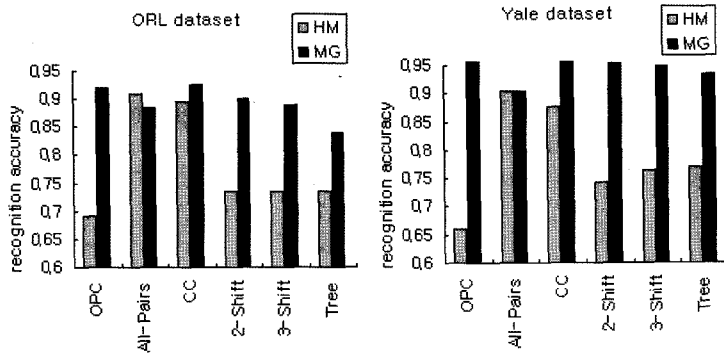


그림 8 해밍디코딩 및 마진디코딩에 따른 인식성능 비교

크게 의존하지 않음을 의미한다.

마진디코딩 적용 시 All-Pairs만이 오히려 성능이 저하된 이유는 다음 두 가지 관점에서 설명할 수 있다. 첫째, All-Pairs의 이진 분류기를 학습하기 위한 샘플의 수가 OPC의 이진 분류기보다 매우 작기 때문에, 특히 얼굴영상의 경우 데이터 수집의 어려움 때문에 이러한 경향이 심한데, 이진 분류기를 충분히 학습시키기에 학습 샘플이 부족하였다고 볼 수 있다. 둘째, 분할행렬이 0 요소를 가지고 있고, 이는 어떤 클래스는 학습에 관여하지 않는다는 것을 의미하고, 결국 넘센스출력의문제를 유발한다. 넘센스출력문제의 정도는 클래스의 수가 늘어날수록 심각해진다.

직관적문제복잡도와 성능간의 관계: 마진디코딩에서는 해밍디코딩 때와는 달리 N-Shift 및 트리기반 기법이 중복학습을 했음에도 불구하고, OPC보다 성능이 저하되었음을 확인할 수 있다. 이러한 이유를 설명하기 위해, IPC와 가중치기반디코딩 방법을 관찰해 본다.

표 3의 IPC 값에 따라 기법을 정렬해 보면, OPC=1, 2-Shift=2, 3-Shift=3, 그리고 트리기반=20 순이다. 이 순서는 그림 9의 마진디코딩에 따른 인식성능과 정확히 대응된다. 정리해보면, 성능은 해밍디코딩 사용 시에는 중복학습에 매우 큰 영향을 받고, 반면 마진디코딩 사용 시에는 중복학습에는 영향을 덜 받으면서 오히려 문제 복잡도에 더 의존한다고 추정할 수 있다. 표 6에 ORL 얼굴 데이터 셋에 대한 IPC 및 인식성능을 비교하였다.

이러한 추정결과에 힘을 실어주기 위해, N-Shift과 트리기반 기법에 대해 마진디코딩과 가중치기반디코딩 방법의 성능을 그림 9에서 비교해 보았다. 표 6과 그림 9에 따르면, 마진디코딩 적용 시 IPC가 증가함에 따라 인식성능이 떨어지는 것을 볼 수 있다. 그러나, 제안한 가중치기반마진디코딩 기법 적용 시 IPC와는 관계없이 성능이 일정함을 보인다. 이것은 제안가중치기반마진디코딩 기법이 문제복잡도에 관련된 어떤 요인을 제거해 주기 때문으로 해석할 수 있으며, 이는 다시 마진디코딩 사용 시 각 분할기법의 성능은 IPC와 관련 있음을 반증한다고 볼 수 있다.

WMG와 MG의 성능비교를 통해 간접적으로, IPC의 중요성을 보여주었다. SVM을 기반 분류기로 사용하는 경우, 보다 정확한 복잡도 평가는 마진의 크기로 측정될 수 있다. 표 7은 분할기법에 따른 마진 크기를 보여준다.

표 8에서 직관적으로(즉, IPC관점에서 볼 때) CC보다 단순한 OPC 및 All-Pairs의 마진이 두 배 가까이 큰

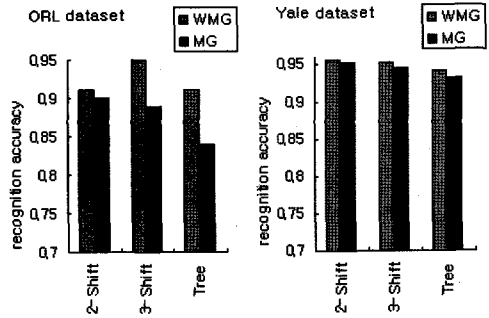


그림 9 마진디코딩 및 가중치기반마진디코딩 적용시의 성능 비교

표 8 분할기법에 따른 마진의 역수 크기 비교

분할기법	OPC	All-Pairs	CC
ORL	15	8	35
Yale	19	11	38

것을 확인할 수 있다.

전반적 성능 분석: 표 9는 ORL 데이터 셋에 대한 다양한 분할기법 및 디코딩 기법 적용에 따른 인식을 비교결과를 보여준다.

OPC 및 All-Pairs에서는 디코딩 방법에 따라 성능우위가 바뀐다. 해밍디코딩 적용 시에는 All-Pairs가 마진디코딩 적용 시에는 OPC가 성능우위를 보인다. 전체적으로는 마진디코딩 기반 OPC가 그 차이는 미미하지만 다소 우수하다. 얼굴인식과 같이 학습 샘플의 수가 작은 경우에는 하나의 분류기가 모든 샘플을 학습하는 OPC가 유리하기 때문인 것으로 보인다. OPC가 비교적 복잡한 이진 문제를 생성함에도 불구하고 비교적 좋은 성능을 보이는 것은, 기반 분류기로 사용된 SVM이 이를 충분히 다룰 수 있기 때문이라고 할 수 있다.

하나의 분류기가 모든 클래스의 샘플을 학습하는, OPC와 CC의 경우를 보면, 해밍디코딩 적용 시 CC가 확실히 월등한 성능을 보인다. 이는 해밍거리에 기반하는 에러정정능력 때문으로 볼 수 있다. 그러나, 마진디코딩 적용 시 이러한 성능 격차는 사라진다. 정리해보면, 학습 샘플의 수가 적은 경우는, 하나의 분류기가 모든 클래스의 샘플을 학습하는 OPC같은 기법이 유리하지만, 그렇다고 에러정정능력을 향상시키기 위해, 대표적 ECOC 기법인 CC와 같이 너무 많은 중복학습은 불필요하다.

표 7 마진디코딩 적용 시 인식성능(%)과 IPC 비교

분할기법	OPC	2-Shift	3-Shift	트리기반
ORL dataset (IPC)	92.0 (1)	90.0 (2)	89.0 (3)	84.0 (20)
Yale dataset (IPC)	66.1 (1)	95.2 (2)	94.6 (3)	94.1 (7)

표 9 인식률(%) 비교

데이터 셋	분할기법	디코딩기법				
		HM	MG	RD	WHM	WMG
ORL dataset	OPC	69.0	92.0	93.0	-	-
	All-Pairs	91.0	88.5	88.5	-	-
	CC	89.5	92.5	93.0	-	-
	2-Shift	73.5	90.0	93.0	73.5	91.0
	3-Shift	73.5	89.0	90.0	71.5	92.5
	Tree	73.5	84.0	85.5	75.0	91.0
Yale dataset	OPC	66.1	96.0	95.5	-	-
	All-Pairs	90.4	90.4	90.4	-	-
	CC	87.7	96.3	95.7	-	-
	2-Shift	74.1	95.2	94.9	74.1	95.7
	3-Shift	76.5	94.6	93.9	76.5	95.4
	Tree	77.0	93.3	92.8	78.7	94.1

6. 결론

본 논문에서는 대표적인 출력코딩 방법론인 OPC, All-Paris, CC 그리고 OPC기반 제안 방법론에 대하여 실험적으로 특성을 비교·분석하였다. 실험을 통해 다음과 같은 사항을 관찰하였다. 첫째, 해밍디코딩 적용 시 인식성능은 중복학습 및 최소해밍거리에 의존한다. 둘째, 마진디코딩은 All-Pairs을 제외한 우리가 적용한 나머지 기법에서 해밍디코딩 보다 우수한 성능을 보였으며, 이로부터 마진디코딩은 중복학습이나 최소해밍거리에 덜 의존적이라고 판단할 수 있다. 중복학습은 해밍디코딩 적용 시에는 성능에 강한 영향을 주지만, 마진디코딩 적용 시에는 그렇지 못하고, 성능은 오히려 문제복잡도에 더 영향을 받는다.

결론적으로, SVM과 같이 마진디코딩이 가능한 경우, 성능은 분할행렬의 최소해밍거리 보다 문제복잡도에 더 의존적이고, 이에 따라서 중복학습을 요구하는 즉 더 많은 이진 분류기를 요구하는 전통적인 에러정정능력 개념으로부터 자유로워져야 한다고 본다. 이러한 관점에서 전통적인 OPC 방법은 여전히 이진 분류기를 다중클래스 문제에 적용하기 위한 적절한 선택이 될 수 있다. 이러한 견지는 최근 발표된 Rifken[40] 연구에도 나타난다.

후속 연구는 클래스의 수가 많은 경우에도 효과적으로 SVM을 적용할 수 있도록, 문제복잡도를 최소화하면서도 이진 분류기의 수를 대폭 줄일 수 있는 트리기반 기법으로 방향을 잡고 있다.

참고 문헌

[1] N. Cristianini and J. Shawe-Taylor, "An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods," Cambridge University Press, 2000.
 [2] V. Vapnik, "An Overview of Statistical Learning

Theory," IEEE Trans. On Neural Networks, Vol. 10, No. 5, pp. 988-999, 1999.
 [3] V. Vapnik, Statistical Learning Theory, John Wiley & Sons, New York, 1998.
 [4] <http://www.clopinet.com/isabelle/Projects/SVM/applist.html>.
 [5] J. Weston and C. Watkins, "Multi-class support vector machines," Proc. of ESANN99, 1999.
 [6] C. Hsu and C. Lin, "A comparison of methods for multiclass support vector machines," IEEE Trans. on Neural Networks, Vol. 13, No. 2, pp. 415-425, 2002.
 [7] E. Allwein, R. Schapire and Y. Singer, "Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers," Journal of Machine Learning Research, Vol. 1, pp. 113-141, 2000.
 [8] G. Guo, S. Li, and K. Chan, "Face Recognition by Support Vector Machines," Proc. of IEEE Int'l Conf. on Automatic Face and Gesture Recognition," pp. 196-201, 2000.
 [9] M. Turk and A. Pentland, "Eigenfaces for Recognition," Cognitive Neuroscience, Vol. 3, pp. 71-86, 1991.
 [10] P. Belhumeur, J. Hespanha, and K. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using Class Specific linear projection," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, pp.711-720, 1997.
 [11] M. Bartlett, J. Movellan, and T. Sejnowski, "Face Recognition by Independent Component Analysis," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 13, No. 6, pp. 1450-1464, 2002.
 [12] P. Penev and J. Atick, "Local feature analysis: A general statistical theory for object representation," Neural Systems, Vol. 7, pp. 477-500, 1996.
 [13] D. Lee and H. Seung, "Algorithms for Non-negative Matrix Factorization," Neural Information Processing Systems, pp. 556-562, 2003.
 [14] H. Yu and J. Yang, "A direct LDA algorithm for high-dimensional data-with application to face recognition," Pattern Recognition, Vol. 34, No. 10,

- pp. 2067-2070, 2001.
- [15] B. Scholkopf, A. Smola, and K. Muller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem," Tech. Report 44, Max-Planck-Institute, 1996.
- [16] Y. Li, S. Gong, and H. Liddell, "Constructing structures of facial identities using Kernel Discriminant Analysis," Proc. of Int'l Workshop on Statistical and Computational Theories of Vision, 2001.
- [17] F. Bach and M. Jordan, "Kernel Independent Component Analysis," Journal of Machine Learning Research, Vol. 3, pp. 1-48, 2002.
- [18] J. Lu, K. Plataniotis, and A. Ventetsanopoulos, "Face Recognition Using Kernel Direct Discriminant Analysis Algorithms," IEEE Trans. on Neural Networks, Vol. 14, No. 1, pp. 117-126, 2003.
- [19] C. Bishop, Neural Networks for Pattern Recognition. New York, Oxford, 1995.
- [20] B. Heisele, P. Ho, and T. Poggio, "Face Recognition with Support Vector Machines: Global versus Component-based Approach," 8th IEEE Int'l Conf. on Computer Vision, pp. 688-694, 2001.
- [21] J. Ghosh, "Multiclassifier Systems: Back to the Future," Proc. of the 3rd Int'l Workshop on Multiple Classifier Systems, Lecture Note in Computer Science, Vol. 2364, pp. 1-15, 2002.
- [22] T. Hastie and R. Tibshirani, "Classification by Pairwise Coupling," Advances in Neural Information Processing Systems, Vol. 10, pp. 507-513, MIT Press, 1998; The Annals of Statistics, Vol. 26, No. 1, pp. 451-471, 1998.
- [23] T. Dietterich and G. Bakiri, "Solving Multiclass Learning Problems via Error-Correcting Output Codes," Journal of Artificial Intelligence Research, Vol. 2, pp. 263-286, 1995.
- [24] C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," Data Mining and Knowledge Discovery, Vol. 2, No. 2, pp. 121-167, 1998.
- [25] L. Hansen, and P. Salamon, "Neural network ensembles," IEEE Trans. on Pattern Recognition and Machine Intelligence, Vol. 12, pp. 993-1001, 1990.
- [26] F. Masulli, and G. Valentini, "Comparing Decomposition Methods for Classification," Proc. of Int'l Conf. Knowledge-based Intelligent Engineering Systems & Allied Technologies, Vol. 2, pp. 788-791, 2000.
- [27] A. Klautau, N. Jevtic, and A. Orlisky, "Combined Binary Classifiers with Applications to Speech Recognition," Proc. of Int'l Conf. on SLP, pp. 2469-2472, 2002.
- [28] J. Ko, "Multiclass Learning with N-Division Output Coding: A Case Study on Face Recognition," PhD Thesis, Dept. of Computer Science, Yonsei University, 2002.
- [29] M. Moreira, E. Mayoraz, "Improved Pairwise Coupling Classification with Correcting Classifiers," Proc. of European Conf. on Machine Learning, pp. 160-171, 1998.
- [30] J. H. Friedman, "Another Approach to Polychotomous Classification," Technical Report, Department of Statistics, Stanford University, 1996.
- [31] J. Platt and N. Cristianini, "Large margin DAGs for multiclass classification," Advances in Neural Information Processing Systems, Vol. 12, pp. 547-553, MIT Press, 2000.
- [32] A. Berger, "Error-correcting output coding for text classification," Proc. of Int'l Joint Conf. on Artificial Intelligence, 1999.
- [33] R. Ghani, "Using error-correcting codes for text classification," Proc. of the 17th Int'l Conf. on Machine Learning, pp. 303-310, 2000.
- [34] J. Rennie and R. Rifkin, "Improving Multiclass Text Classification with the Support Vector Machine," Tech. Report, AI Memo. 2001-026, AI Lab. MIT, 2001.
- [35] A. Klautau, N. Jevtic, and A. Orlisky, "Combined Binary Classifiers with Applications To Speech Recognition," Proc. of Int'l Conf. on Spoken Language Processing, pp. 2469-2472, 2002.
- [36] D. Aha and R. Bankert, "Cloud classification using error-correcting output codes," Artificial Intelligence Applications: Natural Science, Agriculture, and Environmental Science, Vol. 11, pp. 13-28, 1997.
- [37] J. Kittler, R. Ghaderi, T. Windeatt, and J. Matas, "Face Identification and Verification via ECOC," Proc. of IEEE Int'l Conf. on Computer Vision and Pattern Recognition, pp. 755-760, 2001.
- [38] P. Phillips, H. Moon, S. Rizvi and P. Rauss, "The FERET evaluation methodology for face-recognition algorithms," IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 22, No. 10, pp. 1090-1104, 2000.
- [39] J. Platt, "Sequential minimal optimization: A fast algorithm for training support vector machines," Tech. Report 98-14, Microsoft Research at Redmond, 1998.
- [40] R. Rifkin, and A. Klautau, "In Defense of One-Vs-All Classification," Journal of Machine Learning Research, Vol. 5, pp. 101-141, 2004.



고 재 필

1996년 연세대학교 전산학과 학사. 1998년 연세대학교 컴퓨터과학과 석사. 2004년 연세대학교 컴퓨터과학과 박사. 2004년~현재 금오공과대학교 컴퓨터공학과 교수, 관심분야는 모델기반트랙킹, 패턴 인식, 머신러닝