A genetic-based subspace analysis method for improving Error-Correcting Output Coding

Mohammad Ali Bagheri a,*, Qigang Gao a,1, Sergio Escalera b,c

a Faculty of Computer Science, Dalhousie University, 6050 University Avenue, PO BOX 15000, Halifax, Canada NS B3H 4R2
b Centre de Visio per Computador, Campus UAB, Edifici O, Bellaterra, 08193 Barcelona, Spain
c Department of Mathematics, University of Barcelona, Gran Via de les Corts 585, 08007 Barcelona, Spain

A R T I C L E   I N F O
Article history:
Received 17 March 2012
Received in revised form 27 February 2013
Accepted 16 March 2013
Available online 27 March 2013

Keywords:
Error Correcting Output Codes
Evolutionary computation
Multiclass classification
Feature subspace
Ensemble classification

A B S T R A C T
Two key factors affecting the performance of Error Correcting Output Codes (ECOC) in multiclass classification problems are the independence of binary classifiers and the problem-dependent coding design. In this paper, we propose an evolutionary algorithm-based approach to the design of an application-dependent codematrix in the ECOC framework. The central idea of this work is to design a three-dimensional codematrix, where the third dimension is the feature space of the problem domain. In order to do that, we consider the feature space in the design process of the codematrix with the aim of improving the independence and accuracy of binary classifiers. The proposed method takes advantage of some basic concepts of ensemble classification, such as diversity of classifiers, and also benefits from the evolutionary approach for optimizing the three-dimensional codematrix, taking into account the problem domain. We provide a set of experimental results using a set of benchmark datasets from the UCI Machine Learning Repository, as well as two real multiclass Computer Vision problems. Both sets of experiments are conducted using two different base learners: Neural Networks and Decision Trees. The results show that the proposed method increases the classification accuracy in comparison with the state-of-the-art ECOC coding techniques.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

A common task in many real-world pattern recognition problems is to discriminate among instances that belong to multiple classes, known as multiclass classification. There are two general approaches to deal with multiclass problems. One approach is to construct a single decision function by considering all classes concurrently and to solve a complex classification problem, known as the single-machine approach [1,2]. Some classification algorithms, such as the k-Nearest Neighbor (kNN) or Multilayer Perceptron (MLP), are inherently based on this approach. The second approach is to recast the multiclass problem into a series of smaller binary classification problems, which is referred to as “class binarization” [3]. In this way, two-class problems can be solved by binary classifiers and the results can then be combined so as to provide a solution to the original multiclass problem. An extensive comparison of the results demonstrates that the class binarization approach generally achieves a better performance, even for powerful learners [3,4]. In addition, many established classification algorithms are specifically designed for binary problems, such as Support Vector Machine (SVM) or AdaBoost. Therefore, to solve multiclass classification problems using these binary classifiers, the class binarization approach should be employed.

Among the proposed methods for approaching class binarization, three techniques are well-known: one-versus-all (OVA) [5], one-versus-one (OVO) [6], and Error Correcting Output Codes [7,8]. In one-versus-all, the multiclass problem is decomposed into several binary problems in the following way: for each class a binary classifier is trained to discriminate among the patterns of the class and the patterns of the remaining classes. In the one-versus-one technique, one classifier is trained to separate each possible pair of classes. In both approaches, the final classification prediction is usually obtained by means of a voting or committee procedure. More recently, a unified framework was introduced to decompose a multiclass problem into a series of different binary problems, which is known as Error Correcting Output Codes (ECOC). In this framework, each classifier is trained on a two-meta-class problem, where each meta-class consists of some combinations of the original classes. The ECOC method can be broken down into two stages: encoding and decoding. The aim of the encoding stage is to design a discrete decomposition matrix (codematrix) for the given problem. Each row of the codematrix,
named codeword, is a sequence of bits representing each class, where each bit identifies the membership of the class to a classifier [9]. In the decoding stage, the final classification decision is obtained based on the outputs of binary classifiers. Given an unlabeled test sample, each binary classifier casts a vote for one of the two meta-classes used in its training. The output vector is compared to each class codeword of the matrix and the test sample is assigned to the class whose codeword is closest to the output vector, according to a distance measure. Because of the ability of the ECOC framework to correct the bias and variance errors of the base classifiers [10–12], it has been successfully applied to a wide range of applications [13–16].

The priority when designing ECOC matrices is to improve the error correcting capability of the codematrix, mainly by maximizing a separability criterion between any pair of rows and/or any pair of columns. In general, optimizing row separation criteria directly leads to more error-correcting capability. According to error-correcting theory, it can easily be shown that a matrix having d bits error-correcting capability implies that there is a minimum Hamming distance of 2d+1 between any pair of rows (codewords). Assuming that each codetest is transmitted independently, it is then possible to correctly classify a received test codeword having fewer than d bits in error, by assigning that codeword to the closest codeword based on the Hamming distance. Therefore, it is desirable to design a codematrix with a high minimum Hamming distance between any pair of codewords. However, the capability to detect and possibly correct errors is “dependent on the assumption that each error is independently produced” [17,12]. Therefore, the independence of binary classifiers is the cornerstone of the design of ECOC matrices, without which the ECOC method would be ineffective. The intuition is that if each binary classifier makes different errors, then the ECOCs ability to detect and possibly correct errors would be improved.

The conventional strategy in the ECOC literature for designing independent classifiers is to optimize the distance between ECOC dichotomizers. This property is generally achieved by maximizing the Hamming distance between each column and the others, including their complementaries. Several methods have been proposed that aim to simultaneously optimize row and column separation, such as BCH coding [18], CHC coding [19], and evolutionary techniques [20]. Interestingly, the extensive experimental results show that the codes designed using only a row separation criterion almost performed as well as codes designed using column and row separation. However, codes designed using only column separation criteria performed significantly worse [3]. In addition, many researchers agree that a pseudo-random generation of a codematrix is a reasonably good method, and that “more sophisticated methods might have only marginal effect on testing error” [7,21]. These results reveal that conventional strategies to design a codematrix will not promote independence among binary classifiers.

One efficient approach to increase diversity among an ensemble of classifiers is to train each learner with data that consist of different feature subsets, leading to uncorrelated errors by base learners [22]. This idea, usually called subspace approach, can effectively make use of the diversity of base learners to reduce the variance as well as the bias errors [23,24]. Inspired by this idea, we design a new method for the ECOC framework, named Subspace ECOC. The strategy consists of using different feature subsets for each dichotomizer, leading to more independent classifiers and, consequently, increasing the overall system accuracy. In addition to the design of more independent classifiers, the new technique allows for the design of larger codes in comparison to classical methods.

Some previous studies have proposed the use of bagging and boosting within the ECOC framework, mainly by selecting a sampling of data for each dichotomizer in order to increase the diversity of binary problems. In this sense, Schapire proposed a new technique by combining the boosting algorithm with the idea of output codes [21]. Similarly, Windkett and Ardehiri proposed to combine the AdaBoost, a version of boosting, with output coding using the decision tree as a base learner [25]. Although the previous methods have performed sampling of data, they used the same set of available features, so it is likely that some classification errors will be common, arising from noisy or non-discriminant features. To our knowledge, there is no related work that performs feature selection within the ECOC framework independently of the base classifier.

Another relevant factor for a codematrix to achieve a good performance is the problem-dependent design of the codes. That is, for a given problem we need to take into account the characteristics of the problem at hand. While the most previous work tried to design a generic codematrix for any classification problem, few studies tried to develop the codematrix by considering the problem characteristics or the classification performance. In this paper, we attempt to tackle this issue using an evolutionary algorithm-based optimization approach in order to guide the feature selection of the three-dimensional ECOC matrix for each problem. More specifically, the Genetic Algorithm (GA) is employed, which has been shown to provide an efficient trade-off between the quality of the solution and the search complexity. In this way, the efficiency of the whole ensemble for the problem at hand is considered in the optimization process of the Genetic Algorithm. As a result, our problem-dependent coding design in the ECOC framework based on the feature subspace approach not only provides more independent classifiers, but also increases the overall classification accuracy.

The rest of this paper is organized as follows. Section 2 provides a brief introduction to the ECOC framework. The proposed method based on the feature subspace is explained in detail in Section 3. Sections 4 and 5 report the experiments we performed with data from two different environments: benchmark and image vision datasets. Finally, Section 6 draws the main conclusions of the paper.

2. Error Correcting Output Codes

First, we briefly describe some notations used in this paper:

- \( T = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\} \): A training set, where \( x_i \in \mathbb{R}^d \), and each label, \( y_i \), is an integer belonging to \( Y = \{1, 2, \ldots, N_c\} \), where \( N_c \) is the number of classes.
- \( h = [h_1, h_2, \ldots, h_L] \): A set of \( L \) binary classifiers.

2.1. ECOC overview

The basis of the ECOC framework consists of designing a codeword for each of the classes. This method uses a matrix \( M \) of \( (1, -1) \) values of size \( N_c \times L \), where \( L \) is the number of codewords coding each class. This matrix is interpreted as a set of \( L \) binary learning problems, one for each column. That is, each column corresponds to a binary classifier, called dichotomizer \( h_i \), which separates the set of classes into two metaclasses. Instance \( x \), belonging to class \( i \), is a positive instance for the \( j \)th classifier if and only if \( M_{ij} = 1 \) and is a negative instance if and only if \( M_{ij} = -1 \). Table 1 shows a possible binary coding matrix for a 4-class problem \( \{c_1, \ldots, c_4\} \) with respective codewords \( \{\mathbf{M}, \mathbf{r}_j\} \) that uses six dichotomizers \( \{h_1, \ldots, h_6\} \). In this table, each column is associated with a dichotomy classifier, \( h_j \), and each row is a unique codeword that is associated with an individual target class. The
white cells of the table refer to +1 and the dark cells stand for −1. For example, h_3 recognizes two meta-classes: original classes 1 and 4 form the first meta-class, and the other two form the second one.

When testing an unlabeled pattern, x*, each classifier outputs a “−1” or “1”, creating a L long output code vector. This output vector is compared to each codeword in the matrix, and the class whose codeword has the closest distance to the output vector is chosen as the predicted class. The process of merging the outputs of individual binary classifiers is called decoding. The most common decoding method is the Hamming distance. This method looks for the minimum distance between the prediction vector and the codewords:

$$y_H = \arg \min_{r \in \{1, \ldots, N_c\}} \frac{1}{N_c} \sum_{i=1}^{N_c} \frac{1}{2} \left( 1 - \text{sign}(M(r,i) \cdot f_i(x)) \right)$$

where sign(z) is +1 if z > 0, −1 if z < 0 and 0 otherwise. M(r,.) designates the codeword r in the matrix and y_H (i.e., the predicted label. For example, the output [+1−1−1−1+1−1] is closest to c_2 codeword with a Hamming distance of 1, and hence c_2 would be chosen as the predicted label of x*.

Several decoding strategies (combination methods other than distance methods) have been proposed in the literature, such as probabilistic approaches [26,27] and loss-functions strategies [8]. The reader is referred to [28,12] for a more detailed analysis.

2.2. Coding designs

There are several coding designs that can be used for ECOC methods, which can be broadly divided into two main approaches, which we have named static and dynamic coding designs. The static codings are for designing generic code matrices for any classification problem, regardless of the learning algorithm and the problem to which the codematrix is going to be applied. The design of these matrices is usually based on their error correcting capabilities or on maximizing a separability criterion between rows and columns. On the other hand, the dynamic coding methods take into account the characteristics of the problem at hand. This approach can yield higher classification performance for a specific learning algorithm [20] at the cost of higher computational complexity and less generalization of the designed matrices to other problems. In the following subsections, we review the state-of-the-art coding designs based on these two families of approaches.

2.2.1. Static coding designs (problem-independent)

Most of the popular ECOC coding methods fall into the static coding category. In this category, the most well-known coding techniques are dense random coding, consisting of a binary matrix, and sparse random coding, using a third symbol (zero). The zero value in sparse coding means that a given class is not considered in the training phase of a particular classifier. Allwein et al. [8] suggested a length of 10 log_2(N_c) and 15 log_2(N_c) bits per code for dense and sparse coding styles, respectively.

Coding methods in this category are defined independent of the problem domain and seek to satisfy two basic criteria:

- **Row separation:** Each codeword should be as far apart from the other codewords. The standard measure of the error-correcting ability of any codematrix is the minimum Hamming distance between any pair of codewords.
- **Column separation:** In addition to row separation, each dichotomizer, h_i, should be well-separated from the other dichotomizers. This property results in low correlated classifiers in the ensemble.

Taking into account these two criteria, several methods have been proposed in order to optimize the coding design, such as the algebraic-based BCH codes [29], randomized hill climbing [7], simulated annealing and evolutionary computation [30,20]. Kuncheva [31] used the disagreement diversity measure, a criterion from the literature on classifier ensembles, and suggested an evolutionary algorithm for constructing the codematrix.

2.2.2. Dynamic coding methods (problem-dependent)

Utschick and Weichselberger [32] proposed one of the first problem-dependent ECOC designs. In their work, they developed a method based on the application of maximum-likelihood objective function by means of the Expectation–Maximization (EM) algorithm in order to achieve a suboptimal decomposition of the multiclass problem into binary problems. Escalera et al. proposed a new problem-dependent ECOC design based on subclass information in the ECOC framework [33]. Multiclass problems are solved by splitting the original set of classes into subclasses and embedding the binary problems in the ECOC design. Crammer and Singer proposed a method to find an optimal coding matrix by changing its representation from discrete to continuous values [34]. Pujol et al. [35] proposed a heuristic method, named Discriminant ECOC, to build the ECOC matrix based on a hierarchical partition of the class space that maximizes a discriminative criterion. They also proposed ECOC-optimizing node embedding (ECOC-ONE) [36]. This method uses a coding process that trains relevant binary problems guided by a validation set. Their proposed procedure begins with an initial codematrix and aims to recursively optimize the codematrix by minimizing errors in the confusion matrix by using the validation samples. The authors suggested a length of 2Nc bits per code. In [37] a method is proposed to learn the error-correcting output codes from data, where the backpropagation algorithm is used to drive the codewords for each class. In [38], a method is proposed to explore the distribution of data classes and optimize both the decomposition and the number of base learners, named Data-Driven Error Correcting Output Coding (DECOC). DECOC computes the confidence score of each base classifier based on the structural information of the training data. Sorted confidence scores are then used to selectively include some of the binary learners in the codematrix. Zhong et al. proposed a method that learns the ECOC matrix and dichotomizers simultaneously from data by formulating the learning model as a sequence of concave–convex programming problems [39]. Recently, Hatami [40] proposed a heuristic method for an application-dependent design of ECOC matrix based on a thinning algorithm, called Thinned-ECOC. The main idea of the method is to successively remove some unnecessary and redundant columns from the initial codematrix based on a metric defined for each column.

3. Genetic Algorithm-based Subspace ECOC (GA-SS-ECOC)

As we stated earlier, there exist two main factors affecting the performance of ECOC methods. The first is that the error
argue that using the knowledge of the problem domain to learn
on the generation of a coding matrix without considering the
3.2. Improving the Subspace ECOC coding by Genetic Algorithm
since it uses fewer features.
approach is that each binary classi
sparse ECOC, respectively. The other advantage of the subspace
of classes (i.e. ECOC methods is small in problems with a relatively small number
included in the corresponding classi

3.1. Subspace ECOC
The central idea of the proposed Subspace ECOC is based on
using feature space in the design process of the ECOC matrix. That
is, each dichotomizer is trained with a different feature subset,
leading to better classification accuracy. From the design process
point of view, we generate a three-dimensional codematix, where
the third dimension is the feature space of the problem domain. In
order to generate this framework, first, a two-dimensional code-
matrix is created from a previous set of matrices that maximizes
the minimum distances between any pair of codewords. Then, for
each column, a random vector of \([-1, +1]\) values of size \(n\) is
generated, where \(n\) is the number of features. The meaning of ‘+1’
1, (not) included in the corresponding classifier. Note that in both, sparse
and dense coding styles, the value of each cell in the feature space
cannot take the 0 value. The representation of the proposed three-
dimensional codematix is illustrated in Fig. 1.

This approach not only creates more independent classifiers,
but it can also build longer codewords. It can be shown that the
maximum length of codewords in ECOC matrices is \((2^{N \cdot n} - 1)\) and
\((3^{N \cdot n} - 2^{N \cdot n} + 1)/2\) for dense and sparse coding styles, respectively.
Thus, the maximum number of binary classifiers in the classical
ECOC methods is small in problems with a relatively small number
of classes (i.e. \(N_c < 6\)). Conversely, the ECOC method with longer
codes is able to significantly improve the results [8,3]. In our
proposed approach, since each classifier can be trained using a
variety of feature subsets, more diverse classifiers can be built. In
an \(n\)-dimensional feature space, \(2^n - 1\) different non-empty feature
subsets can be selected. So, the number of distinct dichotomizers is
\((2^n - 1) \cdot (2^{N \cdot n} - 1)\) and \((2^n - 1) \cdot (3^{N \cdot n} - 2^{N \cdot n} + 1)/2\) for dense
and sparse ECOC, respectively. The other advantage of the subspace
approach is that each binary classifier requires less training time,
since it uses fewer features.

3.2. Improving the Subspace ECOC coding by Genetic Algorithm
As mentioned earlier, most previous work on ECOC was focused
on the generation of a coding matrix without considering the
characteristics of the problem at hand. Recently, some researchers
argue that using the knowledge of the problem domain to learn
relevant binary problems has a significant effect on ECOC accuracy.
The basic strategy of these studies is to use the training data to
guide the design process, and thus, to develop a coding matrix that
focuses on binary problems that better fit the decision boundaries
of a given problem. This problem-dependent development can be
considered as an optimization design process. One promising
strategy for this optimization issue is to use an evolutionary
algorithm-based approach. One of the most well-known evolu-
tionary approaches is the Genetic Algorithm, which is inspired by
an explicit imitation of biological life, in which weaker units
(individuals with lower fitness) are eliminated and the strongest
(fittest) individuals survive to produce the next generations.

In the proposed GA-based Subspace ECOC method, each chro-
mosome of the population is a three-dimensional codematix.
Consider a problem with \(N_c\) classes, \(L\) dichotomizers, and \(n\) features. Each matrix is encoded by a vector of length \(L(N_c + n)\). The first \(N_c + n\) bits represent the first classifier (the first dichot-
omizer and its corresponding feature subset), followed by the
\(N_c + n\) bits for the second classifier, \(h_2\), and so on. Based on two
ECOC schemes, i.e. dense and sparse coding, two versions of the
GA-based Subspace ECOC are proposed. In the dense scheme, each
bit of a chromosome has a value of \(+1, -1\), whereas in the sparse
scheme, bits related to dichotomizers may take the zero value
as well. A possible chromosome encoding for a problem with
three dichotomizers, three classes, and four features is illustrated
in Fig. 2.

Due to the fact that a chromosome corresponds to a three-
dimensional codematix, there are few changes required in muta-
tion and particularly in cross-over operations of the standard
Genetic Algorithm. Mutation flips the sign of a randomly selected
bit of a chromosome. However, since flipping the sign of bits
corresponding to dichotomizers may result in a non-valid dichot-
omizer, the mutation is only applied to bits corresponding to
feature vectors.\(^2\) This modification ensures that the premature
convergence of the algorithm is avoided. The cross-over operation
is more sophisticated, since a chromosome consists of a sequence
of two entities: a dichotomizer and its corresponding feature
vector, each one having different concepts and clearly different
possible values. The proposed technique is to randomly choose the
cross-over point from the positions that encoded classifiers end.
The cross-over operation is schematically presented in Fig. 3. The
GA-based Subspace ECOC method can be summarized in the
following form:

1. Pick the parameters of the Genetic Algorithm:
   - Population size (\(P\)),
   - Maximum number of iterations,
   - Mutation probability.
2. Generate a random population of chromosomes and calculate
   their fitness values as the classification accuracy of each
   individual (i.e. each ECOC matrix) on the validation data.
3. Perform one-point cross-over and mutation to generate the
   offspring chromosomes.
4. Calculate the fitness values of the offspring chromosomes.
5. Pool offspring and the current population together and select \(P\)
   chromosome with the highest fitness as the next generation.
6. If the stopping criteria are met, finish; else go to step 3.

It is worth mentioning that this special implementation of the
Genetic Algorithm is a kind of hill-climbing strategy, as it guaran-
tees that the fitness value will not decrease in the subsequent
generations.

\(^2\) As a simple example, if we mutate the third bit of [1 1 -1], a non-valid
dichotomizer will be generated.
4. Experimental comparison over benchmark dataset

In this section, we first discuss the experimental settings of the experiments including the data, the comparative methods, and the evaluation measurements. We then provide a detailed comparison of results achieved by different methods.

4.1. Experimental settings

- **Data**: The proposed GA-SS-ECOC method was first validated on 20 multiclass datasets from the UCI machine learning repository [41]. Table 2 shows the number of classes, instances, and features of each UCI dataset.

- **Methods**: We compared our proposed method with classical static methods including OVO, OVA, and dense random and sparse random ECOC methods as well as two state-of-the-art problem-dependent coding methods, including Discriminant ECOC and ECOC-ONE. The class of an instance in the ECOC schemes was chosen using the Exponential Loss-Weighted (ELW) decoding [28]. In order to limit the computational complexity of the experiments in the GA-SS-ECOC design, the population size and maximum number of iterations in the GA optimization algorithm were set to 10 and 3, respectively. In addition, if the best fitness did not change in two successive iterations, the optimization process was stopped. The mutation rate was set to \( p = 0.1 \) in order to promote diversity in the population.

We set the number of different feature subsets for each nontrivial dichotomizer as 10. Thus, codewords are 10 times longer in the Subspace ECOC design for both dense and sparse ECOC methods. In this study, two base learners were chosen: a classification and regression tree (CART) with the Gini-index as a split criterion and a multilayer perceptron (MLP) with 10 hidden nodes and the hyperbolic tangent transfer function. The MLP classifier cannot handle the missing values, so the instances with missing values were removed.

- **Evaluation measurements**: The classification accuracy was obtained by means of 10-fold cross-validation to improve the reliability of the results. In order to have a fair comparison, the training and test sets of all methods were the same for each repetition of experiments. Moreover, using non-parametric tests we showed that the performance of rival methods was statistically different.

4.2. Experimental results

The average accuracy of the rival methods for each dataset is presented in Tables 3 and 4. In these tables, the means of prediction accuracy over 10 runs (expressed in %) are reported for each method on the considered datasets. For each dataset, the best accuracy achieved among all tested methods is in bold. In order to show the superiority of the proposed ECOC method in terms of the classification accuracy, statistical analysis is necessary. According to the recommendations of Demsar [42], we consider the use of non-parametric tests. Non-parametric tests are safer than parametric tests, such as ANOVA and \( t \)-test, since they do not assume normal distribution or homogeneity of variance. In this study, we employed the Iman–Davenport test. If there are statistically significant differences in the classification performance, then we can proceed with the Nemenyi test [43] as a post-hoc test, which is used to compare the methods with each other.
undesirably conservative, and proposed a revised one:

$$F_F = \frac{(N-1) \chi^2_j}{N(k-1)-\chi^2_j},$$

(3)

which is distributed following an $F$ distribution with $k-1$ and $(k-1)(N-1)$ degrees of freedom. By applying this correction we obtained $F_0=46.08$ and 24.39 for CART and MLP, respectively. The critical value of $F(7,19)$ for $\alpha = 0.05$ is 2.54. As the values of $F_0$ are higher than 2.54 we can reject the null hypothesis, that is, the results are not obtained because of randomness.

Further, to compare rival methods with each other, we applied the Nemenyi test. Two methods are significantly different if their corresponding average ranks differ by at least the critical difference value (CD):

$$CD = q_{\alpha} \sqrt{\frac{k(k+1)}{6N}},$$

(4)

In our case, the critical value for a 90% of confidence is $CD = 2.78$ for $N=1400$. The results of the Nemenyi test are illustrated in Fig. 4. In this figure, the mean rank of each method is indicated by a square. The horizontal bar across each square shows the critical difference. Two methods are significantly different if their corresponding average ranks differ by at least the critical difference value. That is, their horizontal bars do not overlap. Looking at the rankings of each coding strategy shown in Fig. 4, we can conclude that the proposed GA-SS-ECOC schemes are significantly better than other strategies in the present experiments.

The results in Tables 3 and 4, along with the statistical tests presented in Fig. 4, indicate that overall, the GA Subspace approach achieves the best performance among all methods. As a general conclusion, the advanced performance of the proposed method does not differ much depending on the base classifier. Using both neural network and decision tree as the base learner, we found significant differences between GA-SS-ECOC and classical ECOC for both dense and sparse schemes. An analysis of the results shows that when the number of training patterns is relatively small compared with the dimensionality of data, the subspace approach is usually a better choice. Ho [22] showed that while most classification approaches suffer from the curse of dimensionality, the subspace approach can take advantage of high dimensionality.

5. Image vision applications

We also applied the proposed GA-SS-ECOC to two machine vision problems: logo recognition and shape categorization. As in the previous experiments with UCI datasets, CART and MLP classifiers were chosen and their adjustable parameters were set as the same as the previous experiments. Again, 10-fold cross-validation method was used for performance evaluation.

5.1. Shape categorization

The first real application in our experiments was shape classification, in which we used the MPEG7 database. This dataset consists of $C=70$ classes (bone, chicken, cellular phone, etc.) with 20 instances per class, which represents a total of 1400 object images. All samples were described using the Blurred Shape Model descriptor [45]. This technique describes each shape by means of 100 features. Thus, the MPEG7 is a 100 dimensional dataset. Fig. 5 shows a few samples for some categories of this dataset.
5.2. Logo recognition

The ECOC approach was then used in the logo recognition problem. The logo images were based on a database of logos which contains pure pictorial logos (e.g. logo 60, Fig. 6), text-like logos (e.g. logo 30, Fig. 6), and text–graphics mixture logos (e.g. logo 10, Fig. 6). The complete dataset contains 105 images and was obtained from the database distributed by the Document Processing Group, Center for Automation Research, University of Maryland [46]. The logos in the dataset have very different sizes; the smallest one is 121 × 145 pixels and the largest one is 802 × 228 pixels.

This dataset provides only a single instance of 105 individual logo classes. In order to increase the number of samples, for each logo class, some artificially degraded images were generated by using the noise models as described in the following subsection.

5.2.1. Noise models

We investigated the robustness of the methods when the logos are corrupted using two different image degradation methods: (1) Gaussian noise (a global degradation as shown in Fig. 7a); and (2) spot noise (a local degradation as shown in Fig. 7b). For each method, we degraded each image in the database, varying the amount of degradation in equally spaced steps. We generated a set of 40 examples for each class of logo images by adding both the Gaussian white noise of mean = [0, 0.1, 0.2, ..., 0.5] and var = [0, 0.01, ..., 0.05] and the spot noise of different sizes (width = [10, 15, ..., 30] pixels).

5.2.2. Feature extraction

Some researchers have studied the problem of logo recognition by applying different feature extraction methods, such as algebraic and differential invariants [47,48], Zernike and pseudo-Zernike moments [49,50], line segment Hausdorff distance [51], and template matching [48]. In this paper, logo images are described in terms of seven invariant moments, which have been proven to be an effective descriptor of logo and trademark images [48]. Consequently, the logo dataset is a seven dimensional dataset.

5.2.3. Moment invariants

These descriptors, also called geometric moment invariants, were first introduced in 1962 by Hu [52] based on the theory of algebraic forms. These moment features have the desirable properties of being invariant under rotation, translation, scale, and reflection of images and have been widely used in many applications due to their invariance properties. For a 2-D image, \( f(x, y) \), the central moment of order \((p+q)\) is defined by

\[
\mu_{pq} = \sum_{x} \sum_{y} (x-X)^p (y-Y)^q f(x, y), \quad p, q = 0, 1, 2, \ldots \tag{5}
\]

where the pixel point \((X, Y)\) is the centroid of the image. Seven moment invariants \((M_1-M_7)\) based on the 2nd and 3rd order moments are defined [52]:

\[
M_1 = \mu_{20} + \mu_{02},
\]

\[
M_2 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2.
\]
\[ M_3 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2. \]

\[ M_4 = (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2. \]

\[ M_5 = (\mu_{30} + \mu_{12})[\mu_{30} - 3\mu_{12}]^2[\mu_{30} + \mu_{12}]^2 - 3(\mu_{21} + \mu_{03})^2] \\
+ (3\mu_{21} - \mu_{03})[\mu_{21} + \mu_{03}]^2[\mu_{21} + \mu_{03}]^2 - (\mu_{21} + \mu_{03})^2]. \]

\[ M_6 = (\mu_{20} - \mu_{02})[\mu_{30} + \mu_{12}]^2 - (\mu_{21} + \mu_{03})^2] \\
+ 4\mu_{11}(\mu_{30} + \mu_{12})[\mu_{21} + \mu_{03}] \\
M_7 = (3\mu_{21} - \mu_{03})[\mu_{30} + \mu_{12}]^2[\mu_{30} + \mu_{12}]^2 - 3(\mu_{21} + \mu_{03})^2] \\
- (\mu_{30} - 3\mu_{12})[\mu_{21} + \mu_{03}]^2[\mu_{30} + \mu_{12}]^2 - (\mu_{21} + \mu_{03})^2]. \]
different subsets of features can be activated for a given dichotomy. The proposed method takes advantage of some basic concepts of ensemble classification, such as diversity of classifiers, and also benefits from the evolutionary algorithm-based approach to optimize the three-dimensional codematrix, taking into account the characteristics of data. As a result, we obtain a problem-dependent coding design with more independent classifiers, which reduces the bias and variance errors of the multiclass problem and, consequently, increases the discrimination power of the ensemble. The method was evaluated with several UCI datasets and two challenging computer vision problems using two different base learners. The proposed method shows significant performance improvements compared to the state-of-the-art approaches.

Conflict of interest statement

None declared.

Acknowledgments

The authors would like to thank Ms. Fatemeh Yazdanpanah for her kind support in conducting experiments.

References

Mohammad Ali Bagheri

Bagheri was born in Shiraz, Iran, in 1983. He received his B.S. degree in Industrial Engineering from Sharif University of Technology in 2005 and his M.S. degree in Information Technology from Tarbiat Modares University, Tehran, Iran, in 2008. He is currently a research assistant at Dalhousie University, Canada. His research interests include pattern recognition, machine learning with special interest in machine olfaction, image analysis, and evolutionary computation.

QiQiang Gao

is currently a professor of Computer Science at Dalhousie University, Canada (1993–). Gao got his undergraduate education in Xi’an Jiao Tong University, China (1973–1976), and received both M.A.Sc. and Ph.D. degrees from the University of Waterloo, Canada (1988 and 1993, respectively). His research interests include perceptual knowledge based visual pattern recognition, data mining and data warehouses, and web-based intelligent information systems. He has supervised/co-supervised 44 graduate students at both Ph.D. and Masters levels, and contributed to 75 research articles (authored/co-authored) in the related areas.

Sergio Escalera

received the B.S. and M.S. degrees from the Universitat Autònoma de Barcelona (UAB), Barcelona, Spain, in 2003 and 2005, respectively. He obtained the Ph. D. degree on Multi-class visual categorization systems at Computer Vision Center, UAB. He obtained the 2008 best Thesis award on informatics at Universitat Autònoma de Barcelona. He is a lecturer of the Department of Applied Mathematics and Analysis, Universitat de Barcelona. He is a partial time professor at Universitat Oberta de Catalunya. He is the member of the Computer Vision Center (UAB). He is a member of the Perceptual Computing Group and a consolidated research group of Catalonia. He is also a member of the International Foundation of Research & Analysis. He is an editorial board member of Journal of Convergence Section C: Web and Multimedia, WebmedCentral, and American Journal of Intelligent Systems. His research interests include, between others, machine learning, statistical pattern recognition, visual object recognition, and human computer interaction systems, with special interest in human pose recovery and behavior analysis.