A Unified Feature and Instance Selection Framework Using Optimum Experimental Design

Lijun Zhang, *Student Member, IEEE*, Chun Chen, *Member, IEEE*, Jiajun Bu, *Member, IEEE*, and Xiaofei He, *Senior Member, IEEE*

***Abstract*—The goal of feature selection is to identify the most informative features for compact representation, whereas the goal of active learning is to select the most informative instances for prediction. Previous studies separately address these two problems, despite of the fact that selecting features and instances are dual operations over a data matrix. In this paper, we consider the novel problem of simultaneously selecting the most informative features and instances and develop a solution from the perspective of optimum experimental design. That is, by using the selected features as the new representation and the selected instances as training data, the variance of the parameter estimate of a learning function can be minimized. Specifically, we propose a novel approach, which is called Unified criterion for Feature and Instance selection (UFI), to simultaneously identify the most informative features and instances that minimize the trace of the parameter covariance matrix. A greedy algorithm is introduced to efficiently solve the optimization problem. Experimental results on two benchmark data sets demonstrate the effectiveness of our proposed method.**

***Index Terms*—Active learning, experimental design, feature selection, instance selection.**

I. Introduction

In MANY image processing applications, such as visual recognition and image retrieval, there is usually large amounts of data with high dimensionality. High-dimensional data sets not only consume more storage and computation resources but also degrade the performance of learning algorithms, which is typically referred to as the curse of dimensionality [1]. Feature selection addresses this issue by selecting a subset of features to reduce the dimensionality. Various studies have shown that a large amount of features can be removed without performance deterioration [2]–[9]. In particular, feature selection has been successfully applied to SAR image classification [10], object categorization [11], and codeword selection [12], [13].

A dual problem of feature selection is active learning, which selects the most informative instances for prediction. In statistics, the problem of selecting instances to label is typically referred to as experimental design [14]. Active learning is well motivated in many modern machine learning problems, where unlabeled data are abundant but labels are expensive. Instead of being a passive recipient of data to be processed, the active learner queries the labels of the most informative data instances and use them as its training data [15]. We expect that the active learner can achieve high accuracy with as few labeled instances as possible [16], which is verified by the recent studies in content-based image retrieval [17]–[19] and in face recognition [20], [21]. Existing active learning algorithms can be categorized as either label independent or label dependent [22]. We focus on the former one in this pape, and use instance selection to emphasize this difference.

In general, data are represented by a matrix where one dimension denotes feature and the other denotes instance. Thus, feature selection and instance selection are essential dual operations over the data matrix. As a result, simultaneously performing feature selection and instance selection can potentially make use of the duality between feature space and instance space. Notice that the same idea has been adopted in coclustering, where features and instances are simultaneously clustered [23]. From a practical viewpoint, it is also necessary to consider these two operations simultaneously. Since our data usually contain noise, if we separately perform the two operations, outliers may affect feature selection, and irrelevant features may mislead instance selection. For example, maximum variance is a feature selection method that prefers features with large variance. It is sensitive to noise since the calculation of variance can be significantly affected by outliers.

In this paper, we consider the novel problem of simultaneously selecting the most informative features and instances from the data. Inspired from the techniques of optimum experimental design (OED) [14], the most informative features and instances are defined to be those minimizing the size of the parameter covariance matrix of a learning function. In statistics, there are many different optimality criteria to measure the size of the covariance matrix. Here, we adopt the A-optimality [14]. Specifically, we propose a novel approach called Unified criterion for Feature and Instance selection (UFI), which minimizes the trace of the parameter covariance matrix. The UFI is unsupervised;

Manuscript received November 16, 2010; revised September 30, 2011; accepted December 25, 2011. Date of publication January 12, 2012; date of current version April 18, 2012. This work was supported in part by the National Natural Science Foundation of China under Grant 61125203, Grant 90920303, and Grant 61173186, by the National Basic Research Program of China (973 Program) under Grant 2012CB316400, by the Program for New Century Excellent Talents in University under Grant NCET-09-0685, and by the Ministry of Education under the Scholarship Award for Excellent Doctoral Student Grant. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Gang Hua.

L. Zhang, C. Chen, and J. Bu are with the Zhejiang Provincial Key Laboratory of Service Robot, College of Computer Science, Zhejiang University, Hangzhou 310027, China (e-mail: zljzju@zju.edu.cn; chenc@zju.edu.cn; bjj@zju.edu.cn).

X. He is with the State Key Laboratory of CAD&CG, College of Computer Science, Zhejiang University, Hangzhou 310058, China (e-mail: xiaofeihe@cad.zju.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2012.2183879

therefore, it can be used as a tool for data preprocessing. If the instances are fixed, the UFI reduces to an unsupervised feature selection algorithm, and if the features are fixed, the UFI reduces to A-optimal design (AOD) [14].

The rest of this paper is organized as follows. In Section II, we give a brief review of feature selection and OED. Our proposed UFI is introduced in Section III. In Section IV, we describe an efficient sequential method to solve the optimization problem. Experiments are presented in Section V. Finally, we provide some concluding remarks in Section VI.

II. Related Work

In this section, we give a brief review of the feature selection and OED techniques.

A. Feature Selection

In the last decades, feature selection has been extensively studied in both supervised and unsupervised settings.

Supervised feature selection techniques determine feature relevance by the correlation between feature and class. Fisher score, information gain [24], and relief [25] are several classical supervised methods. These methods select features without involving the learning algorithm that will ultimately be employed, and are usually referred to as filter methods. On the other hand, the wrapper and embedded methods require one predetermined learning algorithm and use its performance as the selection criterion [8]. The wrapper methods [26] utilize the learning algorithm as a black box to score feature subset according to their predictive power. For example, the performance of a support vector machine (SVM) is used to select the most relevant features in [27]. The embedded methods perform feature selection in the process of training and have received much attention in recent years. The most famous embedded methods include the least absolute shrinkage and selection operator [28], least angle regression [29], and ℓ_1-norm regularized SVMs [30], [31].

Due to the lack of labels, unsupervised feature selection is much harder. The unsupervised filter methods usually select features that best preserve the geometrical structure of the data space [6], [7], [32]. The typical algorithms in this category include maximum variance, unsupervised feature selection for PCA [32], and the Laplacian score (LapScore) [7]. Maximum variance selects features with the largest variances and unsupervised feature selection for PCA selects a subset of features that can best reconstruct other features. Different from these two methods, LapScore [7] selects features that best reflect the underlying manifold structure. For unsupervised wrapper and embedded methods, clustering is a commonly used learning algorithm to measure the quality of features [3], [33]–[37]. For example, $Q-\alpha$ [3] measures the cluster coherence by analyzing the spectral properties of the affinity matrix. The feature selection process is based on the optimization over a least-squares objective function.

B. OED

Active learning aims to find the most informative instances such that if they are labeled and used as training data, we can most precisely predict the labels of the other instances. The research literature on active learning is vast [16], [17], [19], [38]–[45]. In statistics, the problem of selecting instances to label is referred to as experimental design. The instance $\mathbf{x}$ is referred to as experiment, and its label y is referred to as measurement. The study of OED [14] is concerned with the design of experiments, which can minimize the variance of a parameterized model.

OED [14], [46], [47] considers the problem of learning a linear function $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$ from experiment–measurement pairs $(\mathbf{z}_i, y_i)$, $i = 1,\ldots,k$. Assume that $y_i = \mathbf{w}^T\mathbf{z}_i + \epsilon_i$, where ϵ_i are independent Gaussian random variables with zero mean and constant variance σ^2. The most popular estimation method is least squares, in which we minimize the residual sum of squares (RSS):

$$\mathrm{RSS}(\mathbf{w}) = \sum_{i=1}^{k} (y_i - f(\mathbf{z}_i))^2. \tag{1}$$

Let $Z = [\mathbf{z}_1,\ldots,\mathbf{z}_k]$ and $\mathbf{y} = [y_1,\ldots,y_k]^T$. The optimal solution is given by

$$\widehat{\mathbf{w}} = (ZZ^T)^{-1}Z\mathbf{y}. \tag{2}$$

It can be proved that $\widehat{\mathbf{w}}$ is an unbiased estimation of $\mathbf{w}$ with the following covariance matrix [1]:

$$\mathrm{Cov}(\widehat{\mathbf{w}}) = \sigma^2(ZZ^T)^{-1}. \tag{3}$$

The goal of OED is to choose instances $\{\mathbf{z}_i\}_{i=1}^k$ from the candidate set to minimize the size of the parameter covariance matrix, which in turn minimizes the confidence region for the estimated parameter $\widehat{\mathbf{w}}$ in some sense. Three of the most popular design criteria are D-optimal design (DOD), AOD, and E-optimal design. DOD minimizes the determinant of $\mathrm{Cov}(\widehat{\mathbf{w}})$ and thus minimizes the volume of the confidence region. AOD minimizes the trace of $\mathrm{Cov}(\widehat{\mathbf{w}})$ and thus minimizes the dimensions of the enclosing box around the confidence region. E-optimal design minimizes the largest eigenvalue of $\mathrm{Cov}(\widehat{\mathbf{w}})$ and thus minimizes the size of the major axis of the confidence region [48].

Another closely related work is active feature selection [49], which combines feature selection and instance selection in a sequential way. The difference is that in this paper, the two problems are simultaneously considered. We aim to develop a unified framework within which feature selection and instance selection can be simultaneously performed, in the hope that the learning performance can be further improved.

III. UFI

In this section, we introduce the UFI. We begin with a formal statement of the problem and the notations.

A. Problem

Let $X \in \mathbb{R}^{n\times m}$ be a data matrix, whose columns correspond to data instances and rows to features. Our goal is to simultaneously find p most informative features and q most informative instances such that, with the selected features as the new representation and the selected instances as the training data, the prediction error of a linear function can be minimized.

We use X_{*i} to denote the ith column (instance) and X_{j*} to denote the jth row (feature) in X. Let $s_1, \ldots, s_q$ be the indexes of the selected instances, and $f_1, \ldots, f_p$ be the indexes of the selected features. Let $H = [X_{f_1*}; \ldots; X_{f_p*}] \in \mathbb{R}^{p \times m}$ denote the data matrix containing only the selected features. Similarly, we denote the ith column of H as H_{*i}, which gives the new representation of the ith instance. Let $Z = [H_{*s_1}, \ldots, H_{*s_q}] \in \mathbb{R}^{p \times q}$ denote the data matrix containing only the selected instances with new representations. Clearly, H is a $p \times m$ submatrix of X, and Z is a $p \times q$ submatrix of X.

B. Criterion

The central idea of our approach is to simultaneously select those features and instances that can minimize the size of the parameter covariance matrix and, in turn, minimize the prediction error.

With the selected p features as the new representation, we consider the problem of learning a linear function as follows:

$$f(\mathbf{h}) = \mathbf{w}^T \mathbf{h} \tag{4}$$

using $\{Z_{*1}, \ldots, Z_{*q}\}$ and their labels $\{y_1, \ldots, y_q\}$ as the training data. In addition, we assume that the observations y_i are independent and have constant variance σ^2. The model parameter $\mathbf{w}$ can be estimated via regularized least squares (RLS) (ridge regression) as follows:

$$\widehat{\mathbf{w}}_{\text{ridge}} = \arg\min_{\mathbf{w}} \sum_{i=1}^{q} (y_i - f(Z_{*i}))^2 + \lambda \|\mathbf{w}\|^2 \tag{5}$$

where $\lambda \geq 0$ is the regularization parameter and $\|\cdot\|$ denotes the vector ℓ_2 norm. The optimal solution of the earlier minimization problem is

$$\widehat{\mathbf{w}}_{\text{ridge}} = (ZZ^T + \lambda I)^{-1} Z\mathbf{y} \tag{6}$$

where I is the identity matrix and $\mathbf{y} = [y_1, \ldots, y_q]^T$. Since $\text{Cov}(\mathbf{y}) = \sigma^2$, the covariance matrix of $\widehat{\mathbf{w}}_{\text{ridge}}$ becomes

$$\text{Cov}(\widehat{\mathbf{w}}_{\text{ridge}}) = \sigma^2 (ZZ^T + \lambda I)^{-1} \left(I - \lambda (ZZ^T + \lambda I)^{-1}\right). \tag{7}$$

Because the regularization parameter λ is usually set to be very small, we can use the following approximation:

$$\text{Cov}(\widehat{\mathbf{w}}_{\text{ridge}}) \approx \sigma^2 (ZZ^T + \lambda I)^{-1}. \tag{8}$$

In statistics, there are different design criteria to measure the size of the covariance matrix, leading to different algorithms. In this paper, we adopt the A-optimality that minimizes the trace of the covariance matrix. However, other design criteria, such as D-optimality and E-optimality, can also be applied in our framework. The definition of our unified criterion is formally stated in the following.

Definition: The UFI is defined as follows:

$$\min_{Z} \quad \text{Tr}(ZZ^T + \lambda I)^{-1} \quad \text{s.t.} \quad Z \in \mathbb{R}^{p \times q} \text{ is a submatrix of } X. \tag{9}$$

IV. Optimization

The optimization problem of the UFI is difficult due to its combinatorial nature. In this section, we develop a greedy algorithm to solve it. The optimization strategy is outlined as follows.

- Initially, we assume that all the features and instances are selected.
- Our sequential optimization approach iteratively removes the least informative features and instances until we obtain p features and q instances.

The detailed algorithmic procedure is presented in Algorithm 1. In our sequential optimization approach, parameter t is required to specify the number of iterations. At each iteration, we remove $\alpha = n - p/t$ features and $\beta = m - q/t$ instances, which are the least informative.

Let $E \in \mathbb{R}^{u \times v}$ denote the data matrix at the current iteration, which contains u features and v instances. We first show how to remove α least informative features from E. Let F be the resulting $(u - \alpha) \times v$ matrix that can be obtained by solving the following optimization problem:

$$\min_{F} \quad \text{Tr}(FF^T + \lambda I)^{-1} \quad \text{s.t.} \quad F \text{ contains } u - \alpha \text{ rows of } E. \tag{10}$$

Following the Woodbury–Morrison formula [50], we have

$$(FF^T + \lambda I)^{-1} = \frac{1}{\lambda} I - \frac{1}{\lambda} F (F^T F + \lambda I)^{-1} F^T.$$

Using the fact that $\text{Tr}(AB) = \text{Tr}(BA)$, the objective function of (10) can be rewritten as

$$\begin{aligned} \text{Tr}(FF^T + \lambda I)^{-1} &= \frac{u - \alpha}{\lambda} - \frac{1}{\lambda} \text{Tr}\left((F^T F + \lambda I)^{-1} F^T F\right) \\ &= \frac{u - \alpha - v}{\lambda} + \text{Tr}(F^T F + \lambda I)^{-1}. \end{aligned}$$

Thus, minimizing $\text{Tr}(FF^T + \lambda I)^{-1}$ is equivalent to minimizing $\text{Tr}(F^T F + \lambda I)^{-1}$. In the following, we discuss how to find the optimal F by sequentially removing rows of E. Initially, we let $F = E$. Let F_{i*} denote the ith row of F; thus, we have

$$F^T F = \sum_i F_{i*}^T F_{i*}.$$

The index of the first row to be deleted is given by

$$l = \arg\min_i \text{Tr}\left(F^T F + \lambda I - F_{i*}^T F_{i*}\right)^{-1}. \tag{11}$$

The most expensive calculation in (11) is the matrix inverse $(F^T F + \lambda I - F_{i*}^T F_{i*})^{-1}$, which need to be computed for each i. We use the Woodbury–Morrison formula to avoid directly inverting a matrix. Let $M = (F^T F + \lambda I)^{-1}$; we have

$$\begin{aligned} \text{Tr}\left(F^T F + \lambda I - F_{i*}^T F_{i*}\right)^{-1} &= \text{Tr}\left(M + \frac{M F_{i*}^T F_{i*} M}{1 - F_{i*} M F_{i*}^T}\right) \\ &= \text{Tr}(M) + \frac{F_{i*} M^2 F_{i*}^T}{1 - F_{i*} M F_{i*}^T}. \end{aligned}$$

Since $\mathrm{Tr}(M)$ is a constant for all i, problem (11) reduces to

$$l = \arg\min_i \frac{F_{i*}M^2F_{i*}^T}{1 - F_{i*}MF_{i*}^T}. \quad (12)$$

Once the least informative feature, i.e., F_{l*}, is obtained, we update matrix F by removing the row vector F_{l*}, and the matrix M is updated according to the following formula:

$$M \leftarrow M + \frac{MF_{i*}^TF_{i*}M}{1 - F_{i*}MF_{i*}^T}. \quad (13)$$

We repeat this process until α rows (i.e., features) have been removed from matrix E.

After obtaining F, we need to remove β columns from F, which correspond to the least informative instances, finally leading to a $(u - \alpha) \times (v - \beta)$ submatrix denoted by G. The parameter covariance matrix corresponding to the $u - \alpha$ features and $v - \beta$ instances is $(GG^T + \lambda I)^{-1}$. Using the A-optimality criterion, the optimal G can be found by solving the following optimization problem:

$$\min_G \quad \mathrm{Tr}(GG^T + \lambda I)^{-1} \quad \text{s.t.} \quad G \text{ contains } v - \beta \text{ columns of } F. \quad (14)$$

As before, G is initially set to be F. Since $GG^T = \sum_i G_{*i}G_{*i}^T$, the index of the first least informative instance is given by

$$k = \arg\min_i \mathrm{Tr}\left(GG^T + \lambda I - G_{*i}G_{*i}^T\right)^{-1}. \quad (15)$$

As shown, (15) is essentially the same as (11); therefore, we can apply the same computational method to find the optimal G. Define $N = (GG^T + \lambda I)^{-1}$. Then, we have

$$\mathrm{Tr}\left(GG^T + \lambda I - G_{*i}G_{*i}^T\right)^{-1} = \mathrm{Tr}(N) + \frac{G_{*i}^TN^2G_{*i}}{1 - G_{*i}^TNG_{*i}}.$$

As a result, the problem (15) reduces to

$$k = \arg\min_i \frac{G_{*i}^TN^2G_{*i}}{1 - G_{*i}^TNG_{*i}}. \quad (16)$$

Once the least informative instance is selected, we update G by removing the kth column vector. In addition, the matrix N is updated as follows:

$$N \leftarrow N + \frac{NG_{*k}G_{*k}^TN}{1 - G_{*k}^TNG_{*k}^T}. \quad (17)$$

This process is repeated until β columns (i.e., instances) have been removed.

Algorithm 1 The sequential algorithm for UFI

Input: The $n \times m$ data matrix X, the number of features to be selected p, the number of instances to be selected q, the ridge regularizer λ, the number of iterations t

Output: The submatrix Z, which contains the most informative features and instances

$\alpha \leftarrow (n - p/t)$
$\beta \leftarrow (m - q/t)$
$E \leftarrow X$
for $i = 1$ to t **do**
  $F \leftarrow \mathrm{DelRow}(E, \lambda, \alpha)$
  $G \leftarrow \mathrm{DelColumn}(F, \lambda, \beta)$
  $E \leftarrow G$
end for
$Z \leftarrow E$

return Z
procedure $\mathrm{DelRow}(E, \lambda, \alpha)$ **do**
  $F \leftarrow E$
  $M \leftarrow (F^TF + \lambda I)^{-1}$
  for $j = 1$ to α **do**
    $l \leftarrow \arg\min_i(F_{i*}M^2F_{i*}^T)/(1 - F_{i*}MF_{i*}^T)$
    $F \leftarrow$ remove the l-th row of F
    $M \leftarrow M + (MF_{l*}^TF_{l*}M)/(1 - F_{l*}MF_{l*}^T)$
  end for
  return F
end procedure
procedure $\mathrm{DelColumn}(F, \lambda, \beta)$ **do**
  $G \leftarrow F$
  $N \leftarrow (GG^T + \lambda I)^{-1}$
  for $j = 1$ to β **do**
    $k \leftarrow \arg\min_i(G_{*i}^TN^2G_{*i})/(1 - G_{*i}^TNG_{*i})$
    $G \leftarrow$ remove the k-th column of G
    $N \leftarrow N + (NG_{*k}G_{*k}^TN)/(1 - G_{*k}^TNG_{*k})$
  end for
  return G
end procedure

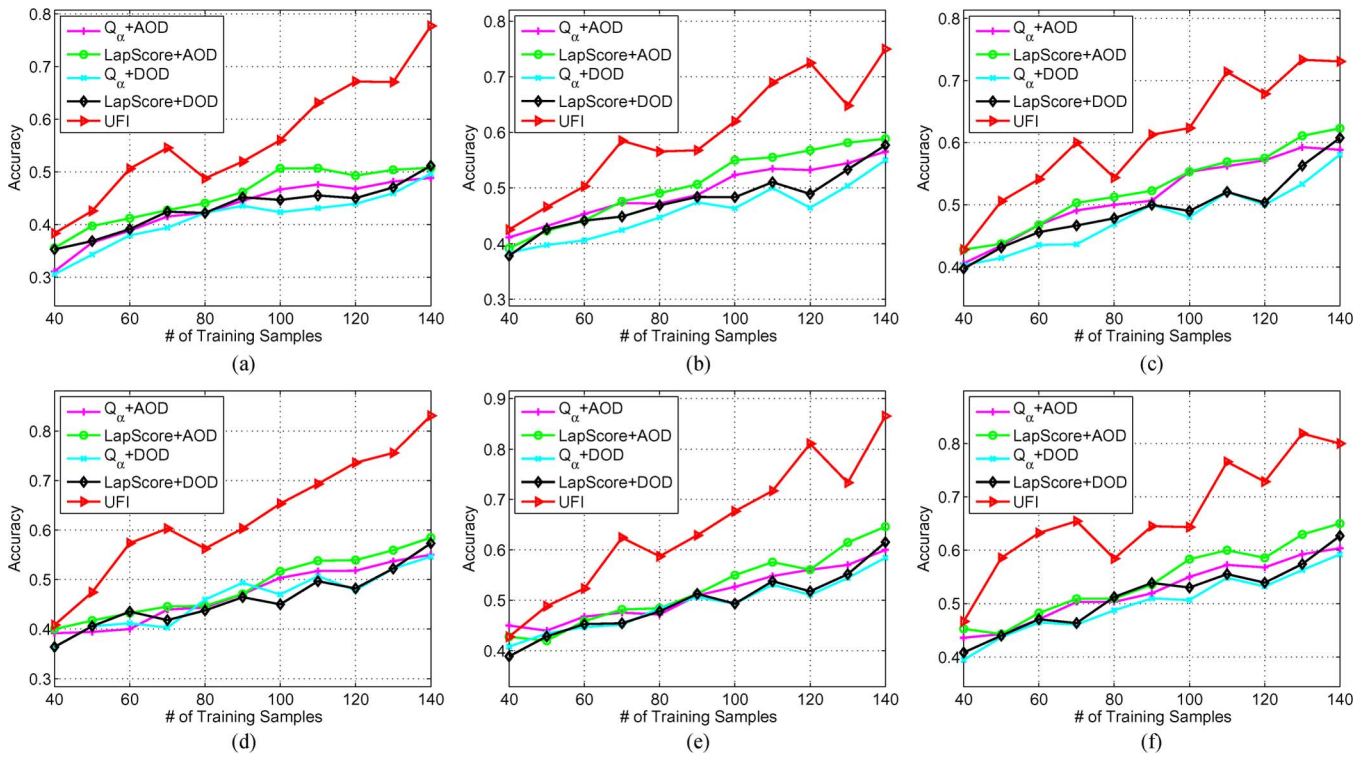


Fig. 1. Comparisons with baselines that perform feature selection and active learning independently on the ORL face data set. (a) 300 features selected and classified by RLS. (b) 500 features selected and classified by RLS. (c) 700 features selected and classified by RLS. (d) 300 features selected and classified by the SVM. (e) 500 features selected and classified by SVM. (f) 700 features selected and classified by the SVM.

TABLE I
DESCRIPTION OF THE DATA SETS

Data Sets	# classes	# instances	# features
ORL	40	400	1024
COIL	6	1500	241

V. EXPERIMENTS

A. Experimental Settings

In this section, we perform classification experiments to demonstrate the effectiveness of the UFI. We select two benchmark data sets for our evaluation: 1) The ORL face data set, which has been a benchmark in face recognition [51], [52];[1] and 2) The COIL data set used in the semisupervised learning book [53].[2] Table I gives the statistics of the data sets used in our experiment.

For each data set, UFI is applied to simultaneously select the most informative features and instances. Then, the whole data set is represented by the selected features. We use the selected instances and their labels to train a classifier, which is used to predict the labels of the unselected instances. The classification accuracy is used to measure the performance. To handle multiclass classification problem, we adopt the one-versus-all (OVA) scheme. If the training data contain c classes, the OVA scheme trains c binary classifiers, and each binary classifier separates one class (positive) from all the other classes (negative). To classify a new testing instance, these c classifiers are all applied, and the class label is determined by the classifier whose output value is the largest. Since our algorithm is based on experimental design, RLS is used to train a linear classifier in our experiments. We also report the classification results obtained by using the classical SVM [54], [55] as the classifier.

For comparison, we design 12 baselines that are combinations of state-of-the-art feature selection and active learning algorithms. The feature selection algorithms that we used are the LapScore [7] and $Q-\alpha$ (Q_α) [3]. The active learning algorithms are AOD and DOD.[3] In the first type of baselines, we independently apply the feature selection and active learning algorithms to select the most informative features and instances, which results in four baselines. We denote this type of baselines in the form of A+B. In the second type of baselines, we perform feature selection and active learning in a sequential way, which leads to the other eight baselines. We denote this type of baselines in the form of A $\rightarrow$ B, which means algorithm A is performed before algorithm B.

B. Classification Results

Comparison With Baselines That Select Features and Instances Independently: Figs. 1 and 2 show the classification results on the ORL and COIL data sets, respectively.

On the ORL data set, we apply UFI, LapScore + AOD, LapScore + DOD, Q_α + AOD, and Q_α + DOD to select p $(= 300, 500, 700)$ features and q $(= 40, 50, \ldots, 140)$ instances. The classification results obtained by using RLS as the classifier are shown in Fig. 1(a)–(c), whereas the classification results achieved by the SVM are shown in Fig. 1(d)–(f). As shown, our

[1]http://www.zjucadcg.cn/dengcai/Data/data.html

[2]www.kyb.tuebingen.mpg.de/ssl-book/

[3]We developed two forward stepwise selection methods for solving the optimization problems of AOD and DOD.

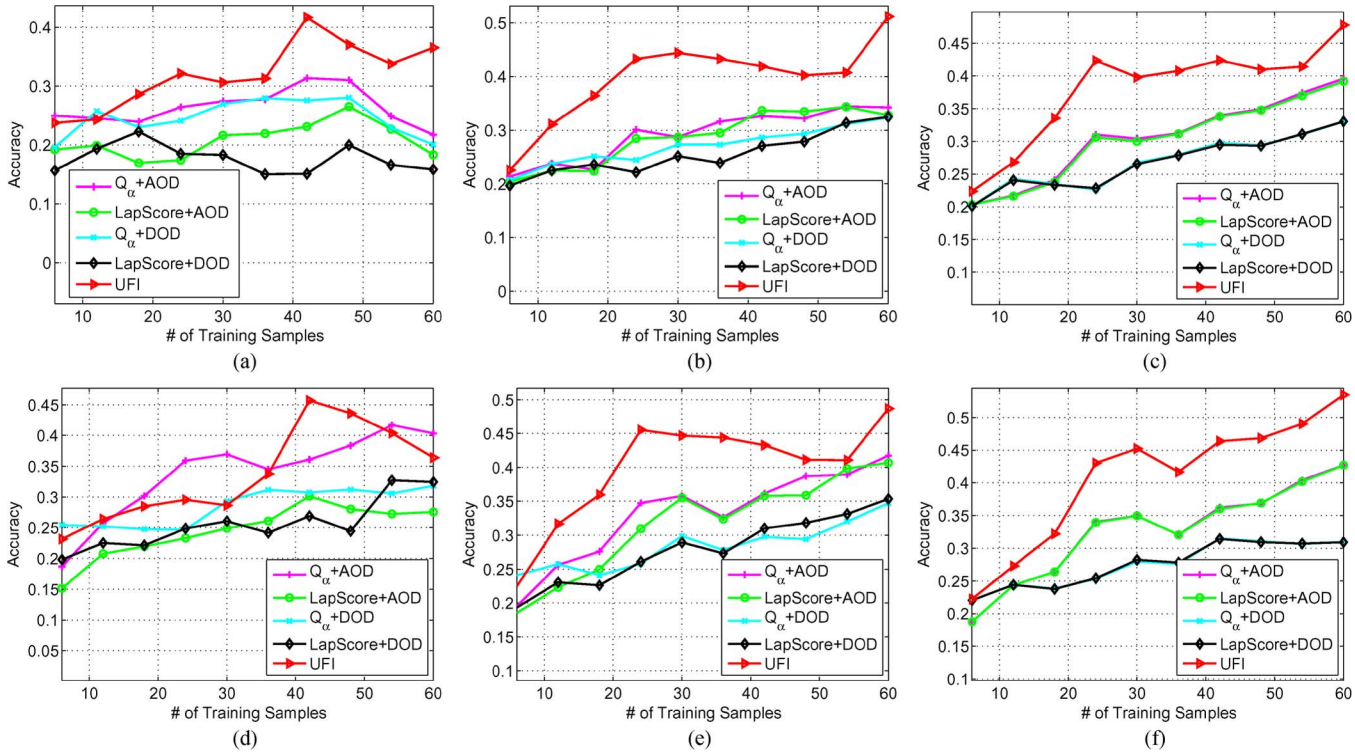


Fig. 2. Comparisons with baselines that perform feature selection and active learning independently on the COIL data set. (a) 60 features selected and classified by RLS. (b) 140 features selected and classified by RLS. (c) 220 features selected and classified by RLS. (d) 60 features selected and classified by the SVM. (e) 140 features selected and classified by the SVM. (f) 220 features selected and classified by the SVM.

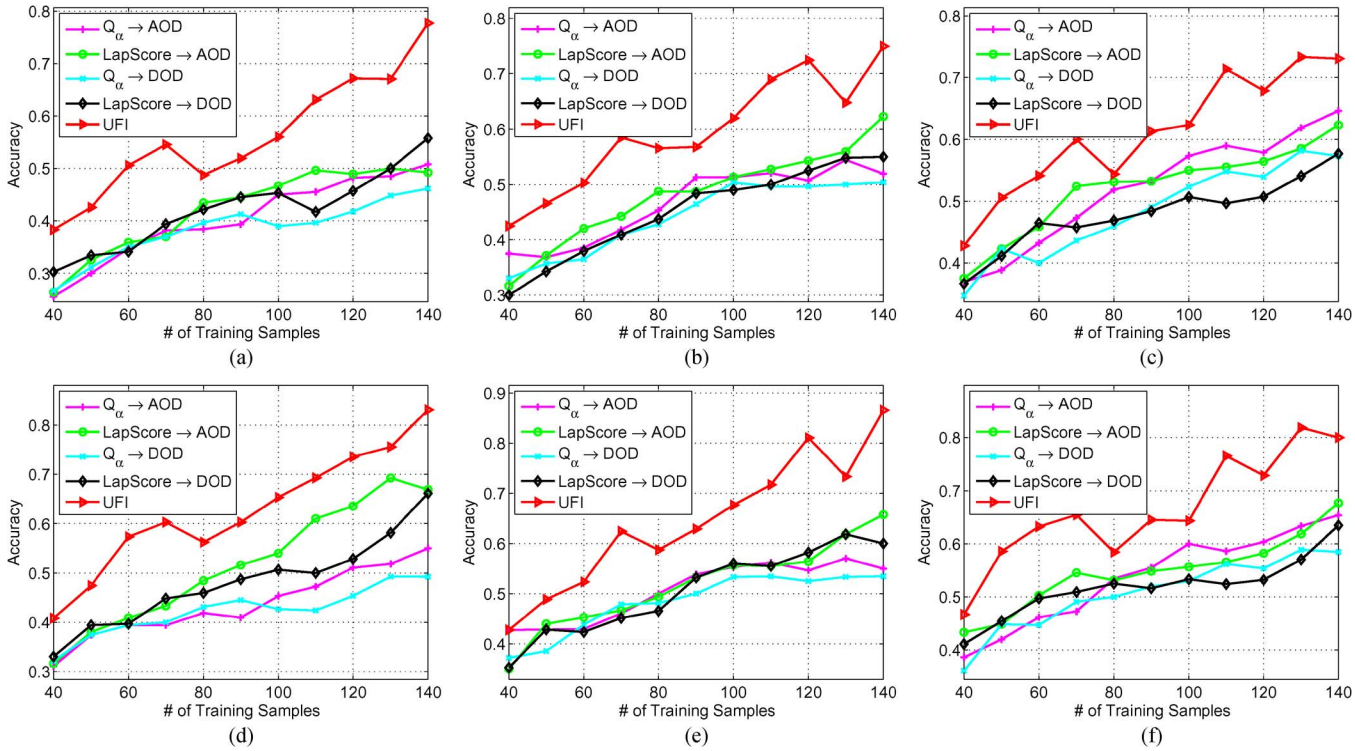


Fig. 3. Comparisons with baselines that select features first on the ORL face data set. (a) 300 features selected and classified by RLS. (b) 500 features selected and classified by RLS. (c) 700 features selected and classified by RLS. (d) 300 features selected and classified by the SVM. (e) 500 features selected and classified by the SVM. (f) 700 features selected and classified by the SVM.

proposed UFI significantly outperforms the other four baselines in all the cases.

Consider the case in Fig. 1(a), where all the algorithms select 300 features. The performance of the UFI with 60 instances

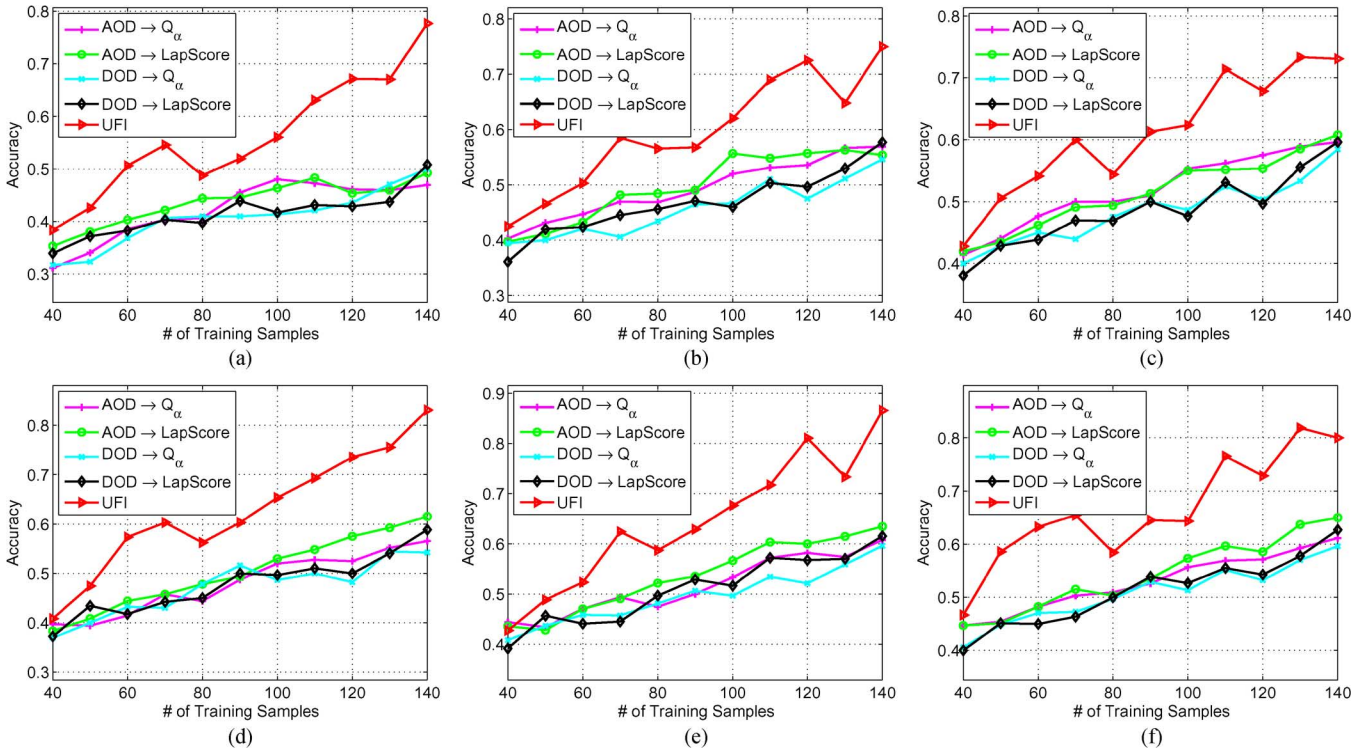


Fig. 4. Comparisons with baselines that select instances first on the ORL face data set. (a) 300 features selected and classified by RLS. (b) 500 features selected and classified by RLS. (c) 700 features selected and classified by RLS. (d) 300 feature selected and classified by the SVM. (e) 500 features selected and classified by the SVM. (f) 700 features selected and classified by SVM.

selected is better than or comparable with that of the four baselines with 140 instances selected. Thus, for the purpose of active learning, the labeling cost can be significantly reduced by using the UFI. On the other hand, let us compare different algorithms by focusing on the case that all of them select the same number of instances, e.g.,100. In Fig. 1(a), we can see that the classification accuracy of the UFI with 100 instances and 300 features selected is about 0.55. In Fig. 1(b), we can see that the best baseline requires 500 features to achieve similar performance. Therefore, our method is more capable of identifying the most informative features.

On the COIL data set, we apply UFI, LapScore + AOD, LapScore + DOD, Q_α + AOD, and Q_α + DOD to select p $(= 60, 140, 220)$ features and q $(= 6, 12, \ldots, 60)$ instances. As indicated in Fig. 2, UFI performs the best in most cases. In Fig. 2(a) and (d), we can see that when only a small number of features is selected, the performance of all the algorithms is not stable. In addition, the advantage of UFI is limited. However, as the number of features increases, the advantage of UFI becomes more and more obvious.

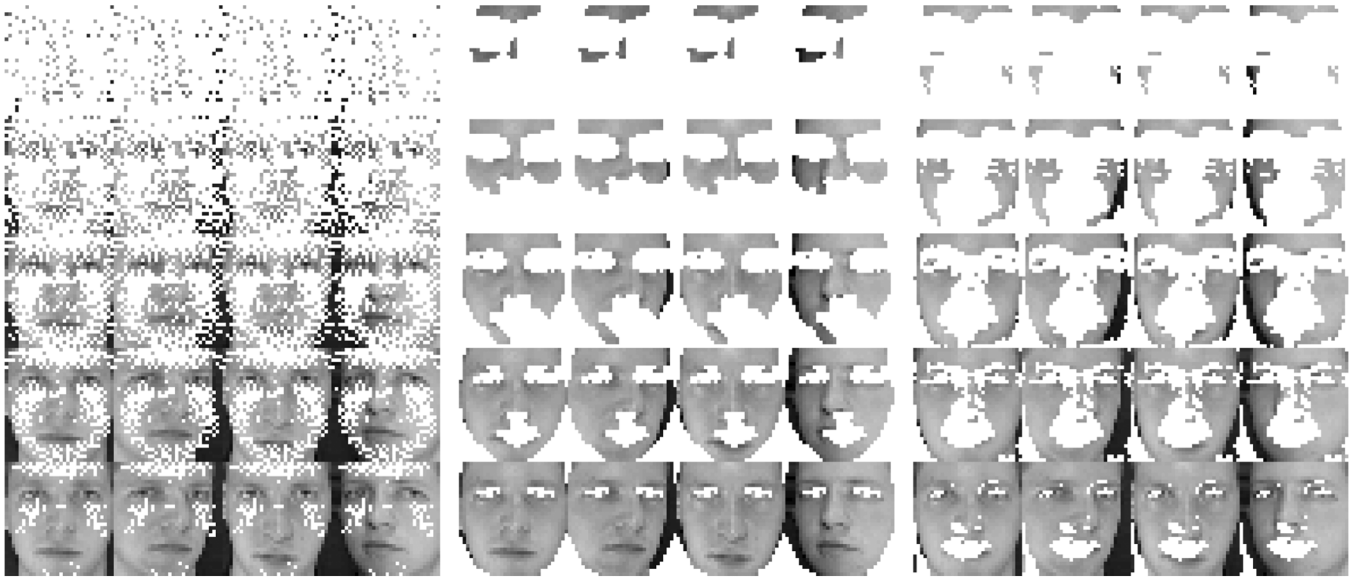
Comparison With Baselines That Select Features and Instances Sequentially: In Figs. 3 and 4, we compare the UFI with the second type of baselines on the ORL data set. From the two figures, it is clear that the UFI has big advantages over the second type of baselines. Comparing Fig. 1, 3, and 4, we can see that there is no significant difference between the two types of baselines. For brevity, we omit the results on the COIL data set since similar behaviors are observed on this data set.

Summary: We summarize some important points in the following.

- In general, the classification accuracy keeps on increasing as the number of training examples increases. In some cases, the performance may decrease when more training data are added. This fluctuation is mainly because the testing set changes as more data are used for training since we evaluate the classification accuracy on the unselected points.
- The classification accuracy does not necessary increases when more features are selected, which can be shown in Fig. 1(e) and (f). This observation again verifies that a large amount of features can be removed without hurting the performance.
- The overall classification accuracy achieved by the SVM is slightly better than that achieved by RLS. Thus, although our UFI is built on RLS, it works well with other types of classifiers.
- In all the experiments, the UFI exhibits an obvious improvement over all the baselines. Thus, considering simultaneously feature and instance selection indeed improves the learning performance.

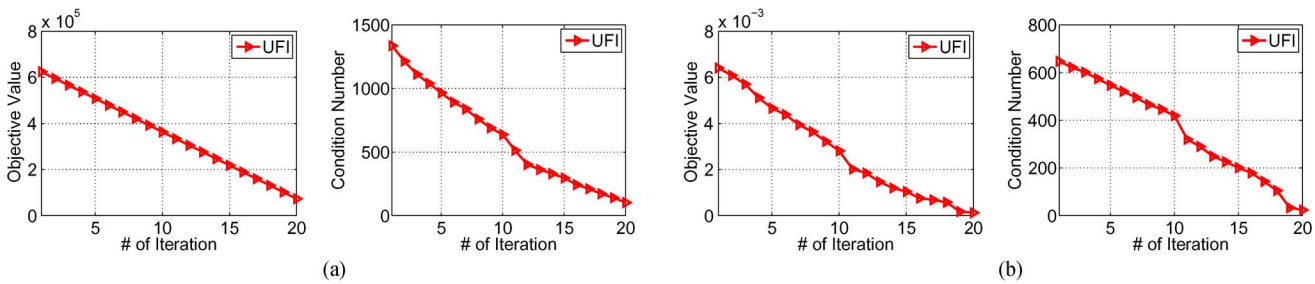
C. The Features

Note that when all the instances are selected, the UFI becomes a novel feature selection algorithm. It would be interesting to see what features are selected by different algorithms. Since features on faces are easy to visualize, we take the ORL face data set as an example. In Fig. 5, we show the selected pixels on the faces using UFI, $Q-\alpha$, and LapScore. The unselected pixels are removed from those faces. The first, second, third, fourth, and



(a) (b) (c)

Fig. 5. Pixels selected by different algorithms on the ORL face data set. The first, second, third, fourth, and fifth lines in each subfigure correspond to the results of 100, 300, 500, 700, and 900 pixels selected by each algorithm. (a) UFI. (b) $Q-\alpha$. (c) LapScore.



(a) (b)

Fig. 6. Empirical studies of the greed algorithm on the two data sets. Here, we plot the value of the objective function and the condition number of the data matrix versus the number of iterations. (a) ORL data set. (b) COIL data set.

fifth lines in each subfigure correspond to the results of 100, 300, 500, 700, and 900 pixels selected by each algorithm.

As shown, the pixels selected by the UFI distribute more evenly on the whole face. In addition, the UFI tends to preserve the pixels in the area of two eyes, nose, mouth, and face contour. On the other hand, Both $Q-\alpha$ and LapScore first remove the pixels in the area of two eyes, nose, and mouth. Clearly, the features selected by UFI are more consistent with human perception.

D. Analysis of the Optimization Algorithm

In the following, we conduct some empirical studies to analyze the greedy algorithm developed in Section IV. We apply the UFI to select 10% features and 10% instances on the two data sets. The regularization parameter λ is set to $1e-3$, and the number of iterations is set to 20.

In Fig. 6, we present how the value of the objective function $\mathrm{Tr}(ZZ^T+\lambda I)^{-1}$ and the condition number of the data matrix Z change as the iterative algorithm proceeds. First, we can see that the objective function decreases rapidly as the number of iterations increases. Second, we observe that the condition number of the data matrix Z also monotonically decreases. Thus, our algorithm tends to make the data matrix Z well conditioned, which is an important reason why the classifier trained on the data matrix generated by the UFI has better performance.

VI. Conclusion

We have considered the novel problem of simultaneously selecting the most informative features and instances. Based on OED, we introduce a novel unified criterion for both feature selection and instance selection. By using the selected instances and features as the training data, the trace of the parameter covariance matrix and, in turn, the prediction error can be minimized. Our empirical tests on two standard data sets have demonstrated that we can benefit from simultaneously considering these two problems.

Since we develop our algorithm under the framework of OED, it is unsupervised. However, in practice, prior knowledge such as class labels [17] or pairwise constraints [56] may be available. In the future, we will investigate how to apply UFI under supervised or semisupervised setting.

References

[1] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. New York: Springer-Verlag, 2009.

[2] Z. Xu, R. Jin, J. Ye, M. R. Lyu, and I. King, "Non-monotonic feature selection," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, 2009, pp. 1145–1152.

[3] L. Wolf and A. Shashua, "Feature selection for unsupervised and supervised inference: The emergence of sparsity in a weight-based approach," *J. Mach. Learn. Res.*, vol. 6, pp. 1855–1887, Dec. 2005.

[4] P. Mitra, C. A. Murthy, and S. K. Pal, "Unsupervised feature selection using feature similarity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 301–312, Mar. 2002.

[5] A. Destrero, C. De Mol, F. Odone, and A. Verri, "A sparsity-enforcing method for learning face features," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 188–201, Jan. 2009.

[6] Z. Zhao and H. Liu, "Spectral feature selection for supervised and unsupervised learning," in *Proc. 24th Int. Conf. Mach. Learn.*, 2007, pp. 1151–1157.

[7] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," *Adv. Neural Inf. Process. Syst.*, vol. 18, pp. 507–514, 2006.

[8] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Mar. 2003.

[9] J. Zhao, K. Lu, and X. He, "Locality sensitive semi-supervised feature selection," *Neurocomputing*, vol. 71, no. 10–12, pp. 1842–1849, Jun. 2008.

[10] A. Jain and D. Zongker, "Feature selection: evaluation, application, and small sample performance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 2, pp. 153–158, Feb. 1997.

[11] D. Liu, G. Hua, P. A. Viola, and T. Chen, "Integrated feature selection and higher-order spatial feature extraction for object categorization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[12] P. K. Mallapragada, R. Jin, and A. K. Jain, "Online visual vocabulary pruning using pairwise constraints," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 3073–3080.

[13] L. Zhang, C. Chen, J. Bu, Z. Chen, S. Tan, and X. He, "Discriminative codeword selection for image representation," in *Proc. 18th ACM Int. Conf. Multimedia*, 2010, pp. 173–182.

[14] A. Atkinson, A. Donev, and R. Tobias*, Optimum Experimental Designs, with SAS*. New York: Oxford Univ. Press, 2007.

[15] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," *J. Artif. Intell. Res.*, vol. 4, no. 1, pp. 129–145, Jan. 1996.

[16] B. Settles, Active learning literature survey Univ. Wisconsin–Madison, Dept. Comput. Sci., Madison, WI, Tech. Rep. 1648, 2009.

[17] S. Tong and E. Chang, "Support vector machine active learning for image retrieval," in *Proc. 9th ACM Int. Conf. Multimedia*, 2001, pp. 107–118.

[18] L. Zhang, C. Chen, W. Chen, J. Bu, D. Cai, and X. He, "Convex experimental design using manifold structure for image retrieval," in *Proc. 17th ACM Int. Conf. Multimedia*, 2009, pp. 45–54.

[19] X. Tian, D. Tao, X.-S. Hua, and X. Wu, "Active reranking for web image search," *IEEE Trans. Image Process.*, vol. 19, no. 3, pp. 805–820, Mar. 2010.

[20] A. Kapoor, G. Hua, A. Akbarzadeh, and S. Baker, "Which faces to tag: Adding prior constraints into active learning," in *Proc. 12th IEEE Int. Conf. Comput. Vis.*, 2009, pp. 1058–1065.

[21] L. Zhang, C. Chen, J. Bu, D. Cai, X. He, and T. S. Huang, "Active learning based on locally linear reconstruction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 10, pp. 2026–2038, Oct. 2011.

[22] K. Yu, S. Zhu, W. Xu, and Y. Gong, "Non-greedy active learning for text categorization using convex transductive experimental design," in *Proc. 31st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2008, pp. 635–642.

[23] I. S. Dhillon, "Co-clustering documents and words using bipartite spectral graph partitioning," in *Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2001, pp. 269–274.

[24] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *Proc. 14th Int. Conf. Mach. Learn.*, 1997, pp. 412–420.

[25] K. Kira and L. A. Rendell, "The feature selection problem: Traditional methods and a new algorithm," in *Proc. 10th Nat. Conf. Artif. Intell.*, 1992, pp. 129–134.

[26] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, no. 1/2, pp. 273–324, Dec. 1997.

[27] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik, "Feature selection for SVMs," *Adv. Neural Inf. Process. Syst.*, vol. 13, pp. 668–674, 2001.

[28] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Statist. Soc., Ser. B, Methodological*, vol. 58, no. 1, pp. 267–288, 1996.

[29] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Statist.*, vol. 32, no. 2, pp. 407–451, Apr. 2004.

[30] G. Fung and O. L. Mangasarian, "Data selection for support vector machine classifiers," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2000, pp. 64–70.

[31] T. Helleputte and P. Dupont, "Partially supervised feature selection with regularized linear models," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, 2009, pp. 409–416.

[32] C. Boutsidis, M. W. Mahoney, and P. Drineas, "Unsupervised feature selection for principal components analysis," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 61–69.

[33] C. Boutsidis, M. Mahoney, and P. Drineas, "Unsupervised feature selection for the k-means clustering problem," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, vol. 22, pp. 153–161.

[34] J. G. Dy and C. E. Brodley, "Feature selection for unsupervised learning," *J. Mach. Learn. Res.*, vol. 5, pp. 845–889, Dec. 2004.

[35] M. H. C. Law, M. A. T. Figueiredo, and A. K. Jain, "Simultaneous feature selection and clustering using mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1154–1166, Sep. 2004.

[36] V. Roth and T. Lange, "Feature selection in clustering problems," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, vol. 16, pp. 473–480.

[37] D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for multi-cluster data," in *Proc. 16th ACM SIGKDD*, 2010, pp. 333–342.

[38] F. R. Bach, B. Schölkopf, J. Platt, and T. Hoffman, Eds., "Active learning for misspecified generalized linear models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, vol. 19, pp. 65–72.

[39] P. Gosselin and M. Cord, "Active learning methods for interactive image retrieval," *IEEE Trans. Image Process.*, vol. 17, no. 7, pp. 1200–1211, Jul. 2008.

[40] X. Zhu, J. Lafferty, and Z. Ghahramani, "Combining active learning and semi-supervised learning using gaussian fields and harmonic functions," in *Proc. ICML—Workshop on Continuum Labeled to Unlabeled Data in Machine Learning and Data Mining*, 2003, pp. 58–65.

[41] N. Roy and A. McCallum, "Toward optimal active learning through sampling estimation of error reduction," in *Proc. 18th Int. Conf. Machine Learn.*, 2001, pp. 441–448.

[42] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proc. 5th Annu. Workshop Comput. Learn. Theory*, 1992, pp. 287–294.

[43] Y. Guo and R. Greiner, "Optimistic active learning using mutual information," in *Proc. 20th Int. Joint Conf. Artif. Intell.*, Hyderabad, India, 2007, pp. 823–829.

[44] X. He, W. Min, D. Cai, and K. Zhou, "Laplacian optimal design for image retrieval," in *Proc. 30th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2007, pp. 119–126.

[45] X. He, "Laplacian regularized D-optimal design for active learning and its application to image retrieval," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 254–263, Jan. 2010.

[46] S. Boyd and L. Vandenberghe*, Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[47] K. Yu, J. Bi, and V. Tresp, "Active learning via transductive experimental design," in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 1081–1088.

[48] S. P. Asprey and S. Macchietto, "Designing robust optimal dynamic experiments," *J. Process Control*, vol. 12, no. 4, pp. 545–556, Jun. 2002.

[49] H. Liu, H. Motoda, and L. Yu, "A selective sampling approach to active feature selection," *Artif. Intell.*, vol. 159, no. 1/2, pp. 49–74, Nov. 2004.

[50] G. Strang*, Introduction to Linear Algebra*, 3rd ed. Wellesley, MA: Wellesley-Cambridge Press, 2003.

[51] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using Laplacianfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 3, pp. 328–340, Mar. 2005.

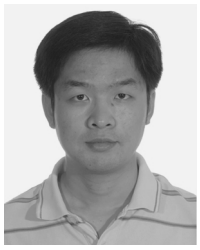
[52] D. Cai, X. He, and J. Han, Using graph model for face analysis Dept. Comput. Sci., UIUC, Champaign, IL, Tech. Rep. UIUCDCS-R-2005-2636, 2005.

[53] , O. Chapelle, B. Schölkopf, and A. Zien, Eds.*, Semi-Supervised Learning*. Cambridge, MA: MIT Press, 2006.

[54] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining Knowl. Discovery*, vol. 2, no. 2, pp. 121–167, Jun. 1998.

[55] C.-C. Chang and C.-J. Lin, LIBSVM: A Library for Support Vector Machines 2001 [Online]. Available: http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[56] T. Yang, R. Jin, and A. K. Jain, "Learning from noisy side information by generalized maximum entropy model," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 1199–1206.



Lijun Zhang (S'10) received the B.S. degree in computer science from Zhejiang University, Hangzhou, China, in 2007. He is currently working toward the Ph.D. degree in computer science with Zheijiang University.

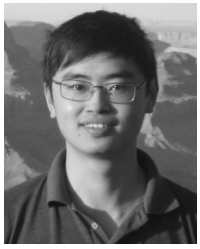
His research interests include machine learning, information retrieval, and data mining.

Chun Chen (M'06) received the B.S. degree in mathematics from Xiamen University, Xiamen, China, in 1981, and the M.S. and Ph.D. degrees in computer science from Zhejiang University, Hangzhou, China, in 1984 and 1990, respectively.

He is a Professor with the College of Computer Science, Zhejiang University. His research interests include information retrieval, data mining, computer vision, computer graphics, and embedded technology.

Jiajun Bu (M'06) received the B.S. and Ph.D. degrees in computer science from Zhejiang University, Hangzhou, China, in 1995 and 2000, respectively.

He is a Professor with the College of Computer Science, Zhejiang University. His research interests include embedded system, data mining, information retrieval, and mobile database.



Xiaofei He (SM'10) received the B.S. degree in computer science from Zhejiang University, Hangzhou, China, in 2000 and the Ph.D. degree in computer science from the University of Chicago, Chicago, in 2005.

He is a Professor with the State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou, China. Prior to joining the Zhejiang University in 2007, he was a Research Scientist with Yahoo! Research Labs, Burbank, CA. His research interests include machine learning, information retrieval, and computer vision.