Adaptive Unsupervised Multi-View Feature Selection for Visual Concept Recognition

Yinfu Feng[1], Jun Xiao[1], Yueting Zhuang[1], Xiaoming Liu[2]

[1] School of Computer Science, Zhejiang University, Hangzhou 310027, P.R.China
[2] Department of Computer Science and Engineering, Michigan State University, USA

Abstract. To reveal and leverage the correlated and complemental information between different views, a great amount of multi-view learning algorithms have been proposed in recent years. However, unsupervised feature selection in multi-view learning is still a challenge due to lack of data labels that could be utilized to select the discriminative features. Moreover, most of the traditional feature selection methods are developed for the single-view data, and are not directly applicable to the multi-view data. Therefore, we propose an unsupervised learning method called Adaptive Unsupervised Multi-view Feature Selection (AUMFS) in this paper. AUMFS attempts to jointly utilize three kinds of vital information, i.e., data cluster structure, data similarity and the correlations between different views, contained in the original data together for feature selection. To achieve this goal, a robust sparse regression model with the $l_{2,1}$-norm penalty is introduced to predict data cluster labels, and at the same time, multiple view-dependent visual similar graphs are constructed to flexibly model the visual similarity in each view. Then, AUMFS integrates data cluster labels prediction and adaptive multi-view visual similar graph learning into a unified framework. To solve the objective function of AUMFS, a simple yet efficient iterative method is proposed. We apply AUMFS to three visual concept recognition applications (i.e., social image concept recognition, object recognition and video-based human action recognition) on four benchmark datasets. Experimental results show the proposed method significantly outperforms several state-of-the-art feature selection methods. More importantly, our method is not very sensitive to the parameters and the optimization method converges very fast.

1 Introduction

Owing to the increasingly powerful computational capabilities and the rapid development of feature selection techniques, objects are often represented by multiple heterogenous features from various representations in many visual concept recognition tasks [1–3]. Each representation of feature characterizes these objects in one specific feature space and has particular physical meaning and statistic property. Conventionally this type of data is named as multi-view[1] data to distinguish from the single-view data. One typical example is that a color image can be represented by multiple heterogeneous

[1] The term "multi-view" in our paper and many related works refers to that the object is represented by multiple features, while in some literatures from computer vision field it means that the object is represented by a set of images acquired from different viewpoints.

visual features, such as global features [1] (e.g., color, texture and shape) and local features (e.g., SIFT [4], LBP [5] and GLOH [6]). Similarly, human action is often associated with multiple visual features, which can be either appearance features (e.g., color, texture, edge) or motion features (e.g., motion history and optical flow) [3].

Since different views of features characterize different aspects of the objects and have different intrinsic discriminative power, an intuitive idea is to combine them to improve the recognition performance. However, most traditional data mining and machine learning methods are developed for the single-view data scenario, and they may not be applied to the multi-view data directly [7]. To tackle this problem, a straightforward solution is to concatenate features of all views and transform a multi-view data into a single-view data. However, this solution disregards the underlying correlations between different views, and moreover, it also lacks of physical meaning. On the other hand, it has shown extensively in prior research that leveraging information contained in multiple views can dramatically improve the learning performance [7–11]. As a result, multi-view learning research has been continuing to flourish in recent years. A great deal of efforts have been carried out in this field with a wide variety of applications, such as clustering [8, 11], classification [9, 10] and dimensionality reduction [7, 12].

To the best of our knowledge, little progress has been made on multi-view feature selection, whereas it plays a crucial role in learning more compact and accurate feature representation from the original multiple high-dimensional features. In general, feature selection has twofold advantages [13]: 1) the learned feature subset has lower dimensionality than the original one, making the subsequential computation more efficient; 2) most relevant features can be selected, thus irrelevant and noisy features are discarded, potentially leading to more accurate results.

Based on whether the data labels are available, existing feature selection methods can be broadly divided into two categories, i.e., supervised feature selection methods and unsupervised feature selection methods. The former methods usually select discriminative features according to labels of the training data, such as Fisher Score [14] and sparse multi-output regression [15]. While the latter ones, such as Laplacian Score [16], Feature Ranking [17] and Multi-Cluster Feature Selection [18], select features best preserve the data similarity or manifold structure derived from the whole feature set. It is well known that, in many real world applications, labeled data are limited while unlabeled data are ample. Also, the unlabeled data are much easier to obtain than the labeled ones. Consequently, there is a growing need for effective and efficient unsupervised learning approaches.

However, most of the existing unsupervised learning methods are also developed for the single-view data, and thus they fail to leverage the correlated and complemental information between different views when they are applied to the multi-view data. Furthermore, in addition to exploit data similarity or manifold structure information, some researchers recently suggested to utilize data cluster labels to select discriminative features in the unsupervised scenario [13, 19]. But, both [13] and [19] are devised for the single-view scenario, so they still suffer from the aforementioned problem. Meanwhile, the cluster label prediction functions used in [13, 19] are not robust, which will be discussed in Section 2.

In light of this, we propose an unsupervised multi-view learning method called Adaptive Unsupervised Multi-view Feature Selection (AUMFS) algorithm in this paper. The flowchart of AUMFS is illustrated in Fig.1. AUMFS integrates three kinds of vital information, i.e., data cluster structure, data similarity and the correlations between different views, together for the unsupervised multi-view feature selection. Specifically, an improved robust sparse regression model with the $l_{2,1}$-norm penalty is adopted to predict data cluster labels based on data cluster structure. At the same time, multiple view-dependent visual similar graphs are constructed to flexibly model the visual similarity in each view and then these learned graphs are united with a non-negative view-weight vector to form the objective function of adaptive multi-view visual similar graph learning, which leverages the correlations between different views and establishes adaptive weights for each view. Finally, we integrate data cluster labels prediction and adaptive multi-view visual similar graph learning into a unified framework. Based on this framework, we can simultaneously estimate data cluster labels, adaptive view weights, and feature selection matrix. We apply AUMFS to three visual concept recognition tasks and compare it with several state-of-the-art methods. Our extensive experiments on four benchmark datasets show that AUMFS has very competitive performance with state-of-the-art feature selection methods. More importantly, AUMFS is not very sensitive to the parameters and the optimization method converges very fast.

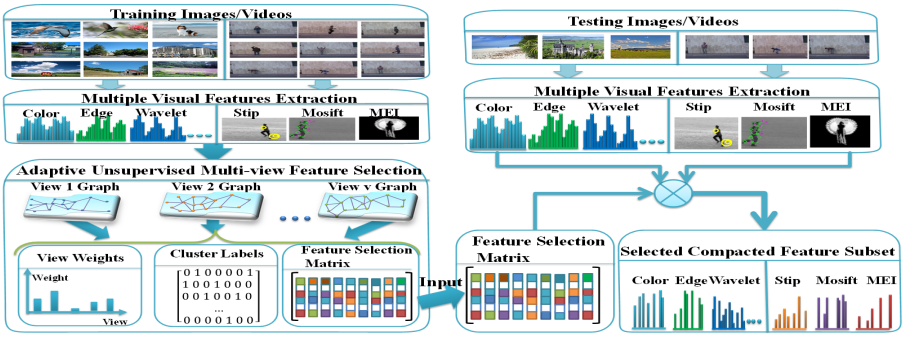


Fig. 1. The flowchart of proposed AUMFS.

The remainder of this paper is organized as follows. In Section 2, we introduce the details of AUMFS, followed by its optimization method. Experiments on three visual concept recognition applications are presented in Section 3 and Section 4 concludes the paper.

2 Proposed Methodology

2.1 Notations

To better present the details of AUMFS, we provide some important notations used in the rest of this paper. Capital letters, e.g., X, represent matrices or sets. X_{ij} is the (i, j)th entry of X and $X_{i:}$ denotes the ith row of X. Lower case letters, e.g., x, represent vectors or scale values, and x_i is the ith element of vector x. Superscript (i), e.g., $X^{(i)}$ and $x^{(i)}$,

represents datum from the ith view. Throughout this paper, I_c denotes the $c \times c$ identity matrix. $\|X\|_F$ denotes the Frobenius norm of matrix X and for an arbitrary matrix $X \in R^{p\times q}$, its $l_{2,1}$-norm is defined as $\|X\|_{2,1} = \sum_{i=1}^{p} \sqrt{\sum_{j=1}^{q} X_{ij}^2}$.

2.2 The Objective Function

Given a centered multi-view data set which consists of n objects from m views, we denote this set as $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$, wherein $x_i = [(x_i^{(1)})^T, (x_i^{(2)})^T, \ldots, (x_i^{(m)})^T]^T \in R^{(\sum_{v=1}^{m} d_v)\times 1}$ is the ith multi-view datum and $x_i^{(v)} \in R^{d_v\times 1}$ is its vth view feature. Thus, the feature data matrix of vth view and all views can be denoted as $X^{(v)} = [x_1^{(v)}, x_2^{(v)}, \ldots, x_n^{(v)}]^T$ and $X = [X^{(1)}, X^{(2)}, \ldots, X^{(m)}]^T \in R^{d\times n}$ respectively, wherein $d = \sum_{v=1}^{m} d_v$. To select the compact and relevant feature subset, we argue that the utilization of three kinds of vital information, which are data cluster structure, data similarity and the correlations between different views, can boost the performance. The reason is that the first one reflects the discriminative information contained in different clusters, the second one holds the data geometric structure in the original high dimensional feature space and the third one may enhance or correct the weak views. Meanwhile,

Now, we first elaborate on how to utilize the data cluster structure information and define a scaled data cluster label matrix $F = [f_1, \ldots, f_n]^T \in R^{n\times c}$, which can be regarded as pseudo class labels, wherein c is the data cluster number and f_i is the estimated label of $x_i \in X$ by a prediction function $p(x)$. Clearly, F represents the discriminative information of the data. Hence we now encounter a problem: how to construct or learn the prediction function $p(x)$? By assuming F is available, to learn $p(x)$ based on F, a reasonable choice is to minimize the total prediction error of $p(x)$ with respect to F over all data samples:

$$\min \sum_{i=1}^{n} loss(p(x_i), f_i). \tag{1}$$

In [13], the authors implicitly assumed that there is a "hard" linear transformation between features and pseudo labels, i.e., $p(x_i) = W^T x_i$. However, this transformation is likely to be nonlinear in real-world applications [20]. To mitigate this problem, an explicitly "soft" linear constrained transformation has been adopted in [19] by using a $l_{2,1}$-norm regularized least square loss function, which can be rewritten in a matrix form:

$$\min \|X^T W - F\|_F^2 + \beta\|W\|_{2,1}, \tag{2}$$

where $W \in R^{d\times c}$. We denote this loss function as LS_L21. Thus, the relationship between data features and data cluster labels are specified by Eq.(2). More importantly, the data cluster structure information has utilized via F which reflects the discriminative information of the data.

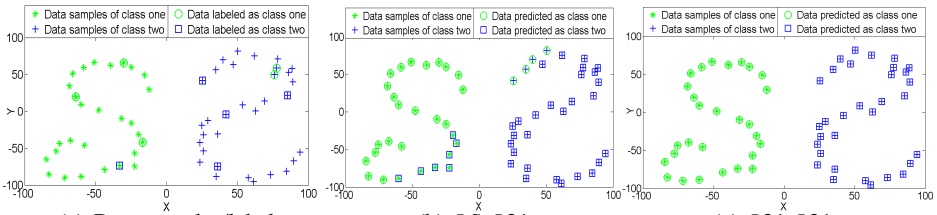
Because the error of each data sample used in Eq.(2) is squared residue error in the form of $\|W^T x_i - f_i\|^2$, a few outliers with large errors can easily dominate the objective function. Therefore, the above loss function is well-known to be unstable w.r.t. noise and outliers [21, 22]. Unfortunately, many real-world data are likely to contain noise

and outliers. Moreover, the data cluster label matrix F is normally learned via clustering methods, and tends to contain some labeling errors. For this reason, a robust loss function for learning data cluster label prediction function is desired. Inspired by [22, 23], we assume that the mapping from data features to data labels can be approximated by a robust sparse regression model with the $l_{2,1}$-norm penalty, which can be formulated as:

$$\min \|X^T W - F\|_{2,1} + \beta \|W\|_{2,1}. \tag{3}$$

In this robust formulation, we replace the Frobenius norm on the regression term with a $l_{2,1}$-norm, which brings twofold benefits: 1) since the residue error has changed to be not squared, the large errors due to outliers do not dominate the loss function; 2) the $l_{2,1}$-norm constraint results in row sparseness property, which is consistent with the ideal feature selection matrix W. We denote this improved robust loss function as $L21\_L21$.

We use a 2D toy data experiment to illustrate the robustness of $L21\_L21$ in Fig.2. In this experiment, two classes of artificial data samples are generated. Ten randomly selected data samples are assigned labels, wherein three samples contain the error labels. We use these labeled data samples to train LS_L21 and $L21\_L21$ respectively and then use the learned W to predict cluster labels for all data. To make the comparison fair , we tune β from $\{10^{-4}, 10^{-2}, 1, 10^2, 10^4\}$ and report the best prediction results for each model. From Fig. 2(c) and Fig.2(b), we observe that $L21\_L21$ is much more robuster than LS_L21.



(a) Data samples/labels (b) LS_L21 (c) $L21\_L21$

Fig. 2. 2D toy data for label prediction. (a) shows the original data samples and the selected labeled samples. (b) and (c) show the prediction results by LS_L21 and $L21\_L21$ respectively.

Recent studies [7, 16, 17, 24] have shown that in many practical applications, data samples lie on a low-dimensional manifold embedding in a high dimensional abient space. Hence, it is necessary to consider the data similarity or data geometric structure in feature selection. For any particular vth view data $X^{(v)}$, a view-depended visual similar graph $A^{(v)}$ is constructed according to the vth view features, whose element $A_{ij}^{(v)}$ reflects the visual similarity between the two features $x_i^{(v)}$ and $x_j^{(v)}$. There exist two popular ways for the graph construction: one is the k-nearest-neighbor method, and the other is the ϵ-ball based method. To reduce the number of parameters, we adopt the former one and define $A^{(v)}$ as follows:

$$A_{ij}^{(v)} = \begin{cases} 1 \text{ if } x_i^{(v)} \text{ is among the } k\text{-nearest-neighbors of } x_j^{(v)} \text{ and vice versa,} \\ 0 \text{ otherwise.} \end{cases}$$

To preserve the data geometric structure, it is essential to preserve the local consistency that similar data should have high probability to be clustered into the same class. To achieve this goal, we minimize the following objective function for the vth view:

$$\begin{aligned} &\min \frac{1}{2} \sum_{l=1}^{c} \sum_{i,j=1}^{n} (F_{il} - F_{jl})^2 A_{ij}^{(v)} \\ &s.t. \quad F^T F = I_c. \end{aligned} \tag{4}$$

Note that

$$\begin{aligned} \sum_{l=1}^{c} \sum_{i,j=1}^{n} (F_{il} - F_{jl})^2 A_{ij}^{(v)} &= \sum_{i,j=1}^{n} A_{ij}^{(v)} (f_i^T f_i + f_j^T f_j - 2 f_i^T f_j) \\ &= 2tr(F^T (D^{(v)} - A^{(v)}) F) = 2tr(F^T L^{(v)} F), \end{aligned} \tag{5}$$

where $tr(\cdot)$ denotes the trace operator, $D^{(v)}$ is a diagonal matrix with $D_{ii}^{(v)} = \sum_{j=1}^{n} A_{ij}^{(v)}$, and $L^{(v)} = D^{(v)} - A^{(v)}$ is the geometric laplacian matrix. Thus, Eq.(4) can be reformulated as:

$$\begin{aligned} &\min tr(F^T L^{(v)} F) \\ &s.t. \quad F^T F = I_c. \end{aligned} \tag{6}$$

Because different views of features characterize different aspects of the objects, the intrinsic difference of each view leads to different contribution to the final recognition results [7]. Meanwhile, the underlying correlated and complemental information between different views may be exploited to enhance or correct the weak views. Thus, we are motivated to exploit these information. By combining all of the view-dependent geometric laplacian matrices using an adaptive non-negative view-weight vector $\lambda = [\lambda_1, \ldots, \lambda_m]^T \in R^{m \times 1}$, we obtain the adaptive multi-view visual similar graph learning objective function as follows:

$$\begin{aligned} &\min_{F,\lambda} \sum_{v=1}^{m} \lambda_v tr(F^T L^{(v)} F) = \min tr(F^T \sum_{v=1}^{m} \lambda_v L^{(v)} F) \\ &s.t. \quad F^T F = I_c, \sum_{v=1}^{m} \lambda_v = 1, \lambda_v \geq 0. \end{aligned} \tag{7}$$

Also, the intrinsic discriminative ability of each view is revealed by λ.

To leverage aforementioned three kinds of vital information simultaneously, we integrate data cluster labels prediction and adaptive multi-view visual similar graph learning into a unified framework:

$$\begin{aligned} &\min_{F,\lambda,W} tr(F^T \sum_{v=1}^{m} \lambda_v L^{(v)} F) + \alpha \|X^T W - F\|_{2,1} + \beta \|W\|_{2,1} \\ &s.t. \quad F^T F = I_c, \sum_{v=1}^{m} \lambda_v = 1, \lambda_v \geq 0. \end{aligned} \tag{8}$$

However, there still remains two issues of Eq.(8). The first one is regarding the sign of F. Like most clustering algorithms, we impose the orthogonal constraint on F. While it is still likely to have mixed signs in the final result of F and it may severely deviate from the ideal solution that only 0 and 1 are contained in F. Moreover, since F is defined as the data cluster label matrix, negative entries in F not only are lack of clear physical

meaning but also make it difficult to assign the cluster labels. Recently, Yang *et al.*[25] declared explicitly imposing non-negative constraint on F would make result much closer to the idea solution. So, it is necessary to impose non-negative constraint on F. The second one is why we adopt the linear weight λ on each view? In fact, the solution of λ in Eq.(7) is $\lambda_v = 1$ corresponding to the minimum $tr(F^T L^{(v)} F)$ over different views, and other entries in λ equal to 0. It means that only one view is selected by this method. To handle this problem, we adopt a trick utilized in [7, 24], i.e., we set $\lambda_v \leftarrow \lambda_v^r$ with $r > 1$. Thus, the improved objective function of AUMFS can be formulated as:

$$\begin{aligned} &\min_{F,\lambda,W} tr(F^T \textstyle\sum_{v=1}^m \lambda_v^r L^{(v)} F) + \alpha \|X^T W - F\|_{2,1} + \beta \|W\|_{2,1} \\ &s.t. \quad F^T F = I_c, F \geq 0, \textstyle\sum_{v=1}^m \lambda_v = 1, \lambda_v \geq 0. \end{aligned} \tag{9}$$

From the definition of the $l_{2,1}$-norm, we can see that when the penalty β increases, many rows of W will shrink (or be closer) to zeros. Consequently, for a datum x, $\tilde{x} = W^T x$ can be treated as a new representation after feature selection wherein only the most relevant feature subset remains. In other words, we can rank all feature components $c_i|_{i=1}^d$ according to the $\|W_{i:}\|$ in descending order and select top ranked components in a batch mode.

2.3 Optimization Method

Clearly, Eq.(9) is a nonlinearly constrained nonconvex optimization problem. In the following, we introduce an iterative approach based on coordinate descent to solve it. Firstly, we relax the orthogonal constraint by adding a large enough penalty term $\gamma \|F^T F - I_c\|_F^2$ (e.g., $\gamma = 10^8$ in our experiment.) and rewrite it as follows:

$$\begin{aligned} &\min_{F,\lambda,W} tr(F^T \textstyle\sum_{v=1}^m \lambda_v^r L^{(v)} F) + \alpha \|X^T W - F\|_{2,1} + \beta \|W\|_{2,1} + \gamma \|F^T F - I_c\|_F^2 \\ &s.t. \quad F \geq 0, \textstyle\sum_{v=1}^m \lambda_v = 1, \lambda_v \geq 0. \end{aligned} \tag{10}$$

Let $\mathcal{J}$ denote the objective function in Eq.(10). We initialize $\lambda_v = \frac{1}{m}$, set F using the clustering result obtained by K-means and set W with a random matrix. Then, we iteratively update W, F and λ individually, while holding the other variables constant.

Optimize W for fixed F and λ For the fixed F and λ, the part of $\mathcal{J}$ that involves W is

$$\min_W \alpha \|X^T W - F\|_{2,1} + \beta \|W\|_{2,1}, \tag{11}$$

which is further equivalent to [23]:

$$\min_{W,E} \|E\|_{2,1} + \|W\|_{2,1}, \quad s.t. \quad X^T W + \frac{\beta}{\alpha} E = F. \tag{12}$$

Rewriting Eq.(12) as:

$$\min_{W,E} = \| \begin{bmatrix} W \\ E \end{bmatrix} \|, \quad s.t. \quad \begin{bmatrix} X^T & \frac{\beta}{\alpha} I_n \end{bmatrix} \begin{bmatrix} W \\ E \end{bmatrix} = F. \tag{13}$$

Let $B = \begin{bmatrix} X^T & \frac{\beta}{\alpha} I \end{bmatrix}$ and $U = \begin{bmatrix} W \\ E \end{bmatrix}$, then Eq.(13) is equivalent to:

$$\min_U \|U\|_{2,1} \quad s.t. \quad BU = F. \tag{14}$$

We introduce the Lagrange multiplier $\psi \in R^{n \times c}$ and the Lagrange function is:

$$\mathscr{L}(U, \psi) = \|U\|_{2,1} - tr(\psi^T (BU - F)). \tag{15}$$

Setting $\frac{\partial \mathscr{L}(U,\psi)}{\partial U} = 0$, we obtain:

$$\frac{\partial \mathscr{L}(U, \psi)}{\partial U} = 2PU - B^T \psi = 0, \tag{16}$$

where P is a diagonal matrix with the ith diagonal element as $P_{ii} = \frac{1}{2\|U_{i:}\|_2}$ [2]. Left multiplying the two sides of Eq.(16) by BP^{-1}, and using the constraint $BU = F$, we have [23]:

$$\begin{aligned} &2BU - BP^{-1}B^T\psi = 0 \Rightarrow 2F - BP^{-1}B^T\psi = 0 \\ &\Rightarrow \psi = 2(BP^{-1}B^T)^{-1}F. \end{aligned} \tag{17}$$

By substituting Eq.(17) into Eq.(16), we obtain:

$$U = P^{-1}B^T(BP^{-1}B^T)^{-1}F. \tag{18}$$

Since Eq.(14) is a convex problem, U is a global optimum solution to the problem if and only if Eq.(18) is satisfied. Therefore, if we iteratively update U and the corresponding P, we can get the global optimum solution of U. Because of $U = \begin{bmatrix} W \\ E \end{bmatrix}$, W can also be directly obtained from U.

Optimize F for fixed W and λ Similarly, given W and λ, we update F to decrease the value of $\mathcal{J}$. Let $L = \sum_{v=1}^{m} \lambda_v^r L^{(v)}$, then $\mathcal{J}$ becomes:

$$\min_F tr(F^T L F) + \alpha \|X^T W - F\|_{2,1} + \gamma \|F^T F - I_c\|_F^2 \quad s.t. \quad F \geq 0. \tag{19}$$

Since $F \geq 0$, we introduce the Lagrange multiplier $\Phi \in R^{n \times c} \geq 0$, thus, the Lagrange function is:

$$\mathscr{L}(F, \Phi) = tr(F^T L F) + \alpha \|X^T W - F\|_{2,1} + \gamma \|F^T F - I_c\|_F^2 - tr(\Phi^T F). \tag{20}$$

Setting $\frac{\partial \mathscr{L}(F,\Phi)}{\partial F} = 0$, we get:

$$\begin{aligned} \frac{\partial \mathscr{L}(F, \Phi)}{\partial F} &= 2LF + 2\alpha Q(X^T W - F) + 4\gamma F(F^T F - I_c) - \Phi = 0 \\ &\Rightarrow \Phi = 2LF + 2\alpha Q(X^T W - F) + 4\gamma F(F^T F - I_c), \end{aligned} \tag{21}$$

[2] In practice, $\|U_{i:}\|_2$ could be close to zero but not zero. When $\|U_{i:}\|_2 = 0$,$P_{ii} = 0$ is a subgradient of $\|U\|_{2,1}$. However, we can not set $P_{ii} = 0$ when $\|U_{i:}\|_2 = 0$, otherwise the derived algorithm for updating U can not be guaranteed to converge [23].So, we regularize $P_{ii} = \frac{1}{2\sqrt{U_{i:}^T U_{i:} + \epsilon}}$, where ϵ is a very small constant.

where Q is a diagonal matrix with the ith diagonal element as $Q_{ii} = \frac{1}{2\|(X^T W - F)_{i:}\|_2}$. Using the Karush-Kuhn-Tucker condition [26] $\Phi_{ij} F_{ij} = 0$, we have:

$$(LF + \alpha Q(X^T W - F) + 2\gamma F(F^T F - I_c))_{ij} F_{ij} = 0. \tag{22}$$

Eq.(22) leads to the following updating formula:

$$F_{ij} = F_{ij} \frac{(\alpha QF + 2\gamma F)_{ij}}{(LF + \alpha QX^T W + 2\gamma FF^T F)_{ij}}. \tag{23}$$

Finally, we normalize F such that $(F^T F)_{ii} = 1, i = 1, ..., c$.

Optimize λ for fixed F and W Now, we decrease the objective function with respect to λ given F and W. The objective function $\mathcal{J}$ degenerates into the following equation:

$$\min_{\lambda} tr(F^F \sum_{v=1}^{m} \lambda_v^r L^{(v)} F) \quad s.t. \sum_{v=1}^{m} \lambda_v = 1, \lambda_v \geq 0. \tag{24}$$

By using a Lagrange multiplier ξ to take the constraint $\sum_{v=1}^{m} \lambda_v = 1$ into consideration, we get the Lagrange function as follows [7]:

$$\mathscr{L}(\lambda, \xi) = tr(F^F \sum_{v=1}^{m} \lambda_v^r L^{(v)} F) - \xi(\sum_{v=1}^{m} \lambda_v - 1). \tag{25}$$

Setting the derivative of $\mathscr{L}(\lambda, \xi)$ with respect to λ_v and ξ to zero, we have:

$$\begin{cases} \frac{\partial \mathscr{L}(\lambda,\xi)}{\partial \lambda_v} = r\lambda_v^{r-1} tr(F^T L^{(v)} F) - \xi = 0, \\ \frac{\partial \mathscr{L}(\lambda,\xi)}{\partial \xi} = \sum_{v=1}^{m} \lambda_v - 1 = 0. \end{cases} \tag{26}$$

Solving Eq.(26), the updating formula for λ_v can be obtained:

$$\lambda_v = \frac{(1/tr(F^T L^{(v)} F))^{1/(r-1)}}{\sum_{v=1}^{m} (1/tr(F^T L^{(v)} F))^{1/(r-1)}}. \tag{27}$$

The updating rules for F, W and λ should be recursively applied until the convergence is achieved. Then, local optimum solution of F,W and λ to the objective function $\mathcal{J}$ can be obtained. Following the aforementioned feature selection rule with respect to W, the desired compact and relevant feature subset can be selected. Due to the space limitation, the convergence proof of the proposed optimization method is omitted. Similar proof can be found in [7, 13, 19, 25].

3 Experiments

In this section, we evaluate the performance of AUMFS by applying it to three visual concept recognition applications, including social image concept recognition, object recognition and video-based human action recognition, on four public datasets.

3.1 Experiment Setup

Two image datasets (i.e., Corel5K [27] and NUS-WIDE-OBJECT [28]) and two video datasets (i.e., Weizmann [29] and KTH [30]) are used in our experiments. For image datasets, we extract five types of features, i.e., color histogram (64d), color auto-correlogram (144D), edge direction histogram (73D), wavelet texture (128D) and block-wise color moments (225D), following [28]. For video datasets, holistic features and space-time local features are combined in these experiments. We use the frame differencing to compute holistic features, avoiding background substraction and object tracking [2]. Based on single differencing frames and motion energy images, two kinds of holistic features are computed with Zernike moments and Hu moments respectively. Because the single differencing frame contains spatial pattern of an action and the motion energy image represents the temporal pattern of it, both spatial and temporal patterns are considered. We adopt stip [31] and mosift [32] features as the space-time local features. After extracting all of these features from all videos, k-means clustering is applied to product the bag-of-words descriptor for each kind of features. Then, each kind of moment features and space-time local features of an video are quantized into 50 dimensional and 500 dimensional features respectively. In other words, each video is represented by six quantized feature vectors. Table 1 summarizes the detailed information of these datasets used in our experiments. We compare the performance of

Table 1. Data set description

Name	Size	# of concept	# of feature	Data Type
Corel5K	5000	50	64+144+73+128+225= 634	Image
NUS-WIDE-OBJECT	30000	31	64+144+73+128+225= 634	Image
Weizmann	90	10	$50 \times 4 + 500 \times 2 = 1200$	Video
KTH	600	6	$50 \times 4 + 500 \times 2 = 1200$	Video

AUMFS with six state-of-the-art unsupervised feature selection methods: All Features (A baseline where all features are used for recognition), Max Variance, Laplacian Score [16], Feature Ranking [17], Multi-Cluster Feature Selection (MCFS) [18] and Nonnegative Discriminative Feature Selection (NDFS) [19]. For image datasets, we randomly select 1000 samples as training data and the remainder samples are served as testing data. For video datasets, we perform leave-one-out cross-validation to evaluate all the methods. In particular, leave-one-out cross-validation selects one subject as testing data and uses the remainder subjects to train models. To fairly compare all the methods, we fix the nearest neighborhood value k to 5 for graph-based methods such as Laplacian Score, MCFS, NDFS and AUMFS. For AUMFS, we empirically set γ to 10^8 and r to 4 and tune α and β from $\{10^{-5}, 10^{-3}, 10^{-1}, 1, 10, 10^3, 10^5\}$. Because the dimensionality of image data is 634 and video data is 1200, we set the selected feature dimensionality as $\{20 : 20 : 600\}$ for image datasets and $\{100 : 100 : 1200\}$ for video datasets. For the other algorithms, we tune the algorithm dependent parameters and adopt the best setting. Also, we repeat all the experiments 10 times in the image datasets and due to using the leave-one-out rule, experiments on Weizmann and KTH have repeated 9 and 25 times. The average accuracy results are reported for all the algorithms.

We have reason to believe that good features should yield high recognition accuracy. To exclude the factor of classifiers, we use the nearest neighbor classifier (NCC)

and SVM to evaluate the performance. NNC is a simple and non-parameter classifier, while SVM is a robust and sophisticated classifier. For the image dataset, we use SVM with the RBF kernel and tune both of its parameters C and γ in the range of $[2^{-5}, 2^{-4}, \ldots, 2^4, 2^5]$ using 5-fold cross-validation to select the best parameters. Because the dimension of video features is very high while the number of instances is small, especially in Weizmann dataset, the linear kernel is adopted for video datasets following [33] and it worked really well in practice.

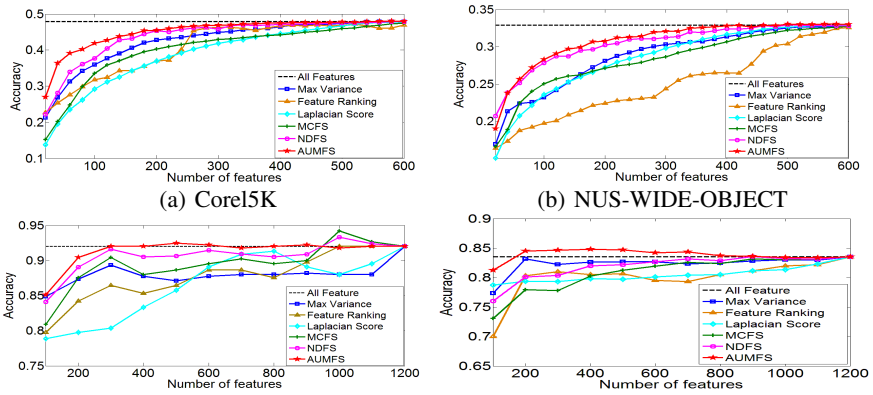
In addition to the comparison of different algorithms, we also study the sensitiveness of parameters and the convergence of AUMFS. We have evaluated α, β and r except γ which has already been empirically fixed to 10^8. As declared in [7] that the optimal value of r is data set dependent. Therefore, we tune r from $\{2, 4, 6, 8, 10\}$ and randomly select 10,20,30 and 50 concepts from Corel5K dataset to construct dataset with different size for testing r. Also, the selected feature number has fixed to 300 and 600 for image datasets and video datasets respectively in these parameter-related experiments.

3.2 Experimental Results

Figure 3 and Fig.4 show the recognition results of different algorithms on four data sets. It is clear that our method almost consistently outperforms other methods on these four data sets, especially when the number of selected feature is relatively low, which is likely to be the operational point in practices. The superiority of our method may arise in the twofold: 1) AUMFS simultaneously leverages the underlying three kinds of vital information,i.e., data cluster structure, data similarity and the correlations between different views; 2) the no-negative constraint on F and the $l_{2,1}$-norm constraint on the loss function $L21\_L21$ are incorporated in the final objective function, making more reasonable and the robust formulation which contributes to more faithful results. Besides, AUMFS performs better than using all features in the video datasets. Meanwhile, we notice that although feature selection methods can not guarantee to consistently improve the performance in some cases, they can achieve almost the same recognition accuracy while merely using less than 50% of the original features. From Fig. 5 and Fig. 6, we can see that AUMFS is not very sensitive to the parameters α, β and r. In Fig.6(b), the optimal value of r is 8, 6, 8 and 2 with respect to 10, 20, 30 and 50 concepts respectively, which demonstrates the conclusion given by [7]: the optimal value of r is dataset dependent. Figure 7 shows the convergence curves of AUMFS over four datasets. It provides empirical evidences on the convergency of AUMFS. We observe that the proposed optimization method for AUMFS always converges very fast, well within 100 iterations.

4 Conclusion

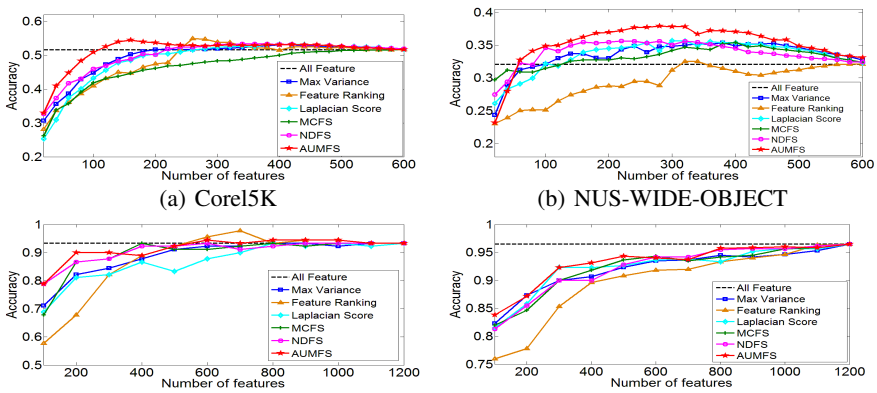
In this paper, we propose an unsupervised learning method, called Adaptive Unsupervised Multi-view Feature Selection (AUMFS) to handle multi-view feature selection problem in the unsupervised learning scenario. An efficient iterative optimization method for AUMFS is also proposed. Experiments with three visual concept recognition applications demonstrate the advantages of our method. More importantly, empir-



(a) Corel5K (b) NUS-WIDE-OBJECT

(c) Weizmann (d) KTH

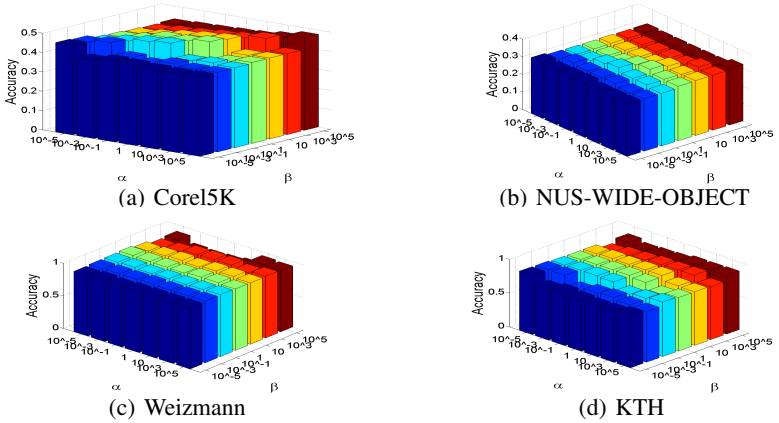
Fig. 3. Performance comparison of different algorithms on four popular datasets using NNC.



(a) Corel5K (b) NUS-WIDE-OBJECT

(c) Weizmann (d) KTH

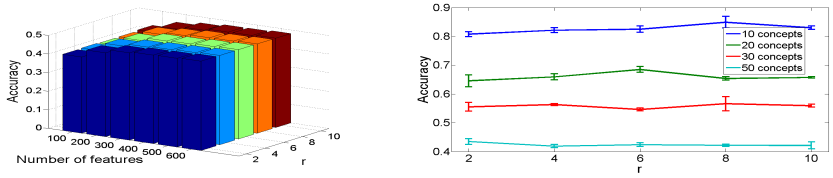
Fig. 4. Performance comparison of different algorithms on four popular datasets using SVM.



(a) Corel5K (b) NUS-WIDE-OBJECT

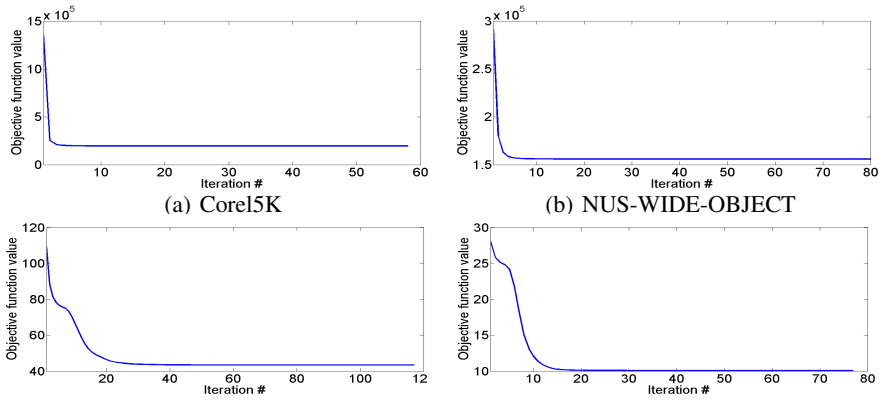
(c) Weizmann (d) KTH

Fig. 5. Performance variations of AUMFS on four datasets with fixed $r = 4$ and different α and β.



(a) Corel5K (b) Different size of Corel5K dataset

Fig. 6. Performance variations of AUMFS with fixed $\alpha = 10, \beta = 10$ and different parameter r.



(a) Corel5K (b) NUS-WIDE-OBJECT

(c) Weizmann (d) KTH

Fig. 7. Illustration of the convergence of AUMFS on four datasets.

ical results show that AUMFS is not very sensitive to the parameters and the corresponding optimization method converges very fast, which suggest that our method can be applied to a wide range of practical problems.

Acknowledgements

This research is supported by the National Key Basic Research Program (973) of China (No. 2012*CB*316400), the National Natural Science Foundation of China (No. 60903134) and the National High Technology Research and Development Program (863) of China (No. 2012*AA*011502).

References

1. Lisin, D., Mattar, M., Blaschko, M., Learned-Miller, E., Benfield, M.: Combining local and global image features for object class recognition. In: IEEE Workshop on Learning in CVPR. (2005)
2. Sun, X., Chen, M., Hauptmann, A.: Action recognition via local descriptors and holistic features. In: CVPR workshops, IEEE (2009) 58–65
3. Cao, L., Tian, Y., Liu, Z., Yao, B., Zhang, Z., Huang, T.S.: Action detection using multiple spatial-temporal interest point features. In: ICME. (2010) 340–345
4. Lowe, D.: Distinctive image features from scale-invariant keypoints. J. of CV **60** (2004) 91–110

5. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: Application to face recognition. J. of TPAMI **28** (2006) 2037–2041
6. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. J. of TPAMI **27** (2005) 1615–1630
7. Xia, T., Tao, D., Mei, T., Zhang, Y.: Multiview spectral embedding. **40** (2010) 1438–1446
8. Bickel, S., Scheffer, T.: Multi-view clustering. In: ICDM. (2004) 19–26
9. Okun, O., Priisalu, H.: Multiple views in ensembles of nearest neighbor classifiers. In: ICML Workshop on Learning with Multiple Views. (2005)
10. Lian, H., Lu, B.: Multi-view gender classification using multi-resolution local binary patterns and support vector machines. J. of IJNS **17** (2007) 479–487
11. Chaudhuri, K., Kakade, S., Livescu, K., Sridharan, K.: Multi-view clustering via canonical correlation analysis. In: ICML. (2009) 129–136
12. Fu, Y., Huang, T.: Graph embedded analysis for head pose estimation. In: FG. (2006) 3–8
13. Yang, Y., Shen, H., Ma, Z., Huang, Z., Zhou, X.: L21-norm regularized discriminative feature selection for unsupervised learning. In: IJCAI. (2011)
14. Duda, R., Hart, P., Stork, D.: Pattern Classification, 2ed. Chichester: Wiley-Interscience (2001)
15. Zhao, Z., Wang, L., Liu, H.: Efficient spectral feature selection with minimum redundancy. In: AAAI. (2010)
16. He, X., Cai, D., Niyogi, P.: Laplacian score for feature selection. In: NIPS. (2005)
17. Zhao, Z., Liu, H.: Spectral feature selection for supervised and unsupervised learning. In: ICML. (2007) 1151–1157
18. Cai, D., Zhang, C., He, X.: Unsupervised feature selection for multi-cluster data. In: KDD. (2010) 333–342
19. Zechao, L., Yi, Y., Jing, L., Xiaofang, Z., Hanqing, L.: Unsupervised feature selection using nonegative spectral analysis. In: AAAI. (2012)
20. Shi, J., Malik, J.: Normalized cuts and image segmentation. J. of TPAMI **22** (2000) 888–905
21. Piepel, G.: Robust regression and outlier detection. Technometrics **31** (1989) 260–261
22. Kong, D., Ding, C., Huang, H.: Robust nonnegative matrix factorization using l_{21}-norm. In: CIKM. (2011) 673–682
23. Nie, F., Huang, H., Cai, X., Ding, C.: Efficient and robust feature selection via joint $l_{2,1}$-norms minimization. In: NIPS. Volume 23. (2010) 1813–1821
24. M.Wang, X.S.Hua, X.Y.Y., L.R.Dai: Optimizing multi-graph learning: Towards a unified video annotation scheme. In: ACM MM. (2007) 862–870
25. Yi Yang, Heng Tao Shen, F.N.R.J., Zhou, X.: Nonnegative spectral clustering with discriminative regularization. In: AAAI. (2011)
26. Boyd, S., Vandenberghe, L.: Convex optimization. Cambridge Univ Pr (2004)
27. Duygulu, P., Barnard, K., De Freitas, J., Forsyth, D.: Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In: ECCV. Volume 4., Springer (2002) 97–112
28. Chua, T.S., Tang, J., Hong, R., Li, H., Luo, Z., Zheng, Y.T.: Nus-wide: A real-world web image database from national university of singapore. In: CIVR. (2009)
29. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: ICCV. (2005) 1395–1402
30. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: A local `SVM` approach. In: ICPR. Volume 3., IEEE (2004) 32–36
31. Laptev, I.: On space-time interest points. J. of IJCV **64** (2005) 107–123
32. Chen, M., Hauptmann, A.: Mosift: Recognizing human actions in surveillance videos. In: CMU-CS-09-161. (2009)
33. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. J. of ACM TIST **2** (2011) 1–27