Semi-Supervised Feature Selection via Spline Regression for Video Semantic Recognition

Yahong Han, Yi Yang, Yan Yan, Zhigang Ma, Nicu Sebe, *Senior Member, IEEE,* Xiaofang Zhou, *Senior Member, IEEE*

***Abstract*—In order to improve both the efficiency and accuracy of video semantic recognition, we can perform feature selection on the extracted video features to select a subset of features from the high dimensional feature set for a compact and accurate video data representation. Provided the number of labeled videos is small, supervised feature selection could fail to identify the relevant features that are discriminative to target classes. In many applications, abundant un-labeled videos are easily accessible. This motivates us to develop semi-supervised feature selection algorithms to better identify the relevant video features, which are discriminative to target classes by effectively exploiting the information underlying the huge amount of un-labeled video data. In this paper, we propose a framework of video semantic recognition by Semi-Supervised Feature Selection via Spline Regression (S^{2}FS^{2}R). Two scatter matrices are combined to capture both the discriminative information and the local geometry structure of labeled and un-labeled training videos: A within-class scatter matrix encoding discriminative information of labeled training videos and a spline scatter output from a local spline regression encoding data distribution. An $\ell_{2,1}$-norm is imposed as a regularization term on the transformation matrix to ensure it is sparse in rows, making it particularly suitable for feature selection. To efficiently solve S^{2}FS^{2}R, we develop an iterative algorithm and prove its convergency. In the experiments, three typical tasks of video semantic recognition, namely video concept detection, video classification, and human action recognition, are used to demonstrate that the proposed S^{2}FS^{2}R achieves better performance compared with the state-of-the-art methods.**

***Index Terms*—Video Analysis, Semi-Supervised Feature Selection, Spline Regression, $\ell_{2,1}$-norm.**

I. Introduction

In many applications of video semantic recognition, such as video concept detection [1], [2], human activity analysis [3], [4], and object tracking [5], [6], data are always represented by high dimensional feature vectors. For example, we can extract high dimensional heterogeneous visual features from one given video key frame, such as global features (color moment, edges direction, and Gabor) and local features (space-time interest points [7] and MoSIFT [8]). In the high dimensional space of visual features, it is hard to discriminate video samples of different classes from each other, which results in the so called “curse of dimensionality” problem [9]. Moreover, in the presence of many irrelevant features, the training process of classification tends to overfitting. This paper explores feature selection and its applications to video semantic recognition.

Feature selection has a twofold role in improving both the efficiency and accuracy of data analysis. First, the dimensionality of selected feature subset is much lower, making the subsequential computation on the input data more efficient. Second, the noisy features are eliminated for a better data representation, resulting in a more accurate classification result. Therefore, during recent years feature selection has attracted much research attention [1], [4], [10], [11], [12], [13]. In video semantic recognition, feature selection is usually applied for a higher classification accuracy and a compact feature representation [1], [4], [10], [6].

Feature selection algorithms can be roughly classified into two groups, i.e., supervised feature selection and unsupervised feature selection. Supervised feature selection determines feature relevance by evaluating a feature’s correlation with the classes [14], [15], [16]. Because discriminative information is enclosed in the labels, supervised feature selection is usually able to select discriminative features. Without labels, unsupervised feature selection exploits data variance and separability to evaluate feature relevance. A frequently used criterion is to select the features which best preserve the data distribution or local structure derived from the whole feature set [17]. However, because there is no label information directly available, it is much more difficult for unsupervised feature selection to select the discriminative features [10].

In real-world applications, collecting high-quality labeled training videos is difficult, and at the same time abundant un-labeled videos are often easily accessible. Provided the number of labeled data is small, supervised feature selection could fail to identify the relevant features that are discriminative to target classes. This motivates us to develop semi-supervised feature selection algorithms to better identify the relevant features. In order to use both labeled and un-labeled data, inspired by the semi-supervised learning algorithms [18], [19], semi-supervised feature selection algorithms utilize the data distribution or local structure of both labeled and un-labeled

This work is partially supported by NSFC (Grant No. 61202166) and Doctoral Fund of Ministry of Education of China (Grant No. 20120032120042). Dr. Yi Yang is partially supported by the DECRA project (Grant No. DE130101311). Prof. Xiaofang Zhou is partially supported by NSFC (Grant No. 61232006) and the Australian Research Council (Grant No. DP140103171)

Yahong Han is with the School of Computer Science and Technology, and the Tianjin Key Laboratory of Cognitive Computing and Application, Tianjin University, China, e-mail: yahong@tju.edu.cn

Yi Yang and Yan Yan are with the School of Information Technology and Electrical Engineering, The University of Queensland, Australia, e-mail: yee.i.yang@gmail.com, yanyan.tju@gmail.com

Zhigang Ma is with the School of Computer Science, Carnegie Mellon University, USA, e-mail: kevinma@cs.cmu.edu

Nicu Sebe is with the Department of Information Engineering and Computer Science, University of Trento, Italy, email: sebe@disi.unitn.it

Xiaofang Zhou is with the School of Information Technology and Electrical Engineering, The University of Queensland, Australia, and the School of Computer Science and Technology, Soochow University, China, email: zxf@itee.uq.edu.au

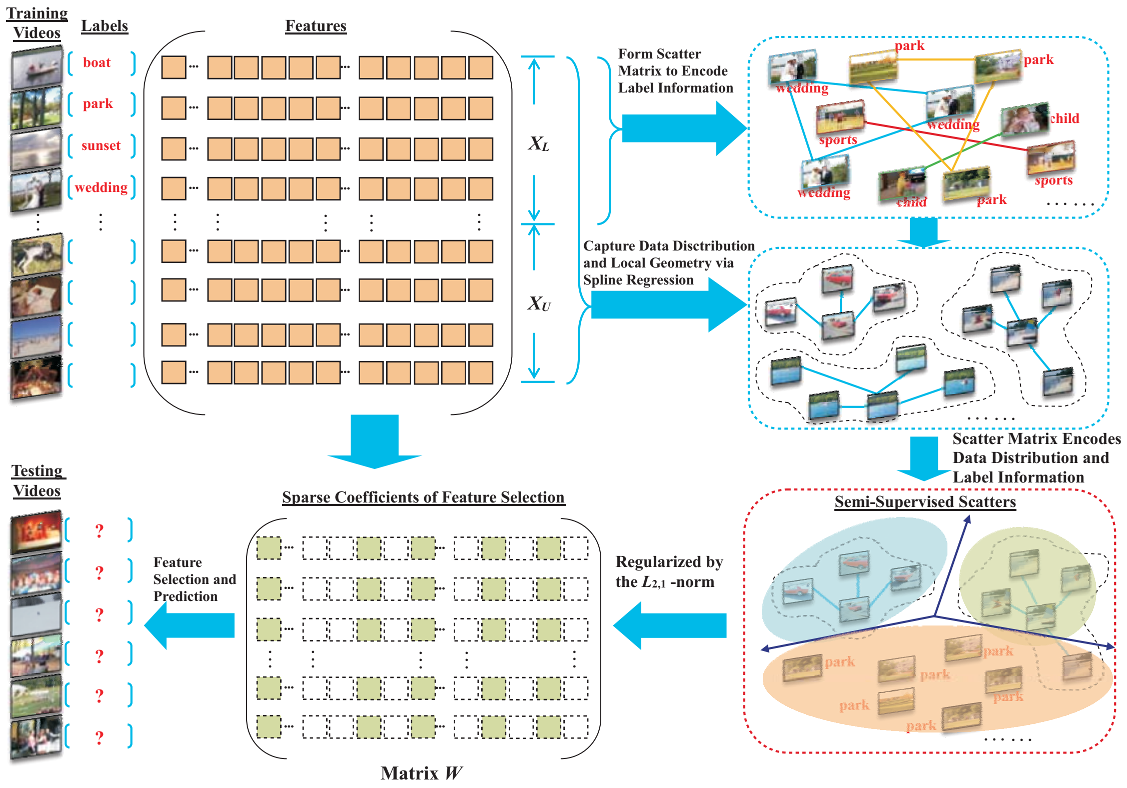

Training Videos
Labels
Features
boat
park
sunset
wedding
X_L
X_U
Form Scatter Matrix to Encode Label Information
park
park
wedding
wedding
child
sports
wedding
child
park
sports
......
Capture Data Distribution and Local Geometry via Spline Regression
......
Scatter Matrix Encodes Data Distribution and Label Information
Semi-Supervised Scatters
park
park
park
park
park
park
park
......
Regularized by the $L_{2,1}$ -norm
Sparse Coefficients of Feature Selection
Feature Selection and Prediction
Matrix W
Testing Videos
?
?
?
?
?
?

Fig. 1. Flowchart of the proposed framework S^{2}FS^{2}R. We first construct the within-class scatter matrix to encode label information of labeled training videos. Data distribution and local geometry structure of both labeled and un-labeled training videos are preserved by the local spline regression. Combining within-class and spline scatters, we form a semi-supervised scatter matrix to encode data distribution and label information. An $\ell_{2,1}$-norm is imposed as a regularization term on the transformation matrix W to ensure that W is sparse in rows, making it particular suitable for feature selection.

data to evaluate the features' relevance. For example, Zhao and Liu [12] introduced a semi-supervised feature selection algorithm based on spectral analysis. Spectral assumption states that the data points forming the same structure are likely to have the same label. Similarly, the method in [20] utilizes manifold regularization to consider the geometry of data distribution. In [21], Kong et al. proposed a semi-supervised feature selection algorithm for graph data. Many local evaluations are introduced to model the neighboring data points so as to explore the data structures. Typical methods include data affinity between neighbors [22] and locally linear representation [23], and locally nonlinear representation with kernels [24]. However, besides the parameter tuning problem in affinity measure with Gaussian function, the locally linear representations and kernel functions may lack the ability to accurately capture the local geometry [25].

In this paper, to better exploit the data distribution and the local geometry of both labeled and un-labeled videos, we propose a framework of Semi-Supervised Feature Selection via Spline Regression (S^{2}FS^{2}R). The flowchart of the proposed framework is illustrated in Figure 1. Both the labeled and un-labeled video data are collected as training videos. For each video sample in the training and testing video sets, we extract high-dimensional features to form the feature matrix $X = [X_L; X_U]$ of the training data. As illustrated in Figure 1, to make use of the discriminative information in the labeled videos, we form a within-class scatter matrix on the labeled training videos. To exploit the data distribution and local geometry underlying the huge amount of un-labeled videos, we use splines developed in Sobolev space [26], [25] to interpolate scattered videos in geometrical design (see the step of spline regression in Fig. 1). By integrating the polynomials and Green's functions into the local spline [27], [25], the local geometry of video data can be smoothly and accurately captured according to their distribution. By summing the local losses estimated from all of the neighboring videos, we construct a spline scatter matrix to preserve the local geometry of labeled and un-labeled video data. Thus, the local structure and geometry of all training videos are preserved in the formed spline scatter matrix. Combining within-class and spline scatters, we form a semi-supervised scatter matrix to encode data distribution and label information. Our goal is to compute a transformation matrix W (see matrix W in Fig. 1) which optimally preserves discriminative information and data distribution of training videos. To make W suitable for feature selection, we add an $\ell_{2,1}$-norm of W as a regularization term to ensure that W is sparse in rows [11], [15]. Then the learned W is able to select the most discriminant features for testing videos prediction. To efficiently solve the $\ell_{2,1}$-norm minimization problem with the orthogonal constraint, we develop an iterative algorithm and prove its convergence.

In the experiments, four open benchmark video datasets are used to evaluate the performance of video semantic recognition by Semi-Supervised Feature Selection via Spline Regression (S^{2}FS^{2}R), which correspond to three typical video semantic recognition tasks: Video concept detection in news videos, video classification of consumer videos, and human action recognition. Experimental results show that S^{2}FS^{2}R gets better performance for video semantic recognition compared with state-of-the-art algorithms.

The remainder of this paper is organized as follows. In Section II, we briefly review the recent related works. The framework of S^{2}FS^{2}R and its solutions are introduced in Section III. In Section III-D, we present an iterative algorithm to solve S^{2}FS^{2}R and prove its convergence. The experimental analysis are given in Section IV. Finally, we summarize the conclusion in Section V.

II. Related Works

In this section, we review some of the representative related works of video representation and feature selection for video semantic recognition.

A. Video Feature Representations

In applications of video classification and video concept detection, one key frame within each shot is obtained as a representative image for that shot. In this way, video shots can be represented by the extracted low-level visual features of corresponding key frames. For example, TRECVID[1] provides global features of each key frame, such as color histograms, textures, and Canny edge. With the popularity of key point based local features, e.g., SIFT feature [28], and the successful applications in scene classification [29], we can also represent each key frame using a Bag-of-Words (BoW) approach. Another important characteristic of video data is the temporal associated co-occurrence. Considering that each video frame is a two-dimensional object represented by image features, the temporal axis makes up the third dimension. Thus, a video stream spans a three-dimensional space. As discussed in [3], the SIFT feature lacks the ability of representing temporal information in videos and does not consider motion information. Recently, multi-instance space-time volumes [30], space-time interest points (STIP) [7], and MoSIFT [8] representations have been respectively proposed to model the time information of video data. In order to perform video event detection in real-world conditions, Ke et al. [30] efficiently match the volumetric representation of an event against over-segmented spatio-temporal video volumes. The STIP descriptor concatenates several histograms from a space-time grid defined on the patch and generalizes the SIFT descriptor to space-time. In contrast, MoSIFT detects interest points and not only encodes their local appearance but also explicitly models the local motion. Owing to above characteristics, STIP and MoSIFT have been widely used in motion analysis and human action recognition [3], [7], [8].

B. Feature Selection for Video Semantic Recognition

Feature selection has an important role in improving both the efficiency and accuracy of video semantic recognition. During recent years, feature selection has attracted much research attention [14], [17], [12]. However, most of the feature selection algorithms evaluate the importance of each feature individually and select features one by one. A limitation is that the correlation among features is neglected. Sparsity-based methods, e.g., lasso [31], use the ℓ_1-norm of coefficient vectors as a penalty to make many coefficients shrink to zeros, which can be used for feature selection. For example, the sparse multinomial logistic regression via Bayesian ℓ_1 regularization (SBMLR) [32] exploits sparsity by using a Laplace prior. Inspired by the block sparsity, [15] employs a joint $\ell_{2,1}$-norm minimization on both the loss function and regularization to realize feature selection across all data points. More recently, researchers have applied the two-step approach, i.e., spectral regression, to supervised and unsupervised feature selection [16]. The works in [15], [16], [33], [34] have shown that it is a better way to evaluate the importance of the selected features jointly. On the other hand, though some multiple kernel feature selection methods have been proposed for video semantic recognition [35], semi-supervised feature selection for video semantic recognition has not been well explored. In this paper, we propose a new one-step approach to perform semi-supervised feature selection by simultaneously exploiting discriminative information and preserving the local geometry of labeled and un-labeled video data.

III. Semi-Supervised Feature Selection via Spline Regression

In this section, we present the framework of Semi-Supervised Feature Selection via Spline Regression (S^{2}FS^{2}R). In order to solve this framework efficiently, we develop an iterative algorithm and prove its convergence. To better present the proposed methods, we also introduce local spline regression in this section. In the following, we first provide the notations used in the rest of this paper.

A. Notations

Let us denote $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ as the training set of videos, where $x_i \in \mathbb{R}^d (1 \leq i \leq n)$ is the i-th video sample and n is the total number of training instances. For each video sample, we extract d-dimensional video features and then the matrix of training videos can be represented by $X = [x_1, \ldots, x_n] \in \mathbb{R}^{d\times n}$. We let $X^{\mathcal{L}} = [x_1, \ldots, x_{n_l}] \in \mathbb{R}^{d\times n_l}$ denote the first n_l $(n_l \leq n)$ video samples in X which are the labeled videos, for which the labels $Y^{\mathcal{L}} = [y_1, \ldots, y_{n_l}] \in \{0,1\}^{c\times n_l}$ are provided for the c semantic categories. $X^{\mathcal{U}} = [x_{n_l+1}, \ldots, x_{n_l+n_u}] \in \mathbb{R}^{d\times n_u}$ denote the un-labeled videos whose labels are not given. Thus we have $X = [X^{\mathcal{L}}, X^{\mathcal{U}}]$ and $n = n_l + n_u$. In this paper, I is an identity matrix. For an arbitrary matrix $M \in \mathbb{R}^{r\times p}$, its $\ell_{2,1}$-norm is defined as

$$||M||_{2,1} = \sum_{i=1}^{r} \sqrt{\sum_{j=1}^{p} M_{ij}^2}. \quad (1)$$

We let $M_{(s,:)}$ and $M_{(:,t)}$ denote the s-th row and t-th column vector of matrix M, respectively.

[1] http://trecvid.nist.gov/

B. Proposed Framework

In applications of video semantic recognition, such as video concept detection, video classification, and human action recognition, the extracted video features are usually high-dimensional. Selecting a subset of features for a compact and accurate video representation will improve the efficiency and accuracy of video semantic recognition. To select the most discriminative video features for video semantic recognition, we assume there is a transformation matrix $W \in \mathbb{R}^{d\times c}(c < d)$ which maps the high-dimensional video samples onto a lower-dimensional subspace, and $x'_i = W^T x_i$ is the new representation for each video sample x_i in such subspace. As each row of W is used to weight each feature, if some rows of W shrink to zero, W can be used for feature selection. In the general framework of graph embedding for dimensionality reduction [36], a better transformation matrix W can be learned by the minimization of $Tr(W^T\mathcal{M}W)$, where matrix $\mathcal{M}$ encodes certain structures of the training data. In this paper, we propose the framework of semi-supervised feature selection to solve the following $\ell_{2,1}$-norm regularized minimization problem:

$$\min_{W^TW=I} Tr(W^T\mathcal{M}W) + \lambda||W||_{2,1}, \tag{2}$$

where the regularization term $||W||_{2,1}$ controls the capacity of W and also ensures that W is sparse in rows, making it particularly suitable for feature selection. Parameter $\lambda > 0$ controls the regularization effect, which should be well tuned. $\mathcal{M} \in \mathbb{R}^{d\times d}$ is a semi-supervised scatter matrix which encodes both data distribution and label information. The orthogonal constraint $W^TW = I$ is imposed to avoid arbitrary scaling and the trivial solution of all zeros.

We define $\mathcal{M}$ as:

$$\mathcal{M} = \mathcal{A} + \mu\mathcal{D}, \tag{3}$$

where the weight parameter μ $(0 \leq \mu \leq 1)$ is used to control the weight of matrix $\mathcal{D}$. Matrix $\mathcal{A} \in \mathbb{R}^{d\times d}$ is a scatter matrix which encodes label information of labeled training videos. Matrix $\mathcal{D} \in \mathbb{R}^{d\times d}$ is a scatter matrix which encodes local structural information of all training videos (both labeled and un-labeled). Thus, if $\mu = 0$ we incorporate no local distribution of training videos. In the experiments of this paper, we set $\mu = 1$ to treat equally the scatter matrices $\mathcal{D}$ and $\mathcal{A}$. In the following section, we present the details of matrix $\mathcal{A}$ and $\mathcal{D}$.

C. Estimation of Scatter Matrices

1) The Within-Class Scatter Matrix: Fisher discriminant analysis [14] is a well-known method to utilize discriminative information of the labeled data to find a low dimensional subspace to better separate samples. Fisher discriminant analysis maximizes the ratio of between-class and within-class scatter matrices. In this way, data from the same class are close to each other and data from different classes are far apart from each other in the subspace. If we incorporate between-class and within-class scatter matrices into $\mathcal{A}$ of Eq. (3) one more parameter has to be introduced [37], adding up the difficulty to tune its value. Thus, in this work, we use the within-class scatter matrix of Fisher discriminant analysis to encode the label information of training videos.

The within-class scatter matrix $\mathcal{A}$ is estimated as follows.

$$\mathcal{A} = \sum_{j=1}^{c} \frac{1}{N_j} \sum_{\mathbf{x}\in\omega_j} (\mathbf{x} - m_j)(\mathbf{x} - m_j)^T, \tag{4}$$

where $m_j = \frac{1}{N_j} Y_{(j,:)} X^T$ is the sample mean m_j $(j = 1,\ldots,c)$ for the j-th class, and $N_j = \sum_{i=1}^{n_l} Y_{(j,i)}$ is the number of labeled samples in class j. $\omega_j = \{x_i | Y_{(j,i)} = 1\}$ is the set of labeled videos in class j.

2) The Spline Scatter Matrix: Suppose matrix $\mathcal{G} \in \mathbb{R}^{n\times n}$ encodes the local similarity relationship of each pair of samples in $\mathcal{X}$, then the local structure of training videos can be preserved in $X\mathcal{G}X^T$. A recent study [25] shows that, if the local geometry of training data (both labeled and un-labeled) are represented in $\mathcal{G}$, then the unsupervised local distribution of training data can be utilized. We define the spline scatter matrix $\mathcal{D}$ to be:

$$\mathcal{D} = X\mathcal{G}X^T, \tag{5}$$

where matrix $\mathcal{G}$ is obtained by a local spline regression [25]. It has been shown that splines developed in Sobolev space [26] can be used to interpolate the scattered distribution and preserve the local geometry structure of training data. A Sobolev space is a space of functions with sufficiently many derivatives for some applications domain [26]. One important property of the Sobolev space is that this space provides conditions under which a function can be approximated by smooth functions. Splines developed in Sobolev space [26] are a combination of polynomials and Green's function which is popularly used to interpolate scattered data in geometrical design [27]. This spline is smooth, nonlinear, and able to interpolate the scattered data points with high accuracy. Recent research has showed that it can effectively handle high-dimensional data [25]. In the following, we briefly introduce how to estimate the matrix $\mathcal{G}$.

Given each datum $x_i \in \mathcal{X}$, to exploit its local similarity structure, we add its $k-1$ nearest neighbors as well as x_i itself into a local clique denoted as $\mathcal{N}_i = \{x_i, x_{i_1}, x_{i_2}, \ldots, x_{i_{k-1}}\}$. The goal of local spline regression is to find a function $g_i : \mathbb{R}^d \to \mathbb{R}$ such that it can directly associate each data point $x_{i_j} \in \mathbb{R}^d$ to a class label $y_{i_j} = g_i(x_{i_j})$ $(j = 1, 2, \ldots, k)$, which is a regularized regression process:

$$J(g_i) = \sum_{j=1}^{k} \left(f_{i_j} - g_i(x_{i_j})\right)^2 + \gamma\mathcal{S}(g_i), \tag{6}$$

where $\mathcal{S}(g_i)$ is a penalty functional and $\gamma > 0$ is a trade-off parameter. Parameter γ controls the amount of smoothness of the spline [25], which should be well tuned. According to the setting of [25], we fix γ to be 0.0001 in all the experiments of this paper. In order to utilize the good characteristics of splines in Sobolev space [38], provided the penalty term $\mathcal{S}(g_i)$ is defined as a semi-norm[2], the minimizer g_i in Eq. (6) is given

[2] A norm is a function that assigns a strictly positive length or size to all vectors in a vector space, other than the zero vector (which has zero length assigned to it). A semi-norm, on the other hand, is allowed to assign zero length to some non-zero vectors (in addition to the zero vector).

by

$$g_i(x) = \sum_{j=1}^{m} \beta_{i,j} p_j(x) + \sum_{j=1}^{k} \alpha_{i,j} G_{i,j}(x), \tag{7}$$

where $m = (d+s-1)!/(d!(s-1)!)$ [38]. $\{p_j(x)\}_{j=1}^m$ and $G_{i,j}$ are a set of primitive polynomials and a Green's function, respectively, which are defined in [38]. In mathematics, a Green's function is the impulse response of an inhomogeneous differential equation defined on a domain, with specified initial conditions or boundary conditions. In the spline, Green's function is a conditionally positive semidefinite function, which is used to interpolate scattered data in geometrical design [38]. It has been shown in [25] that the local function $g_i(x)$ can better fit the local geometry structure near the scattered points, as the data points can be locally wrapped by the Green's function $G_{i,j}(x)$. Now, our task is to estimate the parameters α and β. According to [38], The coefficients α_i and β_i can be solved by

$$A \cdot \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix} = \begin{pmatrix} Y_i^T \\ \mathbf{0} \end{pmatrix} \tag{8}$$

where $Y_i = [y_i, y_{i_1}, y_{i_2}, \ldots, y_{i_{k-1}}]$ corresponds to the label indicator of data points in $\mathcal{N}_i$ generated by the local function g_i and $A = \begin{pmatrix} \mathbf{K}_i & P \\ P^T & \mathbf{0} \end{pmatrix} \in \mathbb{R}^{(k+m)\times(k+m)}$, in which $\mathbf{K}_i$ is a $k \times k$ symmetrical matrix with its elements $\mathbf{K}_{p,q} = G_{p,q}(||x_{i_p} - x_{i_q}||)$ and P is a $k \times m$ matrix with its elements $P_{i,j} = p_i(x_{i_j})$. Denoting M_i as the upper left $k \times k$ sub-matrix of the matrix A^{-1}, it can be demonstrated that [25], [38]

$$J(g_i) \approx \eta Y_i^T M_i Y_i, \tag{9}$$

where η is a scalar. Since there are n local functions with respect to n local cliques, now we consider how to integrate the label indicators generated by different local functions. As can be seen that each local indicator matrix $Y_i = [y_i, y_{i_1}, \ldots, y_{i_{k-1}}]$ is a sub-matrix of the global indicator matrix $Y = [y_1, y_2, \ldots, y_n]$, we can find a column selection matrix $S_i \in \mathbb{R}^{n\times k}$ to map the global indicator matrix into the local indicator matrix.

More specifically, given the r-th row and c-th column element $S_i(r,c)$, if the column selection matrix S_i satisfies

$$S_i(r,c) = \begin{cases} 1, & \text{if } r = i_c, \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

then we have $Y_i = YS_i$. In this way, the global label indicator matrix Y can be mapped into n local indicator matrices by n column selection matrices. Thus the combined local loss turns to be

$$\sum_{i=1}^{n} J(g_i) = \gamma \sum_{i=1}^{n} Y_i^T M_i Y_i = \gamma S^T Y^T M Y S \tag{11}$$

where $S = [S_1, S_2, \ldots, S_n]$ and $M = diag(M_1, M_2, \ldots, M_n)$. For each video point, the local indicators generated by different local functions are integrated into one matrix to find the overall optimized label indicator matrix. Defining

$$\mathcal{G} = S^T M S, \tag{12}$$

the spline scatter matrix $\mathcal{D} = X\mathcal{G}X^T$, which sums up local distributions and encodes geometry structure of labeled and un-labeled training videos.

Algorithm 1 Semi-Supervised Feature Selection via Spline Regression (S^{2}FS^{2}R)

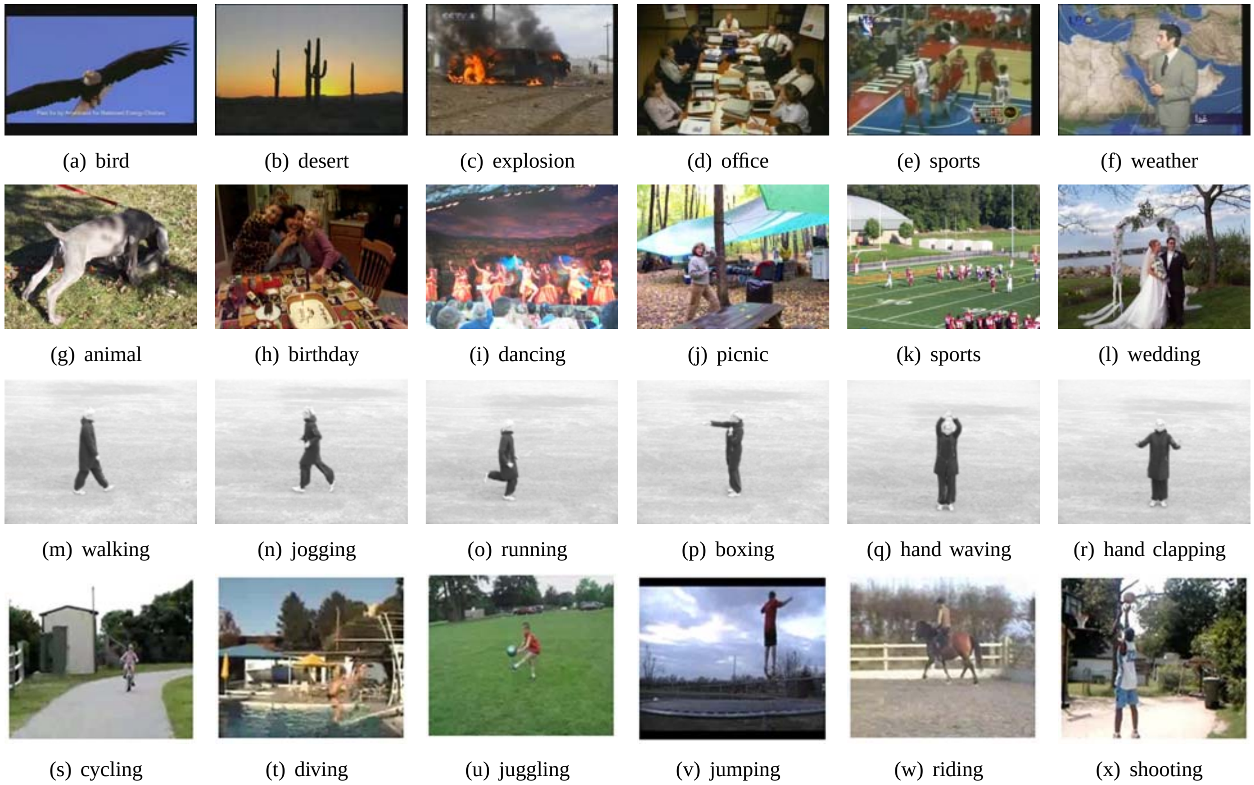
Input: matrix of n training videos $X = [x_1, \ldots, x_n] \in \mathbb{R}^{d\times n}$, $X^{\mathcal{L}} = [x_1, \ldots, x_{n_l}] \in \mathbb{R}^{d\times n_l}$ is a matrix of first $n_l (n_l \le n)$ labeled video samples and $Y^{\mathcal{L}} = [y_1, \ldots, y_{n_l}] \in \{0,1\}^{c\times n_l}$ is the corresponding indicator matrix for c labels (or semantic categories); $X^{\mathcal{U}} = [x_{n_l+1}, \ldots, x_{n_l+n_u}] \in \mathbb{R}^{d\times n_u}$ is a matrix of un-labeled videos whose labels are not given; k is the number of the nearest neighbors in local clique $\mathcal{N}_i$ for each video x_i; Control parameter μ and regularization parameter λ; f is the number of features to be selected.

Output: index idx of the top f selected features

```
for each video $x_i \in X$ do
    Construct local clique $\mathcal{N}_i$ by adding $x_i$ with its $k-1$ nearest neighbors;
    Construct matrix $\mathbf{K}_i$ using Green's function $G_{i,j}$ defined on $\mathcal{N}_i$;
    Construct matrix $A = \begin{pmatrix} \mathbf{K}_i & P \\ P^T & \mathbf{0} \end{pmatrix}$;
    Construct matrix $M_i$ which is the up left $k \times k$ submatrix of the matrix $A^{-1}$;
end for
Form matrix $\mathcal{D}$ using Eq. (5);
Form matrix $\mathcal{A}$ using Eq. (4);
Form matrix $\mathcal{M}$;
Set $t = 0$ and initialize $D_{(0)} \in \mathbb{R}^{d\times d}$ to be an identity matrix;
repeat
    $U_{(t)} = \mathcal{M} + \lambda D_{(t)}$;
    $W_{(t)} = [u_1, \ldots, u_c]$ where $u_1, \ldots, u_c$ are the eigenvectors of $U_{(t)}$ corresponding to the first $c$ smallest eigenvalues;
    Update matrix $D_{(t+1)}$ as
        $D_{(t+1)} = \begin{bmatrix} \frac{1}{2||w_{(t)}^1||_2} & & \\ & \cdots & \\ & & \frac{1}{2||w_{(t)}^d||_2} \end{bmatrix}$;
    $t = t + 1$;
until convergence.
Sort each feature of the $j$-th video sample $X_{(j,i)}|_{i=1}^d$ according to the value of $||w_i||_2$ in descending order;
Output the index $idx$ of the top $f$ selected features.
```

D. Solution and Algorithm

The $\ell_{2,1}$-norm regularized minimization problem has been studied in previous works [15]. However, it remains unclear how to directly apply the existing algorithms to optimize our objective function in Eq. (2), where the orthogonal constraint $W^T W = I$ is imposed. In this section, we give a new approach to solve the optimization problem shown in Eq. (2) for feature selection. The proposed algorithm is very efficient to solve the $\ell_{2,1}$-norm minimization problem with the orthogonal



(a) bird (b) desert (c) explosion (d) office (e) sports (f) weather

(g) animal (h) birthday (i) dancing (j) picnic (k) sports (l) wedding

(m) walking (n) jogging (o) running (p) boxing (q) hand waving (r) hand clapping

(s) cycling (t) diving (u) juggling (v) jumping (w) riding (x) shooting

Fig. 2. Example video frames from the three datasets. From the top to the bottom rows are videos from TRECVID, Kodak, KTH, and UCF YouTube datasets, respectively.

constraint. We summarize the detailed solution of S^{2}FS^{2}R in Algorithm 1. Once the optimal W is obtained, we sort the d features of the j-th video sample $X_{(j,i)}|_{i=1}^{d}$ according to the value of $||w_i||_2$ $(i = 1, \ldots, d)$ in descending order and select top ranked video features.

From step 11 to step 16 in Algorithm 1, we propose an iterative approach to optimize the minimization problem in Eq. (2). In the following, we verify in Theorem 1 that the proposed iterative approach in Algorithm 1 converges to the optimal W corresponding to Eq. (2). We mainly follow the proof from our previous work [11] to prove Theorem 1. The details of the proof are given in Appendix A.

Theorem 1. *The iterative approach in Algorithm 1 (from step 1 to step 16) monotonically decreases the objective function value of $Tr(W^T \mathcal{M} W) + \lambda \sum_{i=1}^{d} ||w^i||_2, s.t. W^T W = I$ in each iteration until convergence [11].*

According to Theorem 1, we can see that the iterative approach in Algorithm 1 converges to the optimal W corresponding to Eq. (2). In Algorithm 1, because k is much smaller than n, the time complexity of computing $\mathcal{D}$, $\mathcal{A}$, and $\mathcal{M}$ is about $O(n^2)$. Moreover, the computation of $\mathcal{D}$, $\mathcal{A}$, and $\mathcal{M}$ is outside the iterative process of Algorithm 1. Thus, to optimize the objective function of S^{2}FS^{2}R, the most time consuming operation is to perform eigen-decomposition of $U_{(t)}$. Note that $U_{(t)} \in \mathbb{R}^{d \times d}$. According to [39], the eigen-decomposition of $U_{(t)}$ is solved by the tridiagonal QR iteration algorithm, which is the main algorithm of function eig in matlab. It first performs tridiagonal reduction of $U_{(t)}$, which needs $\frac{8}{3}d^3 + O(d^2)$ flops [39]. Then the tridiagonal QR iteration needs $O(d^2)$ flops. Thus, the time complexity of this operation is $O(d^3)$ approximately.

IV. Experiments

In this section, three typical tasks of video semantic recognition, i.e., video concept detection in news videos, video classification of consumer videos, and human action recognition, are used to investigate the performance of the proposed S^{2}FS^{2}R algorithm. Accordingly, we use four open benchmark video datasets to compare S^{2}FS^{2}R with the state-of-the-art algorithms.

A. Video Datasets

We choose four video datasets, i.e., TRECVID[3], Kodak [40], KTH [41], and UCF YouTube action dataset [42] in our experiments. In Figure 2, we show sample videos and corresponding class labels/concepts of TRECVID, Kodak, KTH, and UCF YouTube. We summarize the datasets used in our experiment in Table I. The following is a brief description of the four datasets.

TRECVID: We use the Columbia374 baseline detectors [43] for TRECVID 2005[4] in our experiments. TRECVID 2005 consists of about 170 hours of TV news videos from 13 different programs in English, Arabic, and Chinese. We use the development set in our experiments, since there are

[3] http://trecvid.nist.gov/

[4] http://www-nlpir.nist.gov/projects/tv2005/

TABLE I

A BRIEF SUMMARY OF FOUR VIDEO DATASETS USED IN OUR EXPERIMENT. IN THIS TABLE, N, d, AND c DENOTE THE NUMBER OF INSTANCES, DIMENSIONALITY OF VIDEO FEATURES, AND THE NUMBER OF CLASSES IN EACH OF THE FOUR DATASETS, RESPECTIVELY.

Dataset	TRECVID	Kodak	KTH	UCF YouTube
Video Types	News	Consumer	Human Action	Human Action
Tasks	Video Concept Detection	Video Concept Detection	Human Action Recognition	Human Action Recognition "in the Wild"
N	61,562	3,590	2,391	1,596
d	546	1,000	1,000	1,000
c	39	22	6	11

annotations of semantic concepts defined in LSCOM (Large-Scale Concept Ontology for Multimedia) [43], which could be taken as the ground truth. As there are 39 concepts annotated in the TRECVID 2005 dataset in total, we use all these 39 concepts in our experiment. Thus, the dataset used in our experiments includes 61,562 labeled key frames. Three global feature types used in [43], namely, 73-dimensional edged direction histogram (EDH), 48-dimensional Gabor (GBR), 225-dimensional grid color moment (GCM) and 200-dimensional canny edge provided by NIST are combined to be a 546-dimensional vector of global features to represent each key frame in our experiments.

Kodak: There are 5,166 key frames extracted from 1,358 consumer video clips in this dataset. Among these key frames, 3,590 key frames are annotated by students from Columbia University, who were asked to assign binary labels for each concept. We use all the annotated keyframes belonging to 22 concepts in our experiments. We extracted SIFT points for each key frame. Then the randomly selected subset of extracted SIFT points are clustered and produces the 1,000 centers as the visual dictionary. Finally, each key frame is quantized into a 1,000 dimensional histogram of bag-of-visual-words (BoW).

KTH: KTH actions dataset [41] contains six types of human actions (walking, jogging, running, boxing, hand waving, and had clapping) performed several times by 25 subjects in four different scenarios. Currently the dataset contains 2,391 videos sequences. In our experiments, we describe each video sequences using space-time interest points (STIP) [7]. For each STIP point, descriptors of the associated space-time patch were computed. Two alternative patch descriptors were computed in terms of (i) histograms of oriented (spatial) gradient (HOG) and (ii) histograms of optical flow (HOF). Thus, STIP descriptor concatenates several histograms from a space-time grid defined on the patch and generalizes SIFT descriptor to space-time. We built a 1,000 dimensional visual vocabulary of local space-time descriptors and assign each interest point to a visual word label. In this way, each video sequence in KTH is represented by a 1,000 dimensional STIP feature.

UCF YouTube: UCF YouTube action dataset [42] contains 11 action categories: basketball shooting, biking/cycling, diving, golf swinging, horse back riding, soccer juggling, swinging, tennis swinging, trampoline jumping, volleyball spiking, and walking with a dog. This dataset is very challenging for recognizing realistic actions from videos "in the Wild", due to large variations in camera motion, object appearance and pose, object scale, viewpoint, cluttered background, illumination conditions, etc. For each category, the videos are grouped into 25 groups with more than 4 actions clips in each group. The video clips in the same group may share some common features, such as the same actor, similar background, similar viewpoint, and so on. In our experiments, we describe each video sequence using space-time interest points (STIP) [7]. We built a 1,000 dimensional visual vocabulary of local space-time descriptors and assign each interest point to a visual word label. In this way, each video sequence in UCF YouTube is represented by a 1,000 dimensional STIP feature.

B. Evaluation Metric

We evaluate the classification performance in terms of F1-Score (F-measure). Since there are multiple concepts (semantic categories) in our experiments, to measure the global performance across multiple classes, we use the *microaveraging* methods following [44]. Therefore, the evaluation criterion we use is $microF_1$. More specifically, we present the "micro-" definition as follows.

Let $Y^* \in \{0,1\}^{n\times c}$ denote the indicator matrix of ground truth for testing data, and $\hat{Y}^* \in \mathbb{R}^{n\times c}$ denote the corresponding estimated indicator matrix, where c denotes the number of classes. Function $F_1(a,b)$ compute the F1-score between vector a and b. Let function $Vec(A)$ denote the operator that converts matrix A to a vector by concatenating each column sequentially, then the "micro-" criterion is

$$microF_1 = F_1(Vec(Y^*), Vec(\hat{Y}^*)),$$

where F_1 score is defined as the harmonic mean of precision and recall, where the functions of $precision(a,b)$ and $recall(a,b)$ are defined in [45].

$$F_1(a,b) = \frac{2\cdot precision(a,b)\cdot recall(a,b)}{precision(a,b)+recall(a,b)}.$$

C. Experimental Configuration

1) Parameter Setting: Four parameters, i.e., k, μ, λ, and f in Algorithm 1 need to be set and tuned. In our experiments, we chose $k = 5, 10$ in the construction of local clique $\mathcal{N}_i$ for each video x_i. We set $\mu = 1$ to treat equally the scatter matrices D and A. Parameter λ determines the regularization effect of $\ell_{2,1}$-norm in Eq. (2), which should be well tuned. The best number of features to be selected, i.e., f, will be different for different feature types and different video data. In our experiments, we use a 5-fold cross-validation process to tune parameter λ and f simultaneously. The ranges for λ

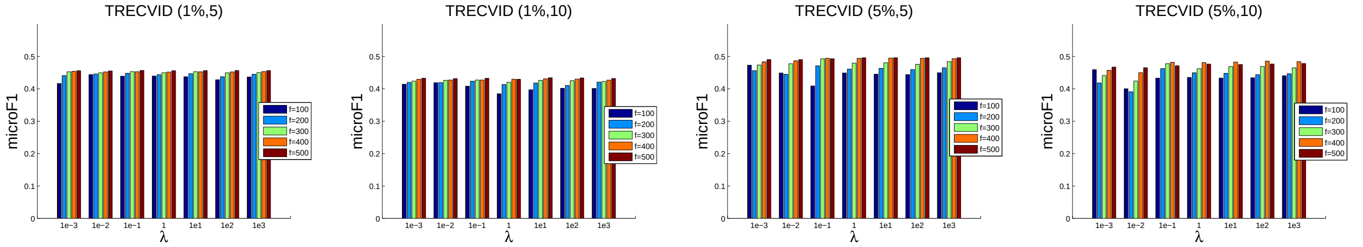


Fig. 3. Different performance of video semantic recognition by the proposed S^{2}FS^{2}R for TRECVID, when λ and f are set to different values. Impacts of parameters are reported when the ratios of labeled training data are set to be 5%, and 1%. The numbers "5" and "10" after the ratio in figures' title denote the value of $k = 5, 10$ in the construction of the local clique $\mathcal{N}_i$. For example, "(1%,5)" denotes that the ratio of labeled training data is 1% and $k = 5$ for $\mathcal{N}_i$.

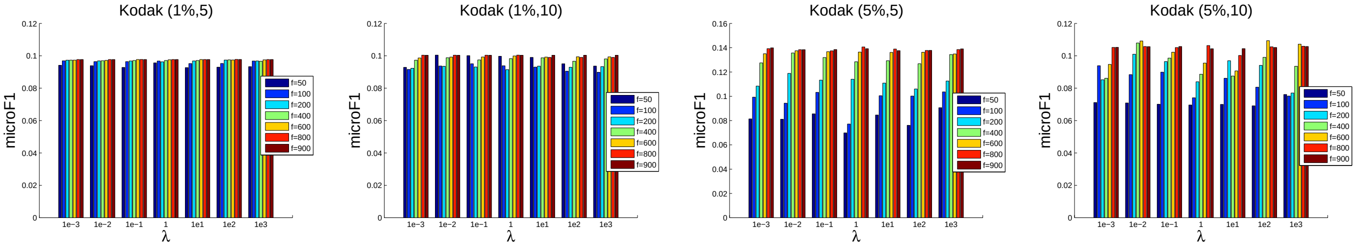


Fig. 4. Different performance of video semantic recognition by the proposed S^{2}FS^{2}R for Kodak, when λ and f are set to different values. Impacts of parameters are reported when the ratios of labeled training data are set to be 5%, and 1%. The numbers "5" and "10" after the ratio in figures' title denote the value of $k = 5, 10$ in the construction of the local clique $\mathcal{N}_i$. For example, "(1%,5)" denotes that the ratio of labeled training data is 1% and $k = 5$ for $\mathcal{N}_i$.

are set to be $\lambda \in$ {1e-3, 1e-2, 1e-1, 1, 10, 100, 1,000} for all datasets. Because the feature dimensionality of TRECVID is $d = 546$ and $f \leq d$ (see Section IV-A), the ranges of f for TRECVID are $f \in$ {100, 200, 300, 400, 500}. And $f \in$ {50, 100, 200, 400, 600, 800, 900} for Kodak, KTH and UCF YouTube datasets, as the feature dimensionality of these three datasets is $d = 1,000$.

2) Partition of Training/Testing Videos: We randomly sampled 10,000 and 2,000 video key frames as the training data for TRECVID and Kodak datasets, respectively. For KTH and UCF YouTube datasets, we randomly sampled 1,000 video clips as training data. The remaining data are used as the corresponding testing data for each of the four datasets. For all these datasets, the sampling processes were repeated five times to generate five random training/testing partitions, and then the average performance of five-round repetitions is reported. The significance of the repeated results has been demonstrated according to the Student's t-test. In this experiment, we report the average results from the repetitions. For the first random partition of the five-round repetitions, we tuned and chose the best parameters λ and f using the 5-fold cross-validation. Then the tuned values of λ and f were fixed for all the rest of the partitions. In order to investigate the performance of semi-supervised feature selection, we set the ratio of labeled training videos in the sampled training videos to different values from {50%, 25%, 10%, 5%, 1%}.

3) Classifiers and Comparison Methods: Once the index idx of features to be selected is obtained, we train a classifier on the selected video features. In our experiments, we chose kNN classifier ($k = 10$) for the four datasets. Furthermore, as shown in [46], the χ^2 kernel SVM is a better classifier for human action recognition, especially for the BoW histogram representations. Thus, in this experiment, for the action recognition task in KTH and UCF YouTube datasets, we also report the results from the χ^2 kernel in a Support Vector Machine (χ^2-SVM). To show the comparative performance, we first compare S^{2}FS^{2}R with two baselines:

- Classification with full features: Conduct classification on the original features by kNN ($k = 10$) or χ^2-SVM.
- Classification with PCA [47]: Conduct classification on the reduced features obtained by dimensionality reduction with PCA.

We also compare S^{2}FS^{2}R with four state-of-the-art feature selection methods. Detailed information of these methods is given as follows.

- Fisher Score (FScore) [14]: It depends on fully labeled training data to select features with the best discriminating ability.
- Feature Selection via Spectral Analysis (FSSA) [12]: It is a semi-supervised feature selection method using spectral regression.
- Feature Selection via Joint $\ell_{2,1}$-Norms Minimization (F-SNM) [15]: It employs joint $\ell_{2,1}$-norm minimization on both loss function and regularization to realize feature selection across all data points.
- Sparse Multinomial Logistic Regression via Bayesian ℓ_1 Regularization (SBMLR) [32]: It exploits sparsity by using a Laplace prior and is used for multi-class pattern recognition. It can also be applied to feature selection.
- Discriminative Semi-Supervised Feature Selection via Manifold Regularization (FS-Manifold) [20]: It selects features through maximizing the classification margin between different classes and simultaneously exploiting the data geometry by the manifold regularization.

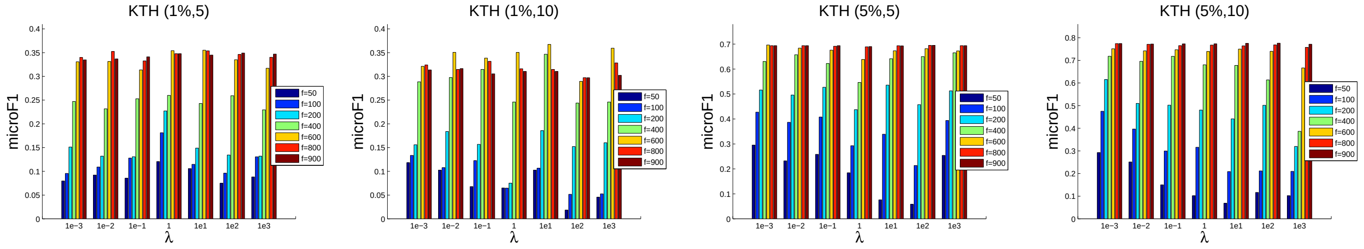


Fig. 5. Different performance of video semantic recognition by the proposed S^2FS^{2}R for KTH, when λ and f are set to different values. Impacts of parameters are reported when the ratios of labeled training data are set to be 5%, and 1%. The numbers "5" and "10" after the ratio in figures' title denote the value of $k = 5, 10$ in the construction of the local clique $\mathcal{N}_i$. For example, "(1%,5)" denotes that the ratio of labeled training data is 1% and $k = 5$ for $\mathcal{N}_i$.

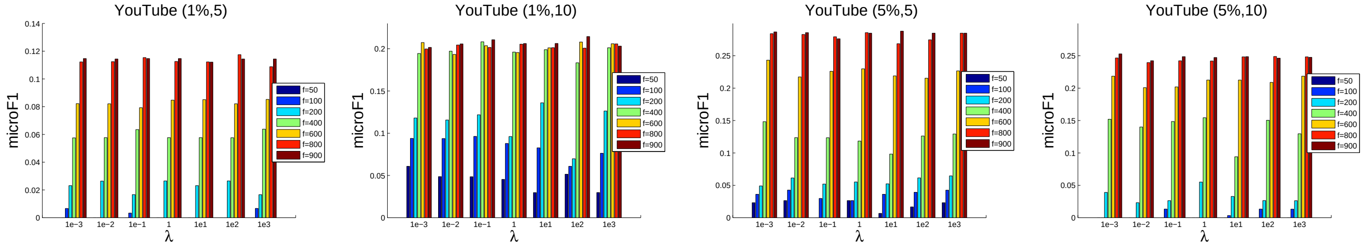


Fig. 6. Different performance of video semantic recognition by the proposed S^2FS^{2}R for YouTube, when λ and f are set to different values. Impacts of parameters are reported when the ratios of labeled training data are set to be 5%, and 1%. The numbers "5" and "10" after the ratio in figures' title denote the value of $k = 5, 10$ in the construction of the local clique $\mathcal{N}_i$. For example, "(1%,5)" denotes that the ratio of labeled training data is 1% and $k = 5$ for $\mathcal{N}_i$.

Moreover, we investigate special instantiations of S^{2}FS^{2}R, which correspond to different settings of $|\mathcal{N}_i| = 5, 10$ and $\mu = 0, 1$. To demonstrate the impact of the size of local clique $\mathcal{N}_i$ in the local spline regression, we let "S^{2}FS^{2}R(5)" and "S^{2}FS^{2}R(10)" denote S^{2}FS^{2}R with $|\mathcal{N}_i| = 5$ and $|\mathcal{N}_i| = 10$, respectively. Note that, when $\mu = 0$ we have $\mathcal{M} = \mathcal{A}$ (see Eq. (3)), which means the spline scatter matrix $\mathcal{D}$ is not included and the information of unsupervised local distribution is not utilized. In the following, we let "S^{2}FS^{2}R(without local)" denote S^{2}FS^{2}R with $\mu = 0$.

D. Experimental Results

1) Impacts of Parameters: In this section, we investigate the impacts of parameters λ and f for different tasks of video semantic recognition. In Figure 3 - Figure 6, we show the performance of video semantic recognition by S^{2}FS^{2}R for TRECVID, Kodak, KTH, and UCF YouTube datasets, respectively. From the figures we note that the parameters λ and f have different impacts on the performance of different video semantic recognition and on different datasets. Firstly, the performance of video concept detection on TRECVID varies little when λ and f are set to different values, whereas, the performances of video semantic recognition on Kodak, KTH, and UCF YouTube have bigger variances than that of TRECVID dataset. From these results we can see that the local features used in Kodak, KTH, and UCF YouTube are more sensitive to parameters λ and f than to the global visual features, which are used to represent key frames in TRECVID. Especially, the performance of action recognition is very sensitive to the number f of selected features. Secondly, we can observe in some cases (e.g., TRECVID (5%,5) and f=100, 200), that the performance of video semantic recognition decreases when increasing f. A possible reason could be that, when f is set to $f = 200$, more noisy features are selected than in the case of $f = 100$. Thirdly, for each of the four datasets we can observe that the best performance of video semantic recognition can be obtained by S^{2}FS^{2}R when f is set to larger values of the tuning ranges, e.g., 400 or 500 of 546 for TRECVID and 600 or 800 of 1,000 for Kodak. This demonstrates that, for the video features used in this experiment, most of the dimensions contribute to video semantic recognition, given that the number of noisy features is small. However in some cases (e.g., f is set to be small values), more noisy features may be selected when f is larger. In this experiment, we choose the best performance when λ and f are set to different values. Moreover, as we will report in the following results, the performance of S^{2}FS^{2}R is better than when using all the features. It is clear that S^{2}FS^{2}R can select the most discriminative subset of features for video semantic recognition.

2) Video Semantic Recognition Results: In this section, we first investigate the performance of S^{2}FS^{2}R compared with the state-of-the-art methods for different tasks of video semantic recognition: video concept detection for TRECVID videos, consumer videos classification for Kodak videos, and human action recognition for videos in KTH and UCF YouTube. In order to show the impacts of different ratios of labeled training videos for semi-supervised methods, we report results when the ratios of labeled training videos are set to 50% and 5%. As is shown in Table II and Table III, results in the left four columns are obtained using the kNN ($k = 10$) classifier, whereas "χ^2-SVM" denotes that we also report the results using the χ^2-SVM classifier for KTH and YouTube. From the results we can observe: (1) The proposed framework

TABLE II

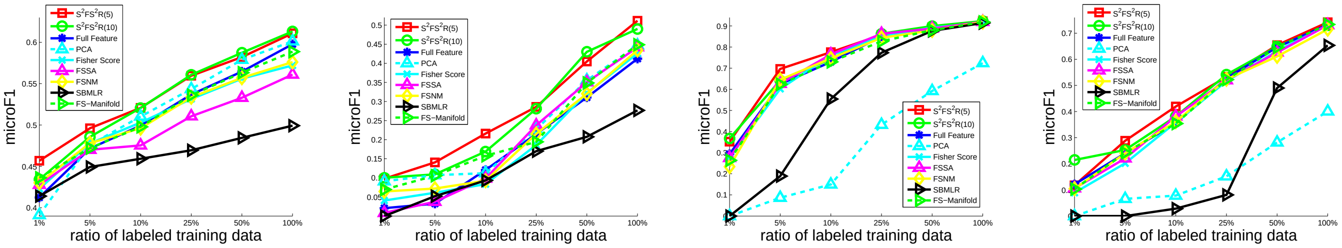
Comparison results of video semantic recognition on different video datasets. Full Feature denotes the baseline of classification with full features. PCA denotes the baseline of classification with PCA. For the semi-supervised S^2FS^2R and FSSA, the ratio of labeled training video is 50%. In the first four columns, we report the results using the kNN ($k = 10$) classifier. For KTH and YouTube, we also report the results using the χ^2-SVM classifier. The number in [] denotes the reference index.

Methods	TRECVID (kNN)	Kodak (kNN)	KTH (kNN)	YouTube (kNN)	KTH (χ^2-SVM)	YouTube (χ^2-SVM)
S^2FS^2R(5)	0.5821	0.4047	0.6252	**0.2982**	0.8940	**0.6540**
S^2FS^2R(10)	**0.5874**	**0.4301**	**0.6714**	0.2894	**0.8994**	0.6485
S^2FS^2R(without local)	0.5511	0.3107	0.5569	0.2660	0.8910	0.6279
Full Feature	0.5646	0.3107	0.5611	0.2376	0.8858	0.6459
PCA	0.5789	0.3556	0.5923	0.2817	0.1592	0.0926
FScore [14]	0.5561	0.3224	0.6080	0.2824	0.8922	0.6314
FSSA [12]	0.5330	0.3506	0.6130	0.2567	0.8876	0.6261
FSNM [15]	0.5571	0.3203	0.5765	0.2693	0.8784	0.6109
SBMLR [32]	0.4845	0.2075	0.6115	0.2562	0.8768	0.4899
FS-Manifold [20]	0.5633	0.3487	0.6133	0.2601	0.8799	0.6455

TABLE III

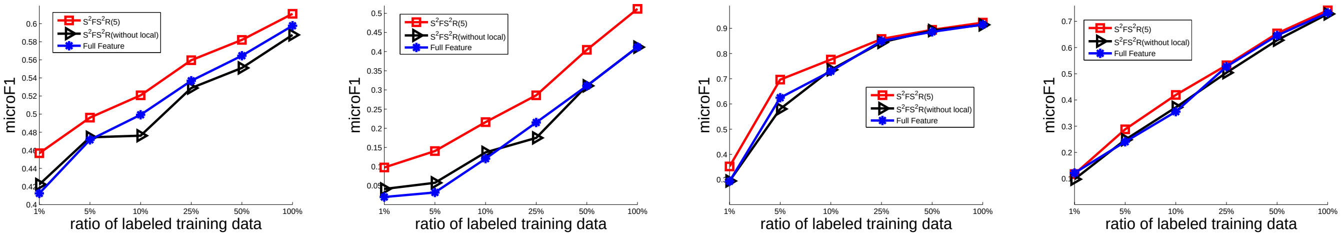
Comparison results of video semantic recognition on different video datasets. Full Feature denotes the baseline of classification with full features. PCA denotes the baseline of classification with PCA. For the semi-supervised S^2FS^2R and FSSA, the ratio of labeled training video is 5%. In the first four columns, we report the results using the kNN ($k = 10$) classifier. For KTH and YouTube, we also report the results using the χ^2-SVM classifier. The number in [] denotes the reference index.

Methods	TRECVID (kNN)	Kodak (kNN)	KTH (kNN)	YouTube (kNN)	KTH (χ^2-SVM)	YouTube (χ^2-SVM)
S^2FS^2R(5)	**0.4961**	**0.1406**	0.1981	0.0824	**0.6965**	**0.2881**
S^2FS^2R(10)	0.4857	0.1093	**0.2080**	**0.1275**	0.6419	0.2530
S^2FS^2R(without local)	0.4744	0.0578	0.0758	0.0228	0.5803	0.2482
Full Feature	0.4716	0.0326	0.0585	0.0298	0.6248	0.2406
PCA	0.4761	0.1071	0.0867	0.0781	0.1345	0.0755
FScore [14]	0.4778	0.0611	0.0917	0.0686	0.6021	0.2020
FSSA [12]	0.4701	0.0375	0.0192	0.0345	0.6261	0.2208
FSNM [15]	0.4781	0.0712	0.0241	0.0373	0.6454	0.2403
SBMLR [32]	0.4493	0.0000	0.0249	0.0000	0.1891	0.0000
FS-Manifold [20]	0.4721	0.1057	0.0604	0.0418	0.6233	0.2419



(a) TRECVID (b) Kodak (c) KTH (d) YouTube

Fig. 7. Performance comparison of S^2FS^2R with the baselines and the state-of-the-art methods on TRECVID, Kodak, KTH, and YouTube datsets. The $micro$F1 scores are plotted when the ratios of labeled training data are set to 100%, 50%, 25%, 10%, 5%, and 1%. The results of PCA are obtained using the kNN ($k = 10$) classifier



(a) TRECVID (b) Kodak (c) KTH (d) YouTube

Fig. 8. Performance comparison of S^2FS^2R with S^2FS^2R(without local) and performing classification on the full features for TRECVID, Kodak, KTH, and YouTube datasets. The $micro$F1 scores are plotted when the ratios of labeled training data are set to 100%, 50%, 25%, 10%, 5%, and 1%.

of semi-supervised feature selection via spline regression outperforms the state-of-the-art methods for different settings of the ratio of labeled training videos. (2) When there are more labeled training videos (see Table II), S^{2}FS^{2}R with a bigger local clique $\mathcal{N}_i$ has a better performance than that with a smaller local clique for spline regression (except for YouTube dataset). Despite a little variance of performance for $\mathcal{N}_i = 5$ and 10, S^{2}FS^{2}R outperforms all the compared methods. (3) Comparing the results of S^{2}FS^{2}R(5) and S^{2}FS^{2}R(10) with that of Full Feature and PCA we note that, S^{2}FS^{2}R gains better performance than the case when using the full feature set and conducting dimensionality reduction using PCA. (4) The performance of conducting χ^2-SVM after performing PCA is poor for KTH and YouTube. As introduced in Section IV-A, we extract BoW histogram of STIP for KTH and YouTube. Thus, “PCA+χ^2-SVM” is not suitable for the BoW histogram. As is shown in Table II and III, the performance of conducting kNN ($k = 10$) after performing PCA is better.

3) Performance of Semi-Supervised Feature Selection: In order to investigate the performance of semi-supervised feature selection, we set the ratio of labeled training videos in the sampled training videos to different values of {50%, 25%, 10%, 5%, 1%}. Figure 7 shows the performance of video semantic recognition of different methods when the ratio of labeled training videos are set to different values. From the results we observe the following: (1) As the number of labeled training samples increases, the performance increases. (2) Compared to the supervised feature selection methods, S^{2}FS^{2}R(5) and S^{2}FS^{2}R(10) have competitive or better performance than that of Fisher Score, FSNM, and SBMLR, thanks to the preservation of local geometry structure of un-labeled videos via spline regression. (3) S^{2}FS^{2}R(5) and S^{2}FS^{2}R(10) outperform the semi-supervised FSSA on all the ratios of labeled training videos for TRECVID, Kodak, KTH, and YouTube. (4) When the ratio of labeled training videos is very low, e.g., 1%, S^{2}FS^{2}R outperforms all the compared methods, which shows a better property of semi-supervised feature selection.

Figure 8 shows the performance of comparing S^{2}FS^{2}R(5) with S^{2}FS^{2}R(without local) and the case when using full features for TRECVID, Kodak, KTH, and YouTube datasets. As introduced in the end of Section IV-C3, the information of local geometry of the training videos is not incorporated into S^{2}FS^{2}R(without local). S^{2}FS^{2}R(without local) can be taken as a supervised version of S^{2}FS^{2}R. From the results we observe that, without the local information, performance of S^{2}FS^{2}R(without local) is worse than the case when using the full feature set. Owing to the preservation of local geometry of the unlabeled data, S^{2}FS^{2}R(5) outperforms S^{2}FS^{2}R(without local) and when using the full feature set for the four datsets, which further demonstrates the strength of semi-supervised feature selection of S^{2}FS^{2}R.

4) Comparison of Computation Time: In Section III-D, we discuss the convergency and computational cost of our algorithm. To show the efficiency of S^{2}FS^{2}R, in this section, we compare the computation time of S^{2}FS^{2}R with two state-of-the-art semi-supervised feature selection algorithms, i.e., FSSA and FS-Manifold, as the proposed S^{2}FS^{2}R is also a semi-supervised feature selection algorithm. In Table IV, we report the comparison results of computational time of the training process of each algorithm. All these results are obtained after running the algorithms in MATLAB R2012b on a workstation with Windows Server 2008 R2 Enterprise. The system is equipped with the Intel(R) Xeon(R) CPU of 2.70 GHz and 64GB physical memory. For Kodak, KTH, and YouTube, we use the same partition of training/testing videos in Section IV-C2. For TRECVID, we randomly sampled 1,000 video key frames as the training data and the remaining data are used as the corresponding testing data. From the results we observe that S^{2}FS^{2}R is more efficient compared to FSSA and FS-Manifold.

TABLE IV
COMPARISON OF COMPUTATION TIME (SECONDS). WE REPORT THE RESULTS WHEN THE RATIOS OF LABELED TRAINING DATA ARE SET TO 50%, 25%, 10%, 5%, AND 1%, RESPECTIVELY.

Dataset	TRECVID				
Ratio	1%	5%	10%	25%	50%
S^{2}FS^{2}R	7.43	8.50	9.12	10.19	13.56
FSSA	77.84	81.10	82.89	83.36	92.41
FS-Manifold	12.54	15.98	36.82	163.45	308.11
Dataset	Kodak				
Ratio	1%	5%	10%	25%	50%
S^{2}FS^{2}R	9.69	16.96	23.44	31.20	36.86
FSSA	607.20	743.63	1010.77	1045.28	1371.25
FS-Manifold	28.27	32.85	35.82	162.29	840.26
Dataset	KTH				
Ratio	1%	5%	10%	25%	50%
S^{2}FS^{2}R	7.84	8.33	8.41	13.09	18.14
FSSA	68.62	68.90	69.47	70.47	72.92
FS-Manifold	60.02	192.08	218.14	243.19	382.50
Dataset	YouTube				
Ratio	1%	5%	10%	25%	50%
S^{2}FS^{2}R	6.91	15.29	23.29	26.24	54.68
FSSA	93.63	95.58	98.82	101.52	109.45
FS-Manifold	121.35	260.93	321.06	352.82	413.43

V. CONCLUSION

This paper proposed a framework for video semantic recognition by Semi-Supervised Feature Selection via Spline Regression (S^{2}FS^{2}R). In this framework, the discriminative information between labeled training videos and the local geometry structure of all the training videos are well preserved by the combined semi-supervised scatters: within-class scatter matrix encoding label information and spline scatter matrix encoding data distribution by spline regression. An $\ell_{2,1}$-norm is imposed as a regularization term on the transformation matrix to control the capacity and also to ensure it is sparse in rows. Three tasks of video semantic recognition were used in our experiments to investigate the performance of S^{2}FS^{2}R. To efficiently solve S^{2}FS^{2}R, we proposed an iterative algorithm and prove its convergence. Experimental results show that the proposed S^{2}FS^{2}R has better performance of feature selection compared to state-of-the-art methods. S^{2}FS^{2}R also has an extension ability of incorporating new neighborhood information into the feature selection process if we define new scatter matrices.

Appendix A
Proof of Theorem 1

Proof: According to the definition of $W_{(t)}$ in step 13 of Algorithm 1, we can see that

$$W_{(t)} = \underset{W^TW=I}{\arg\min}\, Tr\left(W^T(\mathcal{M}+\lambda D_{(t)})W\right) \tag{13}$$

That is to say, for any matrix A such that $A^TA = I$, $Tr\left(W_{(t)}^T(\mathcal{M}+\lambda D_{(t)})W_{(t)}\right) \leq Tr\left(A^T(\mathcal{M}+\lambda D_{(t)})A\right)$. Therefore, we have

$$\begin{aligned} & Tr\left(W_{(t)}^T(\mathcal{M}+\lambda D_{(t)})W_{(t)}\right) \leq \\ & Tr\left(W_{(t-1)}^T(\mathcal{M}+\lambda D_{(t)})W_{(t-1)}\right) \\ \Rightarrow\quad & Tr\left(W_{(t)}^T\mathcal{M}W_{(t)}\right) + \lambda\sum_i \frac{||w_{(t)}^i||_2^2}{2||w_{(t-1)}^i||_2} \leq \\ & Tr\left(W_{(t-1)}^T\mathcal{M}W_{(t-1)}\right) + \lambda\sum_i \frac{||w_{(t-1)}^i||_2^2}{2||w_{(t-1)}^i||_2} \end{aligned} \tag{14}$$

Then we have the following inequality

$$\begin{aligned} & Tr\left(W_{(t)}^T\mathcal{M}W_{(t)}\right) + \lambda\sum_i ||w_{(t)}^i||_2 - \\ & \lambda\left(\sum_i ||w_{(t)}^i||_2 - \sum_i \frac{||w_{(t)}^i||_2^2}{2||w_{(t-1)}^i||_2}\right) \\ \leq\quad & Tr\left(W_{(t-1)}^T\mathcal{M}W_{(t-1)}\right) + \lambda\sum_i ||w_{(t-1)}^i||_2 - \\ & \lambda\left(\sum_i ||w_{(t-1)}^i||_2 - \sum_i \frac{||w_{(t-1)}^i||_2^2}{2||w_{(t-1)}^i||_2}\right) \end{aligned} \tag{15}$$

According to Lemma 1 in [15], we have

$$\begin{aligned} & Tr\left(W_{(t)}^T\mathcal{M}W_{(t)}\right) + \lambda\sum_i ||w_{(t)}^i||_2 \\ \leq\quad & Tr\left(W_{(t-1)}^T\mathcal{M}W_{(t-1)}\right) + \lambda\sum_i ||w_{(t-1)}^i||_2, \end{aligned} \tag{16}$$

which indicates that the objective function value of $Tr(W^T\mathcal{M}W) + \lambda\sum_{i=1}^d ||w^i||_2, s.t. W^TW = I$ monotonically decreases until convergence using the updating rule in Algorithm 1. ■

References

[1] R. Ewerth and B. Freisleben, "Semi-supervised learning for semantic video retrieval," in *ACM International Conference on Image and Video Retrieval*, 2007, pp. 154–161.

[2] S. Dagtas, W. Al-Khatib, A. Ghafoor, and R. L. Kashyap, "Models for motion-based video indexing and retrieval," *IEEE Transactions on Image Processing*, vol. 9, no. 1, pp. 88–101, 2000.

[3] M. Chen, A. Hauptmann, A. Bharucha, H. Wactlar, and Y. Yang, "Human activity analysis for geriatric care in nursing home," in *Pacific-Rim Conference on Multimedia*, 2011.

[4] X. Zhen, L. Shao, D. Tao, and X. Li, "Embedding motion and structure features for action recognition," *IEEE Transactions on Circuits Systems for Video Technology*, vol. 23, no. 7, pp. 1182–1190, 2013.

[5] L. Maddalena and A. Petrosino, "Stopped object detection by learning foreground model in videos," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 5, pp. 723–735, 2013.

[6] R. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1631–1643, 2005.

[7] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2, pp. 107–123, 2005.

[8] M. Chen and A. Hauptmann, "Mosift: Recognizing human actions in surveillance videos," *CMU-CS-09-161, Carnegie Mellon University*, 2009.

[9] F. Korn, B. Pagel, and C. Faloutsos, "On the dimensionality curse and the self-similarity blessing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 13, no. 1, pp. 96–111, 2001.

[10] P. Padungweang, C. Lursinsap, and K. Sunat, "A discrimination analysis for unsupervised feature selection via optic diffraction principle," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 10, pp. 1587–1600, 2012.

[11] Y. Yang, H. Shen, Z. Ma, Z. Huang, and X. Zhou, "$\ell_{2,1}$-norm regularized discriminative feature selection for unsupervised learning," in *International Joint Conference on Artifical Intelligence (IJCAI-11)*, 2011, pp. 1589–1594.

[12] Z. Zhao and H. Liu, "Semi-supervised feature selection via spectral analysis," in *International Conference on Data Mining*, 2007, pp. 1151–1158.

[13] S. Xiang, F. Nie, G. Meng, C. Pan, and C. Zhang, "Discriminative least squares regression for multiclass classification and feature selection," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 11, pp. 1738–1754, 2012.

[14] R. Duda, P. Hart, and D. Stork, "Pattern classification, 2nd edition," *New York, USA: John Wiley & Sons.*, 2001.

[15] F. Nie, H. Huang, X. Cai, and C. Ding, "Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization," *Advances in Neural Information Processing Systems*, vol. 23, pp. 1813–1821, 2010.

[16] Z. Zhao, L. Wang, and H. Liu, "Efficient spectral feature selection with minimum redundancy," in *AAAI Conference on Artificial Intelligence (AAAI)*, 2010.

[17] Z. Zhao and H. Liu, "Spectral feature selection for supervised and unsupervised learning," in *International Conference on Machine Learning*, 2007, pp. 1151–1157.

[18] Z. Xiaojin, "Semi-supervised learning literature survey," *Computer Science, University Wisconsin-Madison, Technical Report*, 2007.

[19] Y. Wang, S. Chen, and Z.-H. Zhou, "New semi-supervised classification method based on modified cluster assumption," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 5, pp. 689–702, 2012.

[20] Z. Xu, R. Jin, M.-T. Lyu, and I. King, "Discriminative semi-supervised feature selection via manifold regularization," in *International Joint Conferences on Artificial Intelligence*, 2009, pp. 1303–1308.

[21] X. Kong and P. S. Yu, "Semi-supervised feature selection for graph classification," in *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2010, pp. 793–802.

[22] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," *Advances in neural information processing systems*, vol. 16, pp. 321–328, 2004.

[23] M. Wu and B. Schölkopf, "Transductive classification via local learning regularization," in *International Conference on Artificial Intelligence and Statistics*, 2007, pp. 624–631.

[24] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *The Journal of Machine Learning Research*, vol. 7, pp. 2399–2434, 2006.

[25] S. Xiang, F. Nie, C. Zhang, and C. Zhang, "Nonlinear dimensionality reduction with local spline embedding," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1285–1298, 2009.

[26] R. Adams, *Sobolev Spaces*. Academic Press, 1975.

[27] F. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 6, pp. 567–585, 1989.

[28] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[29] J. Yang, Y. Jiang, A. Hauptmann, and C. Ngo, "Evaluating bag-of-visual-words representations in scene classification," in *International Workshop on Multimedia Information Retrieval*. ACM, 2007, pp. 197–206.

[30] Y. Ke, R. Sukthankar, and M. Hebert, "Volumetric features for video event detection," *International Journal of Computer Vision*, vol. 88, no. 3, pp. 339–362, 2010.

[31] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[32] G. Cawley, N. Talbot, and M. Girolami, "Sparse multinomial logistic regression via bayesian l1 regularisation," *Advances in Neural Information Processing Systems*, vol. 19, p. 209, 2007.

[33] C. Hou, F. Nie, D. Yi, and Y. Wu, "Feature selection via joint embedding learning and sparse regression," in *International Joint Conference on Artificial Intelligence*. AAAI Press, 2011, pp. 1324–1329.

[34] Q. Gu, Z. Li, and J. Han, "Joint feature selection and subspace learning," in *International Joint conference on Artificial Intelligence*. AAAI Press, 2011, pp. 1294–1299.

[35] S. Nilufar, N. Ray, and H. Zhang, "Object detection with dog scale-space: A multiple kernel learning approach," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3744–3756, 2012.

[36] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40–51, 2007.

[37] Y. Yang, F. Wu, D. Xu, Y. Zhuang, and L.-T. Chia, "Cross-media retrieval using query dependent search methods," *Pattern Recognition*, vol. 43, no. 8, pp. 2927–2936, 2010.

[38] J. Duchon, "Splines minimizing rotation-invariant semi-norms in sobolev spaces," *Constructive Theory of Functions of Several Variables*, pp. 85–100, 1977.

[39] J. W. Demmel, *Applied numerical linear algebra*. SIAM, 1997.

[40] A. Loui, J. Luo, S. Chang, D. Ellis, W. Jiang, L. Kennedy, K. Lee, and A. Yanagawa, "Kodak's consumer video benchmark data set: concept definition and annotation," in *International Workshop on Multimedia Information Retrieval*. ACM, 2007, pp. 245–254.

[41] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local svm approach," in *International Conference on Pattern Recognition*, vol. 3. IEEE, 2004, pp. 32–36.

[42] J. Liu, J. Luo, and M. Shah, "Recognizing realistic actions from videos in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1996–2003.

[43] A. Yanagawa, S. Chang, L. Kennedy, and W. Hsu, "Columbia universitys baseline detectors for 374 lscom semantic visual concepts," *Columbia University ADVENT Technical Report*, 2007.

[44] D. Lewis, "Evaluating text categorization," in *Speech and Natural Language Workshop*, 1991, pp. 312–318.

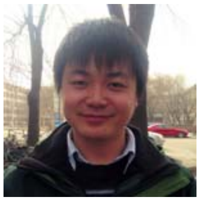
[45] T. Fawcett, "An introduction to roc analysis," *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.

[46] S. Wang, Y. Yang, Z. Ma, X. Li, C. Pang, and A. Hauptmann, "Action recognition by exploring data distribution and feature correlation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1370–1377.

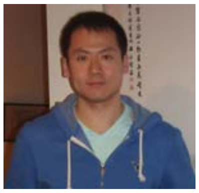
[47] H. Abdi and L. Williams, "Principal component analysis," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 433–459, 2010.

Yahong Han received the Ph.D. degree from Zhejiang University, Hangzhou, China. He is currently an Associate Professor with the School of Computer Science and Technology, Tianjin University, Tianjin, China. His current research interests include multimedia analysis, retrieval, and machine learning.

Yi Yang received the Ph.D degree in Computer Science from Zhejiang University, Hangzhou, China, in 2010. He is now a DECRA fellow with the University of Queensland, Brisbane, Australia. Prior to that, he was a Postdoctoral research fellow at the school of computer science, Carnegie Mellon University, Pittsburgh, PA. His research interests include machine learning and its applications to multimedia content analysis and computer vision, e.g. multimedia indexing and retrieval, surveillance video analysis, video semantics understanding, etc.



Yan Yan received the B.E. degree in computer science and technology from Tianjin University, Tianjin, China, in 2013. He is currently a master student at The University of Queensland. His current research interests include multimedia and computer vision.



Zhigang Ma received the Ph.D. in computer science from University of Trento, Trento, Italy, in 2013. His research interests include machine learning and its application to computer vision and multimedia analysis.

Nicu Sebe (M'01-SM'11) received the Ph.D. in computer science from Leiden University, Leiden, The Netherlands, in 2001. Currently, he is with the Department of Information Engineering and Computer Science, University of Trento, Italy, where he is leading the research in the areas of multimedia information retrieval and human-computer interaction in computer vision applications. He was a General Co-Chair of the IEEE Automatic Face and Gesture Recognition Conference, FG 2008 and ACM Multimedia 2013, and a program chair of ACM International Conference on Image and Video Retrieval (CIVR) 2007 and 2010, ACM Multimedia 2007 and 2011. He is a program chair of ECCV 2016 and ICCV 2017. He is a senior member of IEEE and of ACM and a fellow of IAPR.

Xiaofang Zhou received the BS and MS degrees in computer science from Nanjing University, China, in 1984 and 1987, respectively, and the PhD degree in computer science from The University of Queensland, Australia, in 1994. He is a professor of computer science with The University of Queensland. He is the head of the Data and Knowledge Engineering Research Division, School of Information Technology and Electrical Engineering. He is the director of ARC Research Network in Enterprise Information Infrastructure (EII) and a chief investigator of ARC Centre of Excellence in Bioinformatics. He is also an specially appointed Adjunct Professor at Soochow University, China. From 1994 to 1999, he was a senior research scientist and project leader in CSIRO. His research is focused on ending effective and efficient solutions to managing integrating and analyzing very large amounts of complex data for business and scientific applications. His research interests include spatial and multimedia databases, high performance query processing, web information systems, data mining, bioinformatics, and e-research. He is a senior member of the IEEE.