2021 IEEE International Conference on Multimedia and Expo (ICME) | 978-1-6654-3864-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICME51207.2021.9428428

TENSOR-BASED UNSUPERVISED MULTI-VIEW FEATURE SELECTION FOR IMAGE RECOGNITION

Yongshan Zhang[1,2]*, Xinxin Wang*[1,2]*, Zhihua Cai*[1]*, Yicong Zhou*[2] *and Philip S. Yu*[3]

[1]School of Computer Science, China University of Geosciences, Wuhan 430074, China
[2]Department of Computer and Information Science, University of Macau, Macau 999078, China
[3]Department of Computer Science, University of Illinois at Chicago, IL 60607, USA
yszhang.cug@gmail.com, {wangxinxin, zhcai}@cug.edu.cn
yicongzhou@um.edu.mo, psyu@cs.uic.edu

ABSTRACT

In image analysis, image samples from multiple sources may contain noisy features. Due to the difficulty of obtaining label information and complex intrinsic structures, performing unsupervised feature selection on multi-view data is a challenging problem. Most existing unsupervised multi-view feature selection methods may explore only the inter-view correlations at the view-level, and ignore the explicit correlations between features across multiple views. In this paper, we propose a tensor-based unsupervised multi-view feature selection (TUFS) method. Specifically, TUFS efficiently explores the full-order interactions among multi-view data without physically building a tensor. Besides, multiple local geometric structures for different views are constructed to facilitate unsupervised feature selection. To solve the proposed model, we design an alternating optimization algorithm. Experiments and comparisons on three image datasets demonstrate that the proposed TUFS yields better performance over the state-of-the-art methods.

***Index Terms*—** Multi-view learning, unsupervised feature selection, tensor factorization, image recognition

1. INTRODUCTION

In many applications, the data to be analyzed is naturally represented by multiple representations from heterogeneous sources. For example, image data can be described by different feature descriptors, e.g., wavelet texture (WT), edge direction histogram (EDH) and color moment (CM). Multi-view learning is a typical scenario [1], which focuses on learning from data represented by multiple feature sets. The raw data from different views may contain noisy, irrelevant and redundant features, which may degrade the accuracy of image recognition. Due to the difficulty of obtaining the label information, how to select representative features from multi-view data in an unsupervised manner is a challenging problem.

In the literature, existing methods of unsupervised multi-view feature selection mainly rely on two different strategies, including the concatenating and cross-view strategies. Methods in the first strategy directly employ traditional single-view models by concatenating all features from multiple views as the input [2]. Representative algorithms include Laplacian score (LapScor) [3], multi-cluster feature selection (MCFS) [4] and unsupervised discriminative feature selection (UDFS) [5]. Although these methods have been proved to be effective in many cases, they ignore the underlying correlations between different views. It is apparent that they are not suitable for multi-view data.

In order to take full advantage of the complementary information from multi-view data, methods in the cross-view strategy exploit correlations and interactions between different views to choose a representative feature subset [6]. Representative algorithms include multi-view feature selection (MVFS) [7], robust multi-view feature selection (RMFS) [8] and adaptive similarity embedding for unsupervised multi-view feature selection (ASE-UMFS) [9]. These methods routinely assign one weight for one feature view representation, where the inter-view correlations are only exploited at the view-level, whereas the explicit correlations between features across multiple views are overlooked. It is desired that the full-order structural information are considered in unsupervised multi-view feature selection models.

Motivated by these observations, in this paper, we propose a tensor-based unsupervised multi-view feature selection (TUFS) method with the aim of exploring the unsupervised heterogeneous data fusion and feature selection. With the help of tensor factorization, TUFS explores the full-order interactions among multi-view data, without the need to con-

This work is supported in part by the National Nature Science Foundation of China under Grant 61703355, by the Natural Science Foundation of Hubei Province of China under Grant 2020CFB328, by the Science and Technology Development Fund, Macau SAR (File no. 189/2017/A3), by University of Macau (File no. MYRG2018-00136-FST) and by the Fundamental Research Funds for the Central Universities, China University of Geosciences (Wuhan) under Grant CUG200116.

struct the input tensor physically. Additionally, multiple local geometric structures for different views are constructed to facilitate unsupervised feature selection. To solve the proposed model, we devise an alternating optimization algorithm. Experiments and comparisons on three image datasets demonstrate the superiority and potential of the proposed TUFS.

The structure of this paper is organized as follows. Section 2 introduces important notations and problem description. Section 3 firstly presents the formulation of the proposed TUFS model and then provides an alternative optimization algorithm. Experiments and comparisons are demonstrated in Section 4. Finally, we conclude the paper in Section 5.

2. PRELIMINARY

2.1. Tensor Algebra and Notation

The important symbols used throughout this paper are summarized in Table 1. Tensors are higher order arrays that generalize the notion of vectors (first order) and matrices (second order). Following [10], an Mth order tensor is denoted by $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_M}$ and its elements by $x_{i_1,\cdots,i_M}$. All vectors are column vectors unless otherwise specified. For an arbitrary matrix $\mathbf{X} \in \mathbb{R}^{I \times J}$, the ith row and jth column are denoted by $\mathbf{x}^i$ and $\mathbf{x}_j$, respectively.

The Hadamard product is the element-wise matrix product. An important property of Hadamard product is $\mathbf{a} * \mathbf{b} = diag(\mathbf{a})\mathbf{b}$. The Kronecker product is an operation resulting in a block matrix. An important application of Kronecker product is to rewrite the matrix equation $\mathbf{AXB} = \mathbf{C}$ into the equivalent vector equation $(\mathbf{B}_{\mathbf{A}}^T)vec(\mathbf{X}) = vec(\mathbf{C})$. The inner product of two tensors $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^{I_1 \times \cdots \times I_M}$ is defined by $\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1=1}^{I_1} \cdots \sum_{i_1=1}^{I_M} x_{i_1,\cdots,i_M} y_{i_1,\cdots,i_M}$. The outer product of vectors $\mathbf{x}^{(m)} \in \mathbb{R}^{I_m}$ for $m \in [1 : M]$ is an Mth order tensor and defined element-wise by $(\mathrm{x}^{(1)} \circ \cdots \circ \mathrm{x}^{(M)})_{i_1,\cdots,i_M} = x_{i_1}^{(1)} \cdots x_{i_M}^{(M)} = \prod_{m=1}^{M} x_{i_m}^{(m)}$ for all values of the indices. In particular, for $\mathcal{X} = \mathbf{x}^{(1)} \cdots \circ \mathbf{x}^{(M)}$ and $\mathcal{Y} = \mathbf{y}^{(1)} \circ \cdots \circ \mathbf{y}^{(M)}$, it holds that

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \prod_{m=1}^{M} \langle \mathbf{x}^{(m)}, \mathbf{y}^{(m)} \rangle = \prod_{m=1}^{M} {\mathbf{x}^{(m)}}^{\mathrm{T}} \mathbf{y}^{(m)}. \quad (1)$$

2.2. Problem Description

Given a multi-view dataset $\{\mathbf{X}^{(v)}\}_{v=1}^{V}$ with V views, where $\mathbf{X}^{(v)} \in \mathbb{R}^{D_v \times N}$ represents the vth view feature matrix with N instances and D_v features. Our goal of unsupervised multi-view feature selection is to select a subset of important features from multiple feature spaces by leveraging the complementary information and exploiting the full-order interactions among different views with the help of tensor manipulation.

Table 1. List of important symbols.

Symbol	Definition and description
$x, \mathbf{x}, \mathbf{X}, \mathcal{X}$	denotes a scale, a vector, a matrix, a tensor
$\langle \cdot, \cdot \rangle$	denotes inner product
$\circ$	denotes outer product
$*$	denotes Hadamard (elementwise) product
$\otimes$	denotes Kronecker product
A_{B}	denotes $\mathrm{A} \otimes \mathrm{B}$
$\mathrm{Tr}(\cdot)$	denotes trace function
$diag(\cdot)$	denotes a diagonal matrix
$vec(\cdot)$	denotes the column stacking operator
$\|\cdot\|_F$	denotes Frobenius norm of matrix or tensor
$\|\cdot\|_{2,1}$	denotes $\ell_{2,1}$ norm of matrix

3. PROPOSED MODEL

In this section, we first introduce the formulation of the proposed model, and then provide an optimization algorithm as solution.

3.1. Model Formulation

Unsupervised feature selection methods select a subset of representative features without using any label information [11]. After reviewing existing models, we find that most of them are formulated by a linear regression-like objective [12]. Given an input matrix $\mathbf{X} \in \mathbb{R}^{D \times N}$ and a cluster indicator matrix $\mathbf{F} \in \mathbb{R}^{N \times K}$, the linear model is given by

$$\left\|\mathbf{X}^T\mathbf{W} - \mathbf{F}\right\|_F^2 = \sum_{n=1}^{N}\sum_{k=1}^{K}\left(\mathbf{x}_n^T\mathbf{w}_k - f_{n,k}\right)^2, \quad (2)$$

where $\mathbf{W} \in \mathbb{R}^{D \times K}$ is a weight matrix, $\mathbf{x}_n$ and $\mathbf{w}_k$ are the nth and kth column of $\mathbf{X}$ and $\mathbf{W}$, and $f_{n,k}$ is the nth row and kth column element of $\mathbf{F}$. We denote the indicator vector as $\mathbf{e}_k = [0, \cdots, 0, 1, 0, \cdots, 0]^T \in \mathbb{R}^K$, where the only nonzero element with constant value 1 is the kth entry. From Eq. (2), we have

$$f_{n,k} = \mathbf{x}_n^T\mathbf{w}_k = \mathbf{x}^T\mathbf{W}\mathbf{e}_k = \langle \mathbf{x}_n \circ \mathbf{e}_k, \mathbf{W} \rangle. \quad (3)$$

Similarly, assume that there are two views, Eq. (3) can be rewritten as follows:

$$f_{n,k} = {\mathbf{x}_n^{(1)}}^T\mathbf{W}_k\mathbf{x}_n^{(2)} = \langle \mathbf{x}_n^{(1)} \circ \mathbf{x}_n^{(2)} \circ \mathbf{e}_k, \mathcal{W} \rangle, \quad (4)$$

where $\mathcal{W} \in \mathbb{R}^{D_1 \times D_2 \times K}$ is the weight tensor.

However, only the highest-order interactions are explored in this way. To explain the data sufficiently, the lower-order interactions should be considered. Hence, we nest all interactions up to full-order:

$$f_{n,k} = w_k + \sum_{v=1}^{2} {\mathbf{x}_n^{(v)}}^T\mathbf{w}_k^{(v)} + {\mathbf{x}_n^{(1)}}^T\mathbf{W}_k\mathbf{x}_n^{(2)}. \quad (5)$$

To achieve full-order interactions, an extra feature with constant value 1 is added to the input vector $\mathbf{x}_n^{(v)}$, i.e., $\mathbf{z}_n^{(v)} = [\mathbf{x}_n^{(v)}; 1] \in \mathbb{R}^{D_v+1}$. Then Eq. (5) can be rewritten as

$$f_{n,k} = \langle \mathbf{z}_n^{(1)} \circ \mathbf{z}_n^{(2)} \circ \mathbf{e}_k, \mathcal{W} \rangle = \langle \mathcal{Z}_n \circ \mathbf{e}_k, \mathcal{W} \rangle. \quad (6)$$

Following this recipe, Eq. (6) can be easily extended to the problem with more views. Given a V-view dataset, $\mathcal{Z}_n = \mathbf{z}_n^{(1)} \circ \cdots \circ \mathbf{z}_n^{(V)} \in \mathbb{R}^{(D_1+1)\times\cdots\times(D_V+1)}$ is the tensor representation from the inputs and $\mathcal{W} \in \mathbb{R}^{(D_1+1)\times\cdots\times(D_V+1)\times K}$ is the weight tensor to be learnt. The number of parameters in $\mathcal{W}$ is $K\prod_{v=1}^{V}(D_v+1)$. It is apparent that the extended model for V views suffers from the over-parameterized problem. Besides, all interactions for different views and different clusters are not possible to jointly explore from the extended model. To solve these problems, we assume that there is a low rank in the effect of interactions and $\mathcal{W}$ can be factorized by CANDECOMP/PARAFAC (CP) decomposition [13] as

$$\mathcal{W} = \sum_{r=1}^{R} \mathbf{w}_r^{(1)} \circ \cdots \circ \mathbf{w}_r^{(V+1)} = [\![\mathbf{W}^{(1)}, \cdots, \mathbf{W}^{(V+1)}]\!], \quad (7)$$

where $\mathbf{W}^{(v)} \in \mathbb{R}^{(D_v+1)\times R}$ for $v \in [1:V]$ is the vth view factor matrix, $\mathbf{W}^{(V+1)} \in \mathbb{R}^{K\times R}$ is the cluster weight matrix, $\mathbf{w}_r^{(v)}$ is the column vector, and R is the number of factors.

With the above information, Eq. (6) can be firstly generalized to V views, and then rewritten according to Eq. (1). Thus, we have

$$\begin{aligned} f_{n,k} &= \sum_{r=1}^{R} \langle \mathbf{z}_r^{(1)} \circ \cdots \circ \mathbf{z}_r^{(V)} \circ \mathbf{e}_k, \mathbf{w}_r^{(1)} \circ \cdots \circ \mathbf{w}_r^{(V+1)} \rangle \\ &= \sum_{r=1}^{R} w_{k,r}^{(V+1)} (\mathbf{z}_n^{(1)^T} \mathbf{w}_r^{(1)}) \cdots (\mathbf{z}_n^{(V)^T} \mathbf{w}_r^{(V)}) \\ &= ((\mathbf{z}_n^{(1)^T} \mathbf{W}^{(1)}) * \cdots * (\mathbf{z}_n^{(V)^T} \mathbf{W}^{(V)})) \mathbf{w}^{(V+1)k^T}. \end{aligned} \quad (8)$$

By using Eq. (8), we can derive the multi-view setting of Eq. (2) to explore the full-order interactions. The corresponding formulation is given by

$$\|(\prod_{v=1}^{V} *(\mathbf{Z}^{(v)^{\mathrm{T}}} \mathbf{W}^{(v)})) \mathbf{W}^{(V+1)^{\mathrm{T}}} - \mathbf{F}\|_F^2. \quad (9)$$

To perform feature selection, an $\ell_{2,1}$-norm regularization is imposed on each weight matrix $\mathbf{W}^{(v)}$. This will ensure the sparsity of $\mathbf{W}^{(v)}$ in row, making it suitable for feature selection. Thus, we have

$$\begin{aligned} \min_{\mathbf{W}^{(v)},\mathbf{F}} &\|(\prod_{v=1}^{V} *(\mathbf{Z}^{(v)^T} \mathbf{W}^{(v)})) \mathbf{W}^{(V+1)^T} - \mathbf{F}\|_F^2 \\ &+ \gamma \sum_{v=1}^{V+1} \|\mathbf{W}^{(v)}\|_{2,1}, \quad s.t.\ \mathbf{F}^T\mathbf{F} = \mathbf{I}, \mathbf{F} \geq 0, \end{aligned} \quad (10)$$

where $\mathbf{F}^T\mathbf{F} = \mathbf{I}$ and $\mathbf{F} \geq 0$ is the orthogonal and nonnegative constraint, and γ is the parameter to control the sparsity.

In reality, it is vital to preserve the local geometric structures of data for unsupervised feature selection. The corresponding optimization can be formulated as $\mathrm{Tr}(\mathbf{F}^T\mathbf{L}\mathbf{F})$, where $\mathbf{L} = \mathbf{A} - \mathbf{S}$ is a Laplacian matrix and $\mathbf{A}$ is a diagonal matrix with $a_{n,n} = \sum_{j=1}^{N} s_{n,j}$. $\mathbf{S} \in \mathbb{R}^{N\times N}$ is the similarity matrix learnt by a k-nearest neighbor graph from $\mathbf{X}$. For multi-view data, the local geometric structures of different views should be introduced to exploit their relations by the shared cluster indicators. Finally, we formulate the objective function of the proposed TUFS model as

$$\begin{aligned} \min_{\mathbf{W}^{(v)},\mathbf{F},\alpha_v} &\|(\prod_{v=1}^{V} *(\mathbf{Z}^{(v)^T} \mathbf{W}^{(v)})) \mathbf{W}^{(V+1)^T} - \mathbf{F}\|_F^2 \\ &+ \gamma \sum_{v=1}^{V+1} \|\mathbf{W}^{(v)}\|_{2,1} + \lambda \sum_{v=1}^{V} \alpha_v \mathrm{Tr}(\mathbf{F}^T\mathbf{L}^{(v)}\mathbf{F}) \\ &s.t.\ \mathbf{F}^T\mathbf{F} = \mathbf{I}, \mathbf{F} \geq 0. \end{aligned} \quad (11)$$

where λ is the parameter for trade-off, and α_v is the weight to measure the contribution of the local geometric structures in each view. Once $\mathbf{W}^{(v)}$ is learned, feature selection can be realized by ranking features according to $\|(\mathbf{w}^i)^{(v)}\|_2$ $(i = 1, \ldots, D_v)$ in a descending order.

3.2. Optimization Algorithm

The objective function of TUFS is not convex, and solving Eq. (11) directly is difficult. Therefore, we devise an optimization algorithm by alternatively updating one variable while fixing the others. For convenience, we denote $\mathbf{\Pi} = \prod_{v=1}^{V} *(\mathbf{Z}^{(v)^T} \mathbf{W}^{(v)}) \in \mathbb{R}^{N\times R}$ as the embedding matrix from all the views and $\mathbf{\Pi}^{(-v)} = \prod_{v'\neq v}^{V} *(\mathbf{Z}^{(v')^{\mathrm{T}}} \mathbf{W}^{(v')}) \in \mathbb{R}^{N\times R}$ as the embedding matrix from all other views except the vth view. In addition, due to the non-smooth regularization term of $\ell_{2,1}$-norm, following [14], we relax $\|\mathbf{W}^{(v)}\|_{2,1}$ by $\mathrm{Tr}(\mathbf{W}^{(v)^T}\mathbf{P}^{(v)}\mathbf{W}^{(v)})$, where $\mathbf{P}^{(v)}$ is a diagonal matrix with diagonal elements $p_{i,i}^{(v)} = \frac{1}{2\|\mathbf{w}_i^{(v)}\|_2}$. The pseudocode of TUFS is presented in Algorithm 1.

Update $\mathbf{W}^{(v)}$ ($1 \leq v \leq V$): With other fixed variables, the optimization w.r.t $\mathbf{W}^{(v)}$ becomes

$$\begin{aligned} \min_{\mathbf{W}^{(v)}} &\|(\mathbf{\Pi}^{(-v)} * (\mathbf{Z}^{(v)^{\mathrm{T}}} \mathbf{W}^{(v)})) \mathbf{W}^{(V+1)^{\mathrm{T}}} - \mathbf{F}\|_F^2 \\ &+ \gamma \mathrm{Tr}(\mathbf{W}^{(v)^T} \mathbf{P}^{(v)} \mathbf{W}^{(v)}). \end{aligned} \quad (12)$$

Taking the derivative of Eq. (12) w.r.t $\mathbf{W}^{(v)}$ and setting it to zero, we have

$$\begin{aligned} &\mathbf{Z}^{(v)}(\mathbf{\Pi}^{(-v)} * ((\mathbf{\Pi}^{(-v)} * (\mathbf{Z}^{(v)^T} \mathbf{W}^{(v)})) \mathbf{W}^{(V+1)^T} \mathbf{W}^{(V+1)})) \\ &- \mathbf{Z}^{(v)}(\mathbf{\Pi}^{(-v)} * (\mathbf{F}\mathbf{W}^{(V+1)})) + \gamma \mathbf{P}^{(v)} \mathbf{W}^{(v)} = 0. \end{aligned} \quad (13)$$

To solve Eq. (13), we state the following theorem, and the proof can be found in [15].

Algorithm 1 TUFS

Input: Multi-view data $\{\mathbf{X}^{(v)}\}_{v=1}^{V}$, where $\mathbf{X}^{(v)} \in \mathbb{R}^{D_v \times N}$, number of clusters K, number of factors R, and regularized parameters γ and λ;

Output: The weight matrices $\{\mathbf{W}^{(v)}\}_{v=1}^{V}$;

Initialize $\mathbf{W}^{(v)}$, $\mathbf{F}$ and $\alpha_v = 1/V$;
Allocate $\mathbf{Z}^{(v)} = [\mathbf{X}^{(v)}; 1]$;
repeat
  for $v = 1$ to V **do**
    Calculate the diagonal matrix $\mathbf{P}^{(v)}$ by $\mathbf{W}^{(v)}$;
    Update $\mathbf{W}^{(v)}$ by solving Eq. (13);
  end for
  Calculate the diagonal matrix $\mathbf{P}^{(V+1)}$ by $\mathbf{W}^{(V+1)}$;
  Update $\mathbf{W}^{(V+1)}$ by solving Eq. (17);
  Update $\mathbf{F}$ via Eq. (20);
  for $v = 1$ to V **do**
    Update $\alpha^{(v)} = 1/(2 * \sqrt{\mathbf{F}^T\mathbf{L}^{(v)}\mathbf{F}})$;
  end for
until Convergence

Theorem 1. *Given any matrices* $\mathbf{A} \in \mathbb{R}^{M\times N}$, $\mathbf{B} \in \mathbb{R}^{N\times R}$, $\mathbf{C} \in \mathbb{R}^{R\times R}$, $\mathbf{D} \in \mathbb{R}^{M\times M}$ *and* $\mathbf{E} \in \mathbb{R}^{M\times R}$, *the solution to the following equation*

$$\mathbf{A}(\mathbf{B} * ((\mathbf{B} * (\mathbf{A}^T\mathbf{X}))\mathbf{C})) + \mathbf{D}\mathbf{X} = \mathbf{E} \tag{14}$$

is equivalent to the solution to the following equation

$$\mathbf{H}vec(\mathbf{X}) = vec(\mathbf{E}), \tag{15}$$

where $\mathbf{H} = \mathbf{I}_\mathbf{A} diag(vec(\mathbf{B}))\mathbf{C}_\mathbf{I}^T diag(vec(\mathbf{B}))\mathbf{I}_\mathbf{A} + \mathbf{I}_\mathbf{D}$.

According to Theorem 1, letting $\mathbf{A} = \mathbf{Z}^{(v)}$, $\mathbf{B} = \mathbf{\Pi}^{(-v)}$, $\mathbf{C} = \mathbf{W}^{(V+1)^T}\mathbf{W}^{(V+1)}$, $\mathbf{D} = \gamma\mathbf{P}^{(v)}$, and $\mathbf{E} = \mathbf{Z}^{(v)}(\mathbf{\Pi}^{(-v)} * (\mathbf{F}\mathbf{W}^{(V+1)}))$, we can formulate Eq. (13) in the form of Eq. (15). Since $\mathbf{H}$ is invertible, the solution in the vector form is given by $vec(\mathbf{W}^{(v)}) = \mathbf{H}^{-1}vec(\mathbf{E})$.

Update $\mathbf{W}^{(V+1)}$: With other fixed variables, the optimization w.r.t $\mathbf{W}^{(V+1)}$ becomes

$$\min_{\mathbf{W}^{(V+1)}} \|\mathbf{\Pi}\mathbf{W}^{(V+1)^T} - \mathbf{F}\|_F^2 + \gamma \mathrm{Tr}(\mathbf{W}^{(V+1)^T}\mathbf{P}^{(V+1)}\mathbf{W}^{(V+1)}). \tag{16}$$

Taking the derivative of Eq. (16) w.r.t $\mathbf{W}^{(V+1)}$ and setting it to zero, we have

$$\mathbf{W}^{(V+1)}\mathbf{\Pi}^T\mathbf{\Pi} + \gamma\mathbf{P}^{(V+1)}\mathbf{W}^{(V+1)} = \mathbf{F}^T\mathbf{\Pi}. \tag{17}$$

Eq. (17) is the Sylvester equation, which can be solved by the lyap function in MATLAB.

Update F: With other fixed variables, by introducing the Lagrangian multiplier Φ, the Lagrange function for the optimization w.r.t $\mathbf{F}$ becomes

$$\begin{aligned}\min_{\mathbf{F}} \|\mathbf{\Pi}\mathbf{W}^{(V+1)^T} - \mathbf{F}\|_F^2 + \lambda\sum_{v=1}^{V}\alpha_v \mathrm{Tr}(\mathbf{F}^T\mathbf{L}^{(v)}\mathbf{F}) \\ + \frac{\beta}{2}\|\mathbf{F}^T\mathbf{F} - \mathbf{I}_c\|_F^2 + \mathrm{Tr}(\Phi\mathbf{F}^T).\end{aligned} \tag{18}$$

Fig. 1. Example images from the VOC07 dataset.

Taking the derivative of Eq. (18) w.r.t $\mathbf{F}$ and setting it to zero, we have

$$\mathbf{F} - \mathbf{\Pi}\mathbf{W}^{(V+1)^T} + \lambda\mathbf{M}\mathbf{F} + \beta\mathbf{F}\mathbf{F}^T\mathbf{F} - \beta\mathbf{F} + \Phi = 0. \tag{19}$$

According to the Karush-Kuhn-Tuckre (KKT) condition $\phi_{i,j}f_{i,j} = 0$, the update rule for $\mathbf{F}$ is given by

$$f_{i,j} = f_{i,j}\frac{[\mathbf{\Pi}\mathbf{W}^{(V+1)^T} + \beta\mathbf{F}]_{i,j}}{[\mathbf{F} + \lambda\mathbf{M}\mathbf{F} + \beta\mathbf{F}\mathbf{F}^T\mathbf{F}]_{i,j}}. \tag{20}$$

Update α_v: With other fixed variables, following [16, 17], the optimal solution of α_v is adaptively updated by the equation $\alpha_v = 1/(2 * \sqrt{\mathrm{Tr}(\mathbf{F}^T\mathbf{L}^{(v)}\mathbf{F})})$.

4. EXPERIMENTS

4.1. Experimental Setup

Datasets. In the experiments, three public image datasets are adopted for evaluation. The VOC07 dataset consists of 1424 different images, which are associated with 5 categories and described by three feature representations, including dense hue (DH), generalized search trees (GIST) and harris hue (HH). The OBJECT dataset contains 3467 distinct images, which are related to 7 categories and represented by two feature representations, including color moments (CM) and color correlogram (CORR). The SCENE dataset includes 5109 different images, which are affiliated to 10 categories and described by CM and CORR. Example images from the VOC07 image database are shown in Fig. 1.

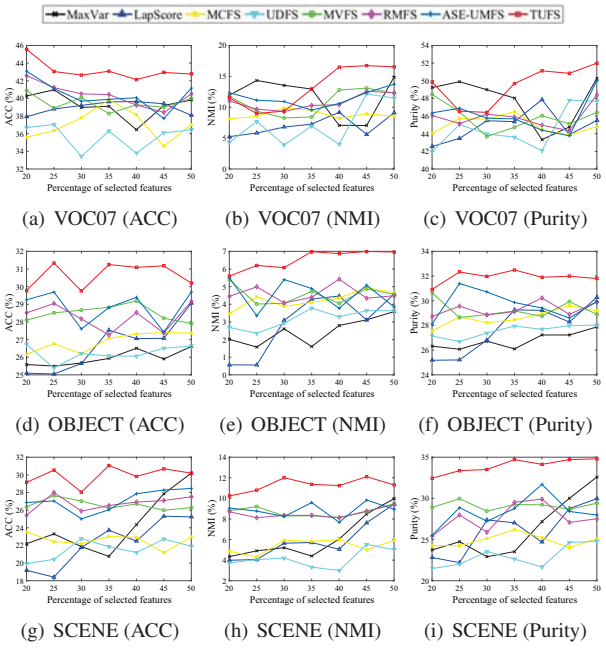
Comparison methods. To verify the effectiveness of TUFS, we use the state-of-the-art methods for comparison. MaxVar, LapScore [3], MCFS [4] and UDFS [5] are typical unsupervised single-view feature selection methods. They concatenate all features from multiple views as the input. MVFS [7], RMFS [8] and ASE-UMFS [9] are representative unsupervised multi-view feature selection methods.

Evaluation Metrics. To assess the quality of selected features, we adopt three popular metrics for evaluation, including clustering accuracy (ACC), normalized mutual information (NMI) and purity. For the above evaluation metrics, the larger the value, the better the clustering performance.

Parameter Settings. There are some important parameters in TUFS to be set in advance. The parameters γ and λ are both tuned from $\{10^{-2}, 10^{-1}, ..., 10^{2}\}$, while the parameter R is selected from $\{5, 10, \cdots, 30\}$. The numbers of selected features are varied as $\{20\%, 25\%, ..., 50\%\}$ of the size of

Table 2. Comparisons of different methods on three image datasets.

Dataset	Measure	MaxVar	LapScore	MCFS	UDFS	MVFS	RMFS	ASE-UMFS	TUFS
VOC07	ACC	40.98	39.64	39.85	37.04	40.89	42.59	43.09	**45.58**
	NMI	14.89	9.19	9.88	12.16	13.09	12.59	13.73	**16.73**
	Purity	50.28	47.86	46.51	47.79	48.46	48.40	50.01	**52.00**
OBJECT	ACC	26.61	29.06	27.40	26.72	29.19	29.16	29.70	**31.34**
	NMI	3.59	4.58	4.94	3.78	5.40	5.43	5.50	**6.99**
	Purity	27.88	30.29	29.61	28.04	30.64	30.23	31.39	**32.49**
SCENE	ACC	30.20	25.31	23.54	22.77	27.63	28.00	28.46	**31.06**
	NMI	9.97	9.44	5.99	5.53	9.49	9.40	9.84	**12.11**
	Purity	32.60	29.96	26.22	24.78	29.95	29.91	31.70	**34.80**



(a) VOC07 (ACC) (b) VOC07 (NMI) (c) VOC07 (Purity)

(d) OBJECT (ACC) (e) OBJECT (NMI) (f) OBJECT (Purity)

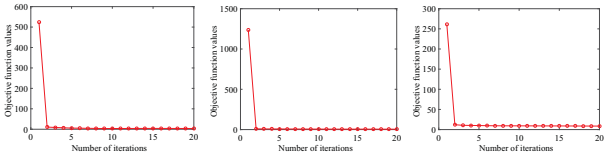
(g) SCENE (ACC) (h) SCENE (NMI) (i) SCENE (Purity)

Fig. 2. Comparisons of different methods w.r.t different numbers of selected features on three image datasets.

features. The multi-view learning with adaptive neighbours (MLAN) [18] is used to cluster samples based on the selected features. We repeat each experiment 20 times and report the average performance.

4.2. Experimental Results

Comparison of clustering results. The experimental results of different methods on the three image datasets are shown in Table 2. As can be seen from Table 2, TUFS consistently achieves superior results over other competing methods. Compared to MaxVar, LapScore, MCFS and UDFS, the enhancement of TUFS is obvious. For instance, on the VOC07 dataset, TUFS achieves more than 15% ACC improvement in average. On the OBJECT dataset, the average purity improvement of TUFS is also high at 15%. These observations confirm the superiority of TUFS than the single-view feature selection methods. In comparison with MVFS, RMFS and ASE-UMFS, TUFS also achieves competitive results. On the SCENE dataset, TUFS gets more than 10% ACC improvement and 14% purity improvement in average. This demonstrates the advantage of TUFS than the compared multi-view feature selection methods.

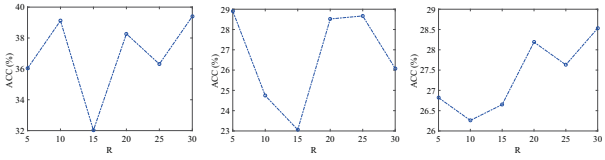


(a) VOC07 (b) OBJECT (c) SCENE

Fig. 3. The convergence curves of TUFS.

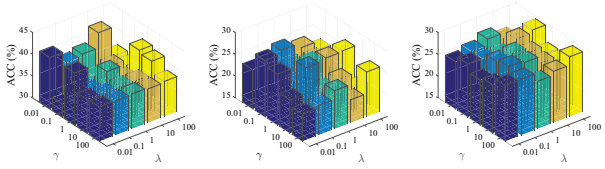
To further validate the effectiveness of TUFS, we conduct the experimental study with different numbers of selected features. Fig. 2 presents the comparison results on the three image datasets. On the OBJECT and SCENE datasets, TUFS consistently outperforms the competing methods with different numbers of selected features. On the VOC07 datset, TUFS obtains inferior results with a small number of selected features. Moreover, the performance of TUFS is much more stable when the percentage of selected features is greater than 35%. This indicates that by jointly utilizing tensor factorization and constructing the local geometric structures of different views, the quality of selected features is greatly improved, which benefits image recognition.

Convergence study. To verify the convergence of Algorithm 1, we experimentally study the convergence behavior of TUFS. The parameters γ, λ and R are fixed to be 0.1, 0.1 and 5. Fig. 3 presents the convergence learning curves. From Fig. 3, we find that the objective function values rapidly decrease at the first few iterations and converge within 10 iterations for all image datasets. This demonstrates that the proposed optimization algorithm is effective and converges quickly to solve the problem of Eq. (11).



(a) VOC07 (b) OBJECT (c) SCENE

Fig. 4. Performance variation of TUFS in terms of ACC w.r.t different values of R.



(a) VOC07 (b) OBJECT (c) SCENE

Fig. 5. Performance variation of TUFS in terms of ACC w.r.t different values for the parameters γ and λ.

Parameter sensitivity study. There are three important parameters R, γ and λ in TUFS. To study the sensitivity of these parameters, we run with different values for $R \in \{5, 10, ..., 30\}$ and $\gamma, \lambda \in \{10^{-2}, 10^{-1}, ..., 10^{2}\}$. As shown in Fig. 4, with the fixed parameters $\gamma = 0.1$ and $\lambda = 0.1$, TUFS achieves best performance when $R = 30$ on the VOC07 and SCENE datasets, while $R = 5$ on the OBJECT dataset. From Fig. 5, by setting $R = 5$, different settings for the parameters γ and λ show different results. TUFS obtains the best result on the VOC07 dataset when $\gamma = 0.01$ and $\lambda = 10$. Therefore, it is necessary to set suitable values of the three parameters for TUFS.

5. CONCLUSION

In this paper, we proposed a tensor-based unsupervised multi-view feature selection (TUFS) method for image recognition. To select representative features, TUFS uses tensor factorization and constructs the local geometric structures of different views, which can explore the full-order interactions and learn the latent structures for multiple views, without physically constructing the higher-order tensor data. An alternative optimization algorithm is designed to solve the proposed TUFS model. Experiments and comparisons on three image datasets demonstrated that TUFS is effective for unsupervised multi-view feature selection and outperforms the state-of-the-art methods.

6. REFERENCES

[1] C. Xu, D. Tao, and C. Xu, "Multi-view learning with incomplete views," *IEEE TIP*, vol. 24, no. 12, pp. 5812–5825, 2015.

[2] C. Tang, M. Bian, X. Liu, M. Li, H. Zhou, P. Wang, and H. Yin, "Unsupervised feature selection via latent representation learning and manifold regularization," *Neural Networks*, vol. 117, pp. 163–178, 2019.

[3] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *NIPS*, 2006, pp. 507–514.

[4] D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for multi-cluster data," in *KDD*, 2010, pp. 333–342.

[5] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou, "$\ell_{2,1}$-norm regularized discriminative feature selection for unsupervised learning," in *IJCAI*, 2011, pp. 1589–1594.

[6] C. Tang, X. Zhu, X. Liu, and L. Wang, "Cross-view local structure preserved diversity and consensus learning for multi-view unsupervised feature selection," in *AAAI*, 2019, vol. 33, pp. 5101–5108.

[7] J. Tang, X. Hu, H. Gao, and H. Liu, "Unsupervised feature selection for multi-view data in social media," in *SDM*, 2013, pp. 270–278.

[8] H. Liu, H. Mao, and Y. Fu, "Robust multi-view feature selection," in *ICDM*, 2016, pp. 281–290.

[9] Y. Wan, S. Sun, and C. Zeng, "Adaptive similarity embedding for unsupervised multi-view feature selection," *IEEE TKDE*, 2020, DOI:10.1109/TKDE.2020.2969860.

[10] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[11] J. Guo, Y. Quo, X. Kong, and R. He, "Unsupervised feature selection with ordinal locality," in *ICME*, 2017, pp. 1213–1218.

[12] L. Shi, L. Du, and Y.-D. Shen, "Robust spectral learning for unsupervised feature selection," in *ICDM*, 2014, pp. 977–982.

[13] C.-T. Lu, L. He, W. Shao, B. Cao, and P. S Yu, "Multilinear factorization machines for multi-task multi-view learning," in *WSDM*, 2017, pp. 701–709.

[14] Y. Zhang, J. Wu, Z. Cai, and P. S. Yu, "Multi-view multi-label learning with sparse feature selection for image annotation," *IEEE TMM*, vol. 22, no. 11, pp. 2844–24857, 2020.

[15] L. He, C. T. Lu, Y. Chen, J. Zhang, L. Shen, P. S. Yu, and F. Wang, "A self-organizing tensor architecture for multi-view clustering," in *ICDM*, 2018, pp. 1007–1012.

[16] F. Nie, J. Li, and X. Li, "Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification.," in *IJCAI*, 2016, pp. 1881–1887.

[17] Y. Zhang, J. Wu, C. Zhou, Z. Cai, J. Yang, and P. S Yu, "Multi-view fusion with extreme learning machine for clustering," *ACM TIST*, vol. 10, no. 5, pp. 1–23, 2019.

[18] F. Nie, G. Cai, and X. Li, "Multi-view clustering and semi-supervised classification with adaptive neighbours," in *AAAI*, 2017, pp. 2408–2414.