Joint Feature-Space-Aware and Label-Space-Aware Feature Learning for Aerial Scene Classification

Anonymous CVPR submission

Paper ID 1225

Abstract

The performance of scene classification heavily relies the spatial and structural features extracted from high spatial resolution remote sensing images. Exiting approaches, however, are limited to adequately exploit discriminative object-based feature and capture high-level semantic structures. Aiming to learn more representative feature and alleviate the semantic gap between the object-based feature and feature extraction, in this paper, we present a feature learning framework which joints the graph learning, latent feature space constraint, and label space regularization for high-resolution aerial image classification. To describe the relationships among scene images, we construct the adaptive graph that is embedded into the constrained joint space for features and labels. To deal with out-of-sample data, linear regression is adopted to project semi-supervised classification results into the linear classifier. The learning efficiency is gained by minimizing the objective function via the linearized alternating direction method with adaptive penalty. We test our method on three widely used aerial scene image datasets. Experimental results demonstrate the superior performance of our method over the state-of-the-art algorithms in aerial scene images classification.

1. Introduction

With the availability of huge volumes of high spatial resolution remote sensing images (HSR-RSIs) produced by the satellites and airborne sensors, HSR-RSIs analysis has been an active research topic in remote sensing and computer vision fields [7, 8, 9, 10, 11, 12]. Since HSR-RSIs are obtained with different locations, time and even different satellite or airborne sensors, there are naturally large variations among different kinds of feature representations. The same category images have also appeared large differences. Many existing approaches for scene classification combine spatial and structural features without adequately exploiting discriminative object-based feature for aerial scene images, thus these methods cannot well recognize ground objects in scene images. It is necessary to develop efficient and effective methods for scene images classification.

Since we focus on the task of scene classification, one key problem that should be considered is how to extract discriminative features of scene images. For feature extraction, there are two issues should be solved: the target of feature extraction and the approach of feature extraction.

Towards HSR-RSIs, the feature extraction methods have evolved from per-pixel-oriented methods to object-oriented methods [11]. The main reason is that the object-based feature can achieve more accurate classification results than the per-pixel-oriented way [13, 14, 15, 16, 17]. The suitable feature extraction methods need to be developed to describe the object-based features. SIFT [19], HOG [20] and GLCM [18] are adopted to represent the low-level image features. The features can be further processed using the bag of words [21] or sparse coding [22] to form middle-level features. Since lack of the definition and expression of high-level semantic information, the low-level and middle-level features can not well represent the object-based features in HSR-RSIs. Recently, deep learning algorithms [35, 36, 37, 38, 39] have been employed to extract high-level features for image classification. However, these methods usually need large training samples to obtain more accurate classification results. In the aerial scene classification, collection of huge image labels in the scene dataset is an expensive task. The convergence speed of the objective function in deep learning methods also cannot be overlooked in the training stage. There is a semantic gap between object-based features and these feature extraction methods. Hence, it's a challenge to develop a more competent feature extraction method to describe object-based features of aerial scene images.

In scene classification or other classification applications, feature extraction methods also play a very important role in improving classification accuracies. In [7], based on dense low-level feature descriptors, an unsupervised feature learning approach was developed to extract sparse features of scene images. These features can be considered

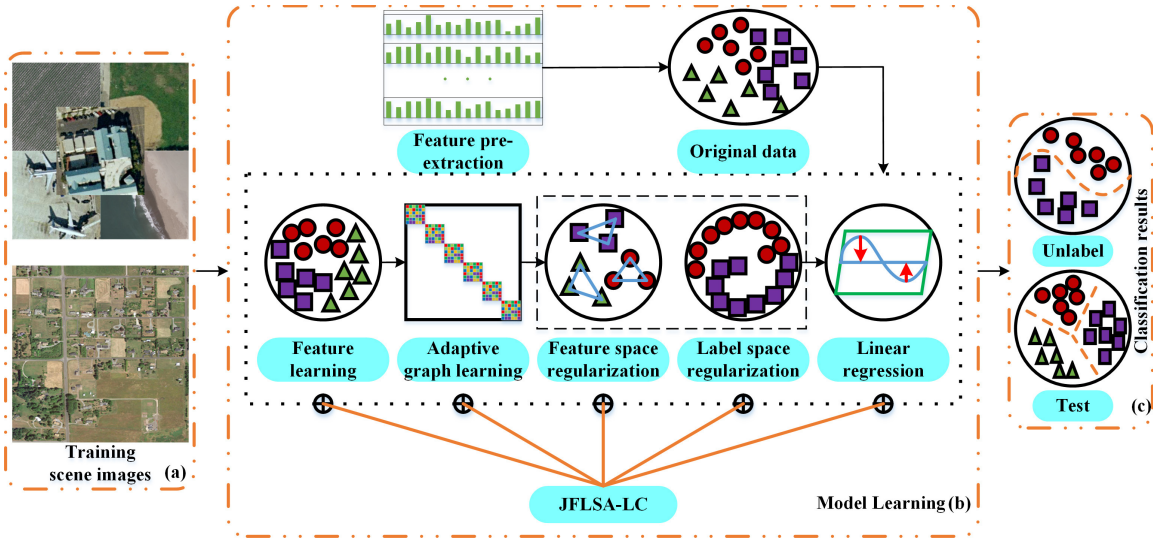

Figure 1. Workflow of the JFLSA_FL for aerial scene classification. (a) The training aerial scene images; (b) The learning process of the JFLSA_FL; (c) Classification results.

as middle-level features. Using these features, SVM was adopted to classify these images. Here the feature extraction procedure was separated with the classifier training process. This maybe cannot well explore the feature data self-adaption to achieve good results of scene classification. In order to describe the HSR-RSIs efficiently, the high-level feature was learned for scene classification in [10]. But feature learning and label regularization didn't form a union framework. In [23], aiming at constructing a good graph to discover the intrinsic data structures under a semi-supervised learning setting, the authors overlooked the role of label space constraint in feature extraction. In [25], feature subspace learning formulated an optimization problem for learning a transformation from the original signal domain to a lower-dimensional one in a way that preserves the sparse structure of data feature, but feature and label space regularization were not well integrated into the subspace learning. From the above observations, not only in geoscience field [7, 8, 9, 10, 11], but also in recognition works of computer vision [23, 24, 25], feature extraction procedure is often separated from classifier training. It means that the objective function of feature extraction is isolated from the constraints of the latent feature space and label space. Therefore, the result of feature extraction is sub-optimal.

To learn more appropriate object-based features for HSR images and alleviate the semantic gaps, we develop a novel feature learning framework, i.e. **J**oint **F**eature-**S**pace-**A**ware and **L**abel-**S**pace-**A**ware **F**eature **L**earning (JFLSA_FL). The workflow of the JFLSA_FL for scene classification is shown in Fig. 1. In the JFLSA_FL, the feature learning model is first constructed for obtaining the object-based features. Then the adaptive graph is employed to describe the relationships among scene image for enhancing the adaptability of learned features. To deal with out-of-sample images, the linear regression is adopted to project the semi-supervised classification results into the linear classifier. To joint both feature space and label space, we combine the feature learning, graph learning, latent feature space constraint, label space regularization and linear regression into an objective function. The objective function is minimized efficiently by means of the linearized alternating direction method with adaptive penalty (LADMAP) [3]. We test our method on three widely used aerial scene image datasets. The experimental results show that our method can outperform most of the existing scene classification algorithms. The main contributions of this paper are summarized as follows:

(1) By jointly exploiting the characteristics of latent feature space and label space, we propose a novel object feature learning method for aerial scene classification, which significantly boosts the performance.

(2) To make the object-based features have the data adaptability, we construct an adaptive graph. The graph is embedded is embedded in the constrained joint space for features and labels to increase feature discriminability.

(3) We develop an effective and efficient optimization algorithm to solve the proposed objective function. The theoretical and experimental analyses reveal the fast convergence of the designed optimization algorithm.

2. JFLSA_FL

In this section, we first describe the JFLSA_FL. Later the optimization procedure of the objective function is discussed in the following sub-sections.

2.1. Feature Pre-extraction

In order to learn more appropriate object-based features for scene classification, the SIFT features of scene images are extracted from HSR-RSIs. These features are considered as low-level features. Next, the sparse representations that measure the responses of each local feature descriptor to the dictionary's "visual elements" are solved. Finally, the responses are pooled across different spatial locations via the maximum pooling over different spatial scales [22], resulting in a set of local features $\mathbf{Z} \in \mathbb{R}^{d\times n}$ for the images, which belongs to middle-level features. For improving computational efficiency, these features are processed with the PCA. Based on reduction features, the JFLSA_FL learns more representative and discriminative features for aerial scene classification.

2.2. Formulation of Proposed Framework

Based on the basic feature $\mathbf{Z}$ defined in Section 2.1, we aim to learn more representative features for the object-based scene classification. In order to make the learned features preserve the information of the original data, the intrinsic data structure of the feature and label manifolds are exploited simultaneously. To achieve the aim, we develop JFLSA_FL to obtain discriminative features of the objects in HSR-RSIs for scene classification. The proposed JFLSA_FL consists of three main components: the basic feature learning model (global reconstruction constraint), the latent feature space regularization and the label space regularization. The basic feature learning model encodes the learned features by using the functions of the reconstruction error and sparsity control. In order to keep the learned features in the manifold assumptions of the feature and label spaces, we construct the adaptive graph by computing the sparse relationships between different image features. We use the adaptive graph integrating the data and label manifold assumptions to regularize the learned features. To classify the out-of-sample scene images efficiently, we apply the linear regression to project the learned the manifold classifier to the linear classifier.

2.2.1 Definition of Feature Learning

Based on the unsupervised feature learning algorithms [1, 2], the JFLSA_FL is to minimize the following objective function:

$$\Theta_1(\mathbf{W}) = \left\|\mathbf{W}^T\mathbf{W}\mathbf{Z} - \mathbf{Z}\right\|_F^2 + \alpha g(\mathbf{W}\mathbf{Z}) \qquad (1)$$

where $\mathbf{Z}$ is a set of d-dimensional feature vectors obtained from images $\mathbf{X} = \{x_1, x_2, \cdots, x_n\}$, i.e. $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_n] \in \mathbb{R}^{d\times n}$, $\mathbf{W} \in \mathbb{R}^{d'\times d}$ is the feature transformation matrix which projects $\mathbf{Z}$ to a d'-dimensional feature space. g is the nonlinear convex function which is defined as a smooth penalty function, such as $g(\cdot) = \log(\cosh(\cdot))$ [1], and α is a tradeoff factor. By means of the first term in Eq. (1), more representative features can be obtained according to the error minimization between the reconstructed data and the original data. The second term g produces sparse representations of the data.

2.2.2 Adaptive Graph Construction

After the feature learning, we get new d'-dimensional feature $\mathbf{WZ}$ instead of the original feature $\mathbf{Z}$. The model defined in Eq. (1) aims to get more representative features only by retaining the original information, but it overlooks the data adaptation, and thus the learned features are lack of discrimination. To add the data adaptation of scene images, the feature learning model should also meet the following assumptions:

I. If two images x_i and x_j have a close relationship in the intrinsic geometry of the data distribution, then x_i and x_j have similar feature structures in **feature space**;

II. If x_i and x_j have a close relationship under the data distribution, then x_i and x_j have the coherent labels in **label space**;

III. If x_i and x_j have the same original labels, x_i and x_j should produce the label results essentially in agreement with each other under **label space**.

To model the intrinsic geometrical structures among different images, an adaptive graph $\mathbf{G}$ is usually designed to express the relationships among the images. In [26, 27, 28], the graphs are constructed by selecting the nearest neighbors and defining the similarities of images manually. They are hard to well reflect the relationships among images since the intrinsic feature structures of data are ignored. In our adaptive graph $\mathbf{G}$, each vertex corresponds to one image x_i, and its nearest neighbors are selected according to the similarities (weight matrix) $\mathbf{U}$, which is computed between one sample and other samples and can be defined by means of the following function:

$$\begin{aligned} &\Theta_2(\mathbf{W},\mathbf{S}) = \|\mathbf{WZ} - \mathbf{WZS}\|_F^2 + \alpha_2 \|\mathbf{S}\|_1 \\ &s.t.\ \ \forall i, \mathbf{S}_{ii} = 0. \end{aligned} \qquad (2)$$

$$\mathbf{U}_{ij} = \frac{\mathbf{S}_{ij} + \mathbf{S}_{ji}}{2} \qquad (3)$$

where the reconstruction coefficient matrix $\mathbf{S}$ is learned by the reconstruction and sparsity functions, and it essentially reflects a close relation between the sample pairs; α_2 is used to balance the two functions. The first term is to minimize the linear reconstruction error, and the second term is to control the sparsity of $\mathbf{S}$ by using l_1 norm.

The edges of the adaptive graph are assigned between the image and its neighbors according to the weight matrix $\mathbf{U}$. From the definition of $\mathbf{U}$ in Eqs. (2) and (3), it is observed that the nearest neighbors are selected automatically,

and the similarities among scene images are computed by means of intrinsic geometric structure of the data. Thus the constructed graph is more adaptive and competitive than the manual defined graphs.

2.2.3 Feature Space Regularization

Under the assumption I, we encode the learned features which simultaneously preserve the local visual similarity among different images and satisfy the manifold assumption, i. e. if two images are similar to each other, their representations are close to each other [5], and the corresponding objective function is expressed as

$$\begin{aligned}\Theta_3(\mathbf{W},\mathbf{S}) &= \frac{1}{2}\sum_{i,j=1}^{n}\mathbf{U}_{ij}\left\|(\mathbf{WZ})_i-(\mathbf{WZ})_j\right\|_2^2 \\ &= tr\left((\mathbf{WZ})\mathbf{L}(\mathbf{WZ})^T\right)\end{aligned} \tag{4}$$

where $\mathbf{U}$ is the weight matrix of the adaptive graph $\mathbf{G}$. $\mathbf{D}$ is a diagonal matrix of which the (i,i)-th element equals to the sum of the i-th row of $\mathbf{U}$. Then $\mathbf{L}=\mathbf{D}-\mathbf{U}$.

In Eq. (4), we add the manifold constraint to the matrices $\mathbf{W}$ and $\mathbf{S}$ to ensure the neighboring scene images have similar feature structures.

2.2.4 Label Space Regularization

To satisfy the assumptions II and III, we hope that intra-class images have more similar features, and inter-class image features have larger diversifications. To achieve more discriminative features, we add manifold smoothness with image labels to regularize the feature learning process. In the label manifold smoothness regularization, we jointly learn the relevance scores $\mathbf{F}$ with the relationships among different images. The objective function can be defined in Eq. (5).

$$\begin{aligned}\Theta_4(\mathbf{F},\mathbf{S}) &= \frac{1}{2}\sum_{i,j=1}^{n}\mathbf{U}_{ij}\left\|\mathbf{F}_i-\mathbf{F}_j\right\|_2^2+\sum_{i=1}^{n}\left\|\mathbf{F}_i-\mathbf{Y}_i\right\|_2^2 \\ &= tr\left(\mathbf{F}^T\mathbf{LF}\right)+tr\left((\mathbf{F}-\mathbf{Y})^T\mathbf{V}(\mathbf{F}-\mathbf{Y})\right)\end{aligned} \tag{5}$$

where $\mathbf{V}\in\mathbb{R}^{n\times n}$ is the diagonal matrix, in which the labeled samples in $\mathbf{X}$, $\mathbf{V}_{ii}=1$; otherwise $\mathbf{V}_{ii}=0$.

In the first term of Eq. (5), the classification scores and the relationships among different images can be optimized simultaneously. The second term introduces the actual label data integrating the first term to ensure to learn the classification results with local and global consistencies. Therefore, the neighboring images have same labels and the scene classification results are consistent with the given image labels.

2.2.5 Linear Regression

From Eq. (5), we can compute the scene classification results by using the manifold assumption. These results reflect the probabilities of the vertices belonging to a certain category. We only obtain the scene classification results of these images that are vertices. For the new scene images (we call these images as out-of-sample data), we need to add these images to $\mathbf{G}$. Constructing a new adaptive graph is very time consuming. In order to classify out-of-sample data efficiently, the linear regression [4] is adopted to transform $\mathbf{F}$ into the linear classifier:

$$\Theta_5(\mathbf{W},\mathbf{F},\mathbf{H}) = \left\|(\mathbf{WZ})^T\mathbf{H}-\mathbf{F}\right\|_F^2 \tag{6}$$

In Eq. (6), we use the projection matrix $\mathbf{H}$ with feature transformation $\mathbf{W}$ to classify the out-of-sample scene images.

Consequently, The JFLSA_FL with the adaptive graph can be defined as:

$$\begin{aligned}&\min_{\mathbf{W},\mathbf{S},\mathbf{F},\mathbf{H}} \underbrace{\Theta_1(\mathbf{W})}_{reconstruction}+\lambda_1\Theta_2(\mathbf{W},\mathbf{S})+\underbrace{\lambda_2\Theta_3(\mathbf{W},\mathbf{S})}_{feature\text{-}space\text{-}aware} \\ &+\underbrace{\lambda_3\Theta_4(\mathbf{F},\mathbf{S})}_{label\text{-}space\text{-}aware}+\lambda_4\Theta_5(\mathbf{W},\mathbf{F},\mathbf{H}) \\ &s.t.\ \forall i,\mathbf{S}_{ii}=0.\end{aligned} \tag{7}$$

The four terms $\mathbf{W}$, $\mathbf{S}$, $\mathbf{F}$ and $\mathbf{H}$ in Eq. (7) need to be optimized. The difficulty is that $\mathbf{W}$, $\mathbf{S}$ and $\mathbf{F}$ are closely interdependent to each other. Thus the auxiliary matrices $\mathbf{M}$ and $\mathbf{N}$ are introduced to separate Eq. (7). We set $\mathbf{S}=\mathbf{M}$ and $\mathbf{S}=\mathbf{N}$. Then this objective function is transformed into a new style in Eq. (8) by means of LADMAP. In Eq. (8), $\mathbf{T}_1$ and $\mathbf{T}_2$ are Lagrangian multipliers, and $\mu>0$ is a penalty parameter. $\mathbf{L}$ is constructed by the matrix $\mathbf{M}$, and $\beta=\lambda_1\alpha_2$. Eq. (8) describes an unconstrained problem, and it can be optimized with respect to $\mathbf{W}$, $\mathbf{S}$, $\mathbf{F}$ and $\mathbf{H}$ by fixing other variables. With some algebra, the updating schedule is described in the following section.

$$\begin{aligned}&L(\mathbf{W},\mathbf{S},\mathbf{F},\mathbf{H},\mathbf{M},\mathbf{N},\mathbf{T}_1,\mathbf{T}_2,\mu) \\ &=\left\|\mathbf{W}^T\mathbf{WZ}-\mathbf{Z}\right\|_F^2+\alpha g(\mathbf{WZ})+\lambda_1\left\|\mathbf{WZ}-\mathbf{WZS}\right\|_F^2 \\ &+\beta\left\|\mathbf{N}\right\|_1+\lambda_2 tr\left((\mathbf{WZ})\mathbf{L}(\mathbf{WZ})^T\right) \\ &+\lambda_3\left(tr\left(\mathbf{F}^T\mathbf{LF}\right)+tr\left((\mathbf{F}-\mathbf{Y})^T\mathbf{V}(\mathbf{F}-\mathbf{Y})\right)\right) \\ &+\lambda_4\left\|(\mathbf{WZ})^T\mathbf{H}-\mathbf{F}\right\|_F^2 \\ &+\frac{\mu}{2}\left(\left\|\mathbf{S}-\mathbf{M}+\frac{\mathbf{T}_1}{\mu}\right\|_F^2+\left\|\mathbf{S}-\mathbf{N}+\frac{\mathbf{T}_2}{\mu}\right\|_F^2\right) \\ &-\frac{1}{2\mu}\left(\left\|\mathbf{T}_1\right\|_F^2+\left\|\mathbf{T}_2\right\|_F^2\right) \\ &s.t.\ \forall i,\mathbf{N}_{ii}=0.\end{aligned} \tag{8}$$

3. Optimization of JFLSA_FL

3.1. Optimization for H

H is solved when **W**, **S** and **F** are fixed. The optimization problem defined in Eq. (8) is written as Eq. (9):

$$\min_{\mathbf{H}} L(\mathbf{H}) = \min_{\mathbf{H}} \lambda_4 \left\| (\mathbf{WZ})^T \mathbf{H} - \mathbf{F} \right\|_F^2 \tag{9}$$

Eq. (9) presents an unconstrained optimization problem. If $\mathbf{WZZ}^T\mathbf{W}^T$ is a singular square matrix, let $\mathbf{B} = (\mathbf{WZZ}^T\mathbf{W}^T + \mu\mathbf{I})^{-1}\mathbf{WZ}$; otherwise let $\mathbf{B} = (\mathbf{WZZ}^T\mathbf{W}^T)^{-1}\mathbf{WZ}$, where μ is a small positive constant, and **I** is the identity matrix. By setting the derivative $\partial L(\mathbf{H})/\partial \mathbf{H} = 0$, we obtain Eq. (10):

$$\mathbf{H} = \mathbf{BF} \tag{10}$$

3.2. Optimization for W

When the values of **F**, **S** and **H** are fixed, we can rewrite Eq. (8) as follows:

$$\begin{aligned} &\min_{\mathbf{W}} L(\mathbf{W}) = \min_{\mathbf{W}} \left\| \mathbf{W}^T\mathbf{WZ} - \mathbf{Z} \right\|_F^2 + \alpha g(\mathbf{WZ}) \\ &+ \lambda_1 \left\| \mathbf{WZ} - \mathbf{WZS} \right\|_F^2 + \lambda_2 tr\left((\mathbf{WZ})\mathbf{L}(\mathbf{WZ})^T \right) \\ &+ \lambda_4 \left\| (\mathbf{WZ})^T \mathbf{H} - \mathbf{F} \right\|_F^2 \end{aligned} \tag{11}$$

Given a training data matrix **Z**, we can compute the function cost and the gradient of Eq. (11). Then the objective function defined in Eq. (11) is minimized by using the unconstrained optimizer (e.g., L-BFGS or CG [6]) to update **W**.

3.3. Optimization for F

When the values of **W**, **S** and **H** are fixed, Eq. (8) can be written as

$$\begin{aligned} &\min_{\mathbf{F}} L(\mathbf{F}) = \min_{\mathbf{F}} \lambda_3 tr\left(\mathbf{F}^T\mathbf{LF}\right) \\ &+ \lambda_3 tr\left((\mathbf{F}-\mathbf{Y})^T\mathbf{V}(\mathbf{F}-\mathbf{Y}) \right) + \lambda_4 \left\| (\mathbf{WZ})^T\mathbf{H} - \mathbf{F} \right\|_F^2 \end{aligned} \tag{12}$$

By substituting the expression for **H** in Eq. (10) into Eq. (12), it is an unconstrained optimization problem. The the derivative of Eq. (12) with respect to **F** is set to zero, then we can have:

$$\mathbf{F} = \left(\mathbf{L} + \mathbf{V} + \frac{\lambda_4}{\lambda_3}\mathbf{A} \right)^{-1} \mathbf{VY} \tag{13}$$

where $\mathbf{A} = \left(\mathbf{Z}^T\mathbf{W}^T\mathbf{B} - \mathbf{I}\right)^T \left(\mathbf{Z}^T\mathbf{W}^T\mathbf{B} - \mathbf{I}\right)$.

3.4. Optimization for S

When given the values of **W**, **F** and **H**, we can compute and update the matrix **S**. Eq. (8) can be rewritten as follows:

$$\begin{aligned} &\min_{\mathbf{S}} L(\mathbf{S}) = \min_{\mathbf{S}} \lambda_1 \left\| \mathbf{WZ} - \mathbf{WZS} \right\|_F^2 \\ &+ \frac{\mu}{2}\left(\left\| \mathbf{S} - \mathbf{M} + \frac{\mathbf{T}_1}{\mu} \right\|_F^2 + \left\| \mathbf{S} - \mathbf{N} + \frac{\mathbf{T}_2}{\mu} \right\|_F^2 \right) \end{aligned} \tag{14}$$

The the derivative of Eq. (14) with respect to **S** is set to zero, and we can have the updating rule for S:

$$\mathbf{S} = \left(2\lambda_1\mathbf{R}^T\mathbf{R} + 2\mu\mathbf{I}\right)^{-1} \times \left(2\lambda_1\mathbf{R}^T\mathbf{R} - \mu\left(\mathbf{C}_1 + \mathbf{C}_2\right)\right) \tag{15}$$

where

$$\mathbf{R} = \mathbf{WZ} \tag{16}$$

$$\mathbf{C}_1 = \frac{\mathbf{T}_1}{\mu} - \mathbf{M} \tag{17}$$

$$\mathbf{C}_2 = \frac{\mathbf{T}_2}{\mu} - \mathbf{N} \tag{18}$$

3.5. Optimization for M

Since we know the Laplacian matrix **L** is constructed by the matrix **M**, the Lagrangian function (8) can be rewritten in the following format:

$$\begin{aligned} &\min_{\mathbf{M}} L(\mathbf{M}) = \min_{\mathbf{M}} \lambda_2 tr\left((\mathbf{WZ})\mathbf{L}(\mathbf{WZ})^T \right) \\ &+ \lambda_3 tr\left(\mathbf{F}^T\mathbf{LF}\right) + \frac{\mu}{2}\left\| \mathbf{S} - \mathbf{M} + \frac{\mathbf{T}_1}{\mu} \right\|_F^2 \end{aligned} \tag{19}$$

Substitute Eqs. (3), (16) and (17) for the variables in the Eq. (19), and we can get:

$$\begin{aligned} L(\mathbf{M}) = &\frac{1}{2}\sum_{i,j=1}^{n} \frac{\mathbf{M}_{ij}+\mathbf{M}_{ji}}{2}\left(\lambda_2\|\mathbf{R}_i - \mathbf{R}_j\|_2^2 + \lambda_3\|\mathbf{F}_i - \mathbf{F}_j\|_2^2\right) \\ &+ \frac{\mu}{2}\left\| \mathbf{S} - \mathbf{M} + \frac{\mathbf{T}_1}{\mu} \right\|_F^2 \end{aligned} \tag{20}$$

In order to solve the problem described in Eq. (20), we can first compute the function cost and gradient. And by means of the unconstrained optimizer, **M** can be updated by L-BFGS or CG.

3.6. Optimization for N

By fixing other variables except for **N**, the Lagrangian function of Eq. (8) can be rewritten as:

$$\min_{\mathbf{N}} L(\mathbf{N}) = \min_{\mathbf{N}} \beta \|\mathbf{N}\|_1 + \frac{\mu}{2}\left\| \mathbf{S} - \mathbf{N} + \frac{\mathbf{T}_2}{\mu} \right\|_F^2 \tag{21}$$
$$s.t.\ \ \forall i, \mathbf{N}_{ii} = 0.$$

From Eq. (21), $\mathbf{N}$ can be computed by the following:

$$(\mathbf{N})_{new} \Leftrightarrow \arg\min_{\mathbf{N}} \left\| \mathbf{S} - \mathbf{N} + \frac{\mathbf{T}_2}{\mu} \right\|_F^2 + \frac{2\beta}{\mu} \|(\mathbf{N})_{old}\|_1 = \Phi\left(\mathbf{S} + \frac{\mathbf{T}_2}{\mu}\right) \quad (22)$$

where Φ is the shrinkage operator.

The Lagrangian multipliers can be updated as follows:

$$\begin{aligned} (\mathbf{T}_1)_{new} &= (\mathbf{T}_1)_{old} + \mu(\mathbf{S} - \mathbf{M}), \\ (\mathbf{T}_2)_{new} &= (\mathbf{T}_2)_{old} + \mu(\mathbf{S} - \mathbf{N}) \end{aligned} \quad (23)$$

Given the output of Algorithm 1, we use the transformations $\mathbf{W}$ and $\mathbf{H}$ as shown in Eq. (24) to classify scene images.

$$label = \arg\max_j \left(\mathbf{Z}_i^T \mathbf{W}^T \mathbf{H}\right) (1 \le j \le m) \quad (24)$$

Algorithm 1: JFLSA_FL

Input: Training set $\mathbf{Z}$, parameters α, β, λ_1, λ_2, λ_3 and λ_4;

Initialization: $\mathbf{W}$ and $\mathbf{S}$; $\mu_0 = 0.1$, $\mu_{\max} = 10^{10}$, $\rho_0 = 1.1$.

repeat

1: Update $\mathbf{W}$ by Eq. (11);
2: Update $\mathbf{S}$ by Eq. (14);
3: Construct the graph, and calculate $\mathbf{L}$;
4: Update $\mathbf{F}$ by Eq. (12);
5: Update $\mathbf{H}$ by Eq. (9);
6: Update $\mathbf{M}$ by Eq. (19);
7: Update $\mathbf{N}$ by Eq. (21);
8: Update Lagrange multipliers by Eq. (23);
9: Update μ by $\mu = \min(\rho\mu, \max\mu)$;
10: Update $t = t + 1$;
11: Obtain the optimal solution $\mathbf{W}$, $\mathbf{S}$, $\mathbf{F}$ and $\mathbf{H}$.

Output: The matrix $\mathbf{F}$ and $\mathbf{H}$.

4. Experiments

In this section, we evaluate the performance of the JFLSA_FL for scene classification. We first briefly describe the experimental data. Afterwards, we compare the classification results of our method with those of the related approaches.

4.1. Datasets

Three different scene image datasets are used in the experiment. They are the UC Merced Land Use–Land Cover (LULC) dataset (21-class) [29], 19-class satellite scene dataset (19-Class) [30, 31], and a large complete aerial scene image from Seattle:

(1) The UC Merced dataset is downloaded from the U.S. Geological Survey (USGS) National Map[1]. It consists of images categorized into 21 classes, with a pixel resolution of 30 cm. Each class contains 100 RGB color samples of size 256 × 256 pixels, examples of which are shown in Fig. 2. The dataset represents highly overlapping classes, such as dense residential, medium residential and sparse residential, which have a large difference in the structures.

(2) The second dataset is 19-class satellite scene dataset[2]. This dataset has 19 categories scene images, in which each class has more than 50 scene images, with a size of 600 × 600 pixels. The examples of this dataset are shown in Fig. 3.

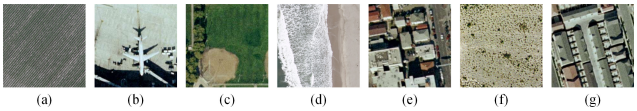

Figure 2. Some examples of the UC Merced Land Use–Land Cover data set. (a) Agricultural. (b) Airplane. (c) Baseball diamond. (d) Beach. (e) Buildings. (f) Chaparral. (g) Dense residential.

(3) The third image dataset is constructed from a large satellite image, Seattle, American (Seattle-Scene) which is acquired from Google Earth. The spatial resolution of this scene image is 0.4 m. The large image to be annotated is of 5120×3072 pixels, as shown in Fig. 4. It mainly contains six categories of land covers: bare land, forest, meadow, ocean, residential, and road. The image is partitioned into 1450 non-overlapping patches with 100 × 100 pixels, and the classification algorithms are performed on this patch-based dataset.

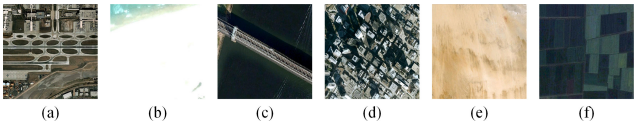

Figure 3. Some examples of 19-class satellite scene data set. (a) Airport. (b) Beach. (c) Bridge. (d) Commercial. (e) Desert. (f) Farmland.

Figure 4. The satellite scene, Seattle, American.

[1] data available at http://vision.ucmerced.edu/datasets/landuse. html.

[2] data available at http://dsp.whu.edu.cn/cn/staff/yw/HRSscene.html.

4.2. Comparisons Methods

Our method is semi-supervised. We compare JFLSA_FL with the following five approaches in terms of classification accuracy: LGC [26], FME [27], NNSG [28], LapRLS [33] and SFSS [34].

The above methods all firstly need to construct the graphs for the semi-supervised classification. For the LGC, FME, LapRLS and SFSS, the weight matrix of the graphs are defined manually, in which the number of the nearest neighbors need to be specified and the distances of images features are computed by means of Euclidean distance. The number of the nearest neighbors is selected from the set {3, 4, 5, 6, 7, 8, 9, 10}. The distances of the image features are calculated by using $\mathbf{U}_{ij} = e^{-\frac{\|\mathbf{z}_i - \mathbf{z}_j\|^2}{\sigma}}$ in the LGC, FME and LapRLS, where σ is the heat kernel, and it is selected from the set $\{10^{-9}, 10^{-8}, \ldots, 10^{8}, 10^{9}\}$. In the SFSS, $\mathbf{U}_{ij} = 1$ when $\mathbf{Z}_i$ and $\mathbf{Z}_j$ are the nearest neighbors; otherwise, $\mathbf{U}_{ij} = 0$. The graph learning functions automatically give the number of the nearest neighbors and compute the image distances in the NNSG and JFLSA_FL. For the FME, NNSG, LapRLS, SFSS and JFLSA_FL, we need the parameters to balance the corresponding terms in the objective functions. These parameters are respectively tuned from the set $\{10^{-9}, 10^{-8}, \ldots, 10^{8}, 10^{9}\}$.

4.3. Scene Classification Results

Our method and the above five methods are employed in the semi-supervised scene classification. In each data set, we randomly select 80% samples as the training data, and the rest are the testing data (out-of-sample data). We further randomly sample s percent of the training data as labeled data. In our experiments, s is set to 10, 20 and 30, respectively. We report the mean scene classification accuracy (AC) and standard deviation (STD) over 20 random splits on the unlabeled data set and unseen test data set, which are referred to as Unlabel and Test, respectively, in Tables 1. Since the LGC can't handle the out-of-sample data, new samples classification results were presented by the FME, NNSG, LapRLS, SFSS and JFLSA_FL.

From Table 1, we have the following observations:

1) Compared with different semi-supervised scene classification results, the FME, NNSG, LapRLS, SFSS and JFLSA_FL usually perform better than the LGC in the three datasets. The FME, NNSG, LapRLS, SFSS and JFLSA_FL all add linear regression into the objective functions. It indicates that the linear regression with other parts jointly constructing the objective function can further improve aerial scene classification performances.

2) Comparing the NNSG with the FME, LapRLS and SFSS, there is no consistent winner on all the databases. The NNSG usually can obtain more accurate scene classification results since the graph learning is adopted in the NNSG. By means of the graph learning, the relationships between images are described more correctly while the graphs in the FME, LapRLS and SFSS are defined manually. But in the Seattle-Scene dataset, the classification accuracy is lower in the NNSG than those of other methods because of the strong variations and mixtures of the patch images in the Seattle-Scene dataset. For example, certain classes in some images are always mixed with other classes, such as bare land and residential.

3) The JFLSA_FL outperforms all the compared methods in Unlabel and Test samples. It is proved that the effectiveness of our method in terms of scene classification accuracy. Thus, it is necessary to unite the feature learning, graph learning, latent feature space constraint, label space regularization and linear regression for feature learning.

4.4. Parameters Analysis

Six parameters α, β, λ_1, λ_2, λ_3 and λ_4 in the JFLSA_FL need to be tuned in each dataset. In the above three datasets, $\alpha = 0.01$ and $\beta = 0.001$. We focus on discussing the influences of different λ_1, λ_2, λ_3 and λ_4 on scene classification results.

From figure 5, it is observed that different datasets obtain the highest scene classification results using the different tuned parameters since each dataset has its own distinctive image features. We also find that λ_1 is larger than λ_2, λ_3 and λ_4 in all datasets, which shows that the adaptive graph plays very important role in feature learning. We employ the adaptive graph to model the relationships between images while other methods like [26, 27] construct the relationships manually. According to the adaptive graph, the JFLSA_FL can well describe the relationships between scene images in the feature space and label space. Then the parameters corresponding to the constraints of the feature space and label space can be evaluated in each dataset.

4.5. Algorithmic Convergence

Simultaneously solving the variables **W**, **S**, **F** and **H** in Eq. (7) is very difficult by directly using gradient descent or Newton's method due to the highly nonlinear nature of Eq. (7). Inspirited by the LADMAP [3], we adopt a customized iterative algorithm to optimize the variables. Here, we prove that the objective function can converge to a local optimum by using Algorithm 1. Six variables **W**, **S**, **F**, **H**, **M** and **N** need to be optimized in Eq. (8). In each iteration, with the help of the unconstrained optimizer L-BFGS, the processes for optimizing **W** and **M** make the objective function achieve a local minimum while other variables are fixed. The functions of optimizing **S**, **F** and **H** are convex, and thus they are convergent. **N** is computed through the shrinkage operator. Its convergence has been proven in [40]. With these optimized variables, the objective function can converge to a local optimum.

Dataset	Method	$s = 10$		$s = 20$		$s = 30$	
		Unlabel	Test	Unlabel	Test	Unlabel	Test
UC Merced	LGC	48.10±1.96		56.66±1.65		60.99±1.76	
	FME	58.66±1.35	57.55±2.23	67.15±1.81	67.35±2.19	71.00±0.85	71.86±1.54
	NNSG	59.10±2.12	57.83±2.80	67.21±1.34	66.58±2.01	70.09±1.13	69.61±1.50
	LapRLS	54.87±1.77	53.92±2.44	60.02±1.89	60.04±2.09	62.15±1.45	61.88±2.90
	SFSS	50.47±2.27	49.44±3.21	61.29±1.79	61.23±2.53	67.08±1.38	67.38±1.89
	JFLSA_FL	**61.09±1.07**	**59.52±1.36**	**70.31±1.26**	**69.56±2.16**	**75.61±1.03**	**74.05±2.16**
19-class scene	LGC	39.36±2.22		47.36±2.26		50.00±1.91	
	FME	53.40±2.61	51.39±3.55	62.60±1.71	59.83±3.41	66.88±2.05	63.14±2.88
	NNSG	53.11±1.42	49.31±2.49	62.21±1.80	58.02±3.35	66.93±1.97	63.16±2.52
	LapRLS	51.91±2.30	48.61±3.36	57.29±3.12	55.40±3.63	60.94±2.10	56.81±3.44
	SFSS	45.06±2.72	44.53±3.56	56.85±2.76	55.12±2.57	62.66±2.32	60.00±3.19
	JFLSA_FL	**55.12±1.06**	**52.52±2.36**	**62.77±1.62**	**60.47±1.76**	**67.69±1.32**	**65.16±2.12**
Seattle-Scene	LGC	37.03±5.65		47.46±4.06		52.29±3.21	
	FME	61.62±1.34	61.35±2.17	65.24±1.77	65.57±2.45	66.42±1.45	66.47±2.05
	NNSG	54.44±1.25	56.70±2.13	57.79±1.30	60.83±3.09	60.95±1.43	62.44±2.07
	LapRLS	61.00±1.57	60.99±2.72	64.03±1.05	64.74±2.96	65.14±1.60	65.54±1.64
	SFSS	58.70±1.77	58.65±2.74	61.43±1.15	62.49±2.81	63.69±1.29	65.90±2.20
	JFLSA_FL	**62.78±1.62**	**62.62±1.69**	**66.58±2.28**	**65.67±1.88**	**68.30±1.26**	**67.61±2.65**

Table 1. Semi-supervised scene classification results (AC% ± STD%) on different datasets. The best results are marked in dark.

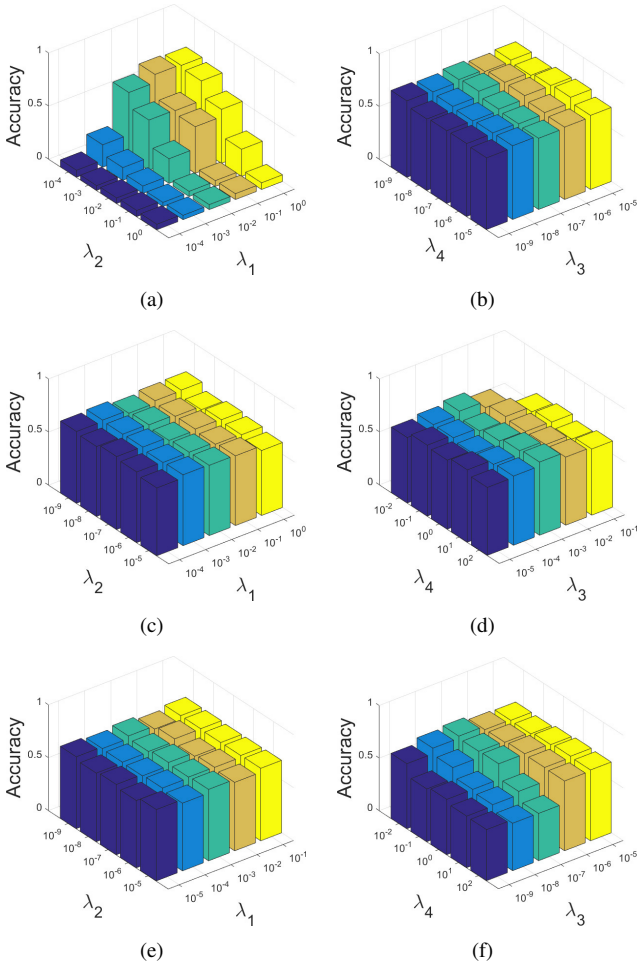

Figure 5. The effects of different values of the paramters on the classification results. (a)–(b) UC Merced dataset. (c)–(d) 19-class scene dataset. (e)–(f) Seattle-Scene dataset.

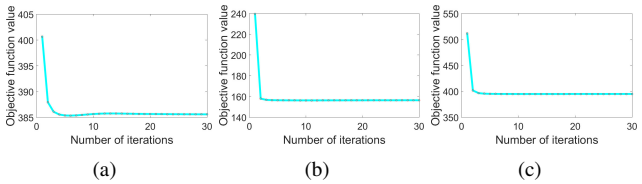

Figure 6. Convergence processes of different datasets. (a) UC Merced. (b) 19-class scene. (c) Seattle-Scene.

The convergence processes under different datasets are shown in Fig. 6. It is noted that the objective function can converge to a local optimum and converge very fast. The objective function usually can reach the convergence within less than 5 iterations for the datasets. Therefore, the proposed update rule in Algorithm 1 is very effective.

5. Conclusions

In this paper, we have proposed a novel object-based feature learning framework JFLSA_FL that integrates the constraints of the feature space and label space with the adaptive graph. The adaptive graph can express the relationships between scene images well. In order to process the out-of-sample data, linear regression is adopted to project semi-supervised classification results into the linear classifier. The solution to the objective function is very effective and efficient. Three aerial scene datasets are used to evaluate the performance of the proposed method. Comparing with the related approaches, our method can achieve better scene classification results.

In feature work, we will combine the JFLSA_FL with deep learning structure to extract multi-level features of the objects for further enhancing performance of aerial scene classification.

References

[1] A. Hyvärinen, J. Hurri, and P. O. Hoyer. Independent component analysis. *Natural Image Statistics*, 2009. 3

[2] Q. V. Le, A. Karpenko, J. Ngiam, and A. Y. Ng. ICA with reconstruction cost for efficient overcomplete feature learning. In *NIPS*, 2011. 3

[3] Z. Lin, R. Liu, and Z. Su. Linearized alternating direction method with adaptive penalty for low rank representation. In *NIPS*, 2011. 2, 7

[4] Y. Yang, F. Wu, F. Nie, H. T. Shen, Y. Zhuang and A. G. Hauptmann. Web and personal image annotation by mining label correlation with relaxed visual graph embedding. *IEEE Transactions on Image Processing*, 2012. 4

[5] J. Liu, Y. Chen, J. Zhang and Z. Xu. Enhancing Low-Rank Subspace Clustering by Manifold Regularization. *IEEE Transactions on Image Processing*, 2014. 4

[6] Schimidt, M. : minfunc., 2005. 5

[7] A. M. Cheriyadat. Unsupervised feature learning for aerial scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 2014. 1, 2

[8] J. Zhao, Y. Zhong, H. Shu, and L. Zhang. High-resolution image classification integrating spectral-spatial-location cues by conditional random fields. *IEEE Transactions on Image Processing*, 2016. 1, 2

[9] J. Zhao, Y. Zhong, and L. Zhang. Detail-preserving smoothing classifier based on conditional random fields for high spatial resolution remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 2015. 1, 2

[10] W. Yang, X. Yin, and G.-S. Xia. Learning high-level features for satellite image classification with limited labeled samples. *IEEE Transactions on Geoscience and Remote Sensing*, 2015. 1, 2

[11] Y. Zhong, Q. Zhu, and L. Zhang. Scene classification based on the multifeature fusion probabilistic topic model for high spatial resolution remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 2015. 1, 2

[12] Y. Wang, L. Zhang, X. Tong, L. Zhang, Z. Zhang, H. Liu, X. Xing, and P. T. Mathiopoulos. A Three-Layered Graph-Based Learning Approach for Remote Sensing Image Retrieval. *IEEE Transactions on Geoscience and Remote Sensing*, 2016. 1

[13] J. C. Tilton, Y. Tarabalka, P. M. Montesano, and E. Gofman. Best merge region-growing segmentation with integrated nonadjacent region object aggregation. *IEEE Transactions on Geoscience and Remote Sensing*, 2012. 1

[14] T. Blaschke. Object based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2010. 1

[15] I. A. Rizvi and B. K. Mohan. Object-based image analysis of high resolution satellite images using modified cloud basis function neural net-work and probabilistic relaxation labeling process. *IEEE Transactions on Geoscience and Remote Sensing*, 2011. 1

[16] R. Bellens, S. Gautama, L. Martinez-Fonte, W. Philips, J. C. W. Chan and F. Canters. Improved Classification of VHR Images of Urban Areas Using Directional Morphological Profiles. *IEEE Transactions on Geoscience and Remote Sensing*, 2008. 1

[17] P. Gamba, F Dell'Acqua., G. Lisini, and G. Trianni. Improved VHR urban area mapping exploiting object boundaries, " *IEEE Transactions on Geoscience and Remote Sensing*, 2007. 1

[18] A. Baraldiand and F. Parmiggiani. An investigation of the textural characteristic associated with gray level cooccurrence matrix statistical parameters. *IEEE Transactions on Geoscience and Remote Sensing*, 1995. 1

[19] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004. 1

[20] N. Dalal and B. Triggs. Histograms of Oriented Gradients for Detection. In *CVPR*, 2005. 1

[21] L. Wu, S. C. H. Hoi, and N. Yu. Semantics-preserving bag-of-words models and applications. *IEEE Transactions on Image Processing*, 2010. 1

[22] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *ICCV*, 2009. 1, 3

[23] L. Zhuang, S. Gao, J. Tang, J. Wang, Z. Lin, Y. Ma and N. Yu. Constructing a nonnegative low-rank and sparse graph with data-adaptive features. *IEEE Transactions on Image Processing*, 2015. 2

[24] V. M. Patel, H. V. Nguyen, and R. Vidal. Latent space sparse subspace clustering. In *ICCV*, 2013. 2

[25] H. V. Nguyen, V. M. Patel, N. M. Nasrabadi, and R. Chellappa. Sparse embedding: A framework for sparsity promoting dimensionality reduction. In *ECCV*, 2012. 2

[26] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *NIPS*, 2003. 3, 7

[27] F. Nie, D. Xu, I. W. Tsang, and C. Zhang. Flexible manifold embedding: A framework for semi-supervised and unsupervised dimension reduction. *IEEE Transactions on Image Processing*, 2010. 3, 7

[28] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 2006. 3, 7

[29] Y. Yang and S. Newsam. Spatial pyramid co-occurrence for image classification. In *ICCV*, 2011. 6

[30] G. Xia, W. Yang, J. Delon, Y. Gousseau, H. Sun and H. Maitre. Structural high-resolution satellite image indexing. In *ISPRS TC VII Symp.-100 Years*, 2010. 6

[31] D. Dai and W. Yang. Satellite image classification via two-layer sparse coding with biased image representation. *IEEE Geoscience and Remote Sensing Letters*, 2011. 6

[32] X. Fang, Y. Xu, X. Li, Z. Lai, and W. K. Wong. Learning a nonnegative sparse graph for linear regression. *IEEE Transactions on Image Processing*, 2015.

[33] V. Sindhwani, P. Niyogi, M. Belkin, and S. Keerthi. Linear manifold regularization for large scale semi-supervised learning. In *ICML Workshop*, 2005. 7

[34] Z. Ma, F. Nie, Y. Yang, J. Uijlings, N. Sebe, and A. Hauptmann. Discriminating joint feature analysis for multime-

dia data understanding. *IEEE Transactions on Multimedia*, 2012. 7

[35] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1

[36] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, 2014. 1

[37] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 1

[38] J. Lu, G. Wang, W. Deng, P. Moulin and J. Zhou. Multi-manifold deep metric learning for image set classification. In *CVPR*, 2015. 1

[39] J. Hu, J. Lu, and Y. Tan. Deep transfer metric learning. In *CVPR*, 2015. 1

[40] M. Figueiredo, R. Nowak, and S. Wright. Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems. *IEEE Journal of Selected Topics in Signal Processing*, 2007. 7